Глубокие самообучающиеся агенты в мультиагентной системе маршрутизации

Мухутдинов Дмитрий, группа M3438
Научный руководитель: Фильченков А. А., к.ф-м.н., доцент кафедры КТ
Рецензент: Тарасов В. Б., к.т.н., МГТУ им. Баумана

Кафедра Компьютерных Технологий
Факультет Информационных Технологий и Программирования
Университет ИТМО, Санкт-Петербург

19 мая 2017 г.

Задача маршрутизации

- Сетевой роутинг
- Транспортная логистика
- Управление конвейерными системами
- Автоматическое управление городским трафиком

Существующие решения

- Link-state
  - Open Shortest Path First (OSPF)
  - IS-IS
- Distance-vector
  - RIP
  - IGRP
- Прочие
  - AntNet
  - ...

Проблемы

- Примерно все алгоритмы маршрутизации заточены под компьютерные сети
- В других задачах существуют свои, более сложные условия
  - Скорую нужно пропустить сквозь пробку, а обычный автомобиль — нет
  - Чемоданы бизнес-класса хочется доставить первыми
  - ...

Задача

Построить алгоритм, способный адаптироваться под гетерогенные условия

Идея

- Обучение с подкреплением
- Нейросети в качестве обучающихся агентов
- Q-routing (Boyan & Littman, 1994)

Постановка задачи в терминах RL

- Рассмотрим *пакет* в сети как обучающегося агента, взаимодействующего с сетью как со средой
- Полное состояние среды неизвестно, состояние текущего роутера — *наблюдение* пакета
- Действие — переход к одному из соседей
- Q-learning:

$$Q(o_t, a_t) \leftarrow r_t + \gamma \cdot \max_{a \in \mathcal{A}_{o_{t+1}}} Q(o_{t+1}, a)$$

- Принцип аналогичный Q-routing:

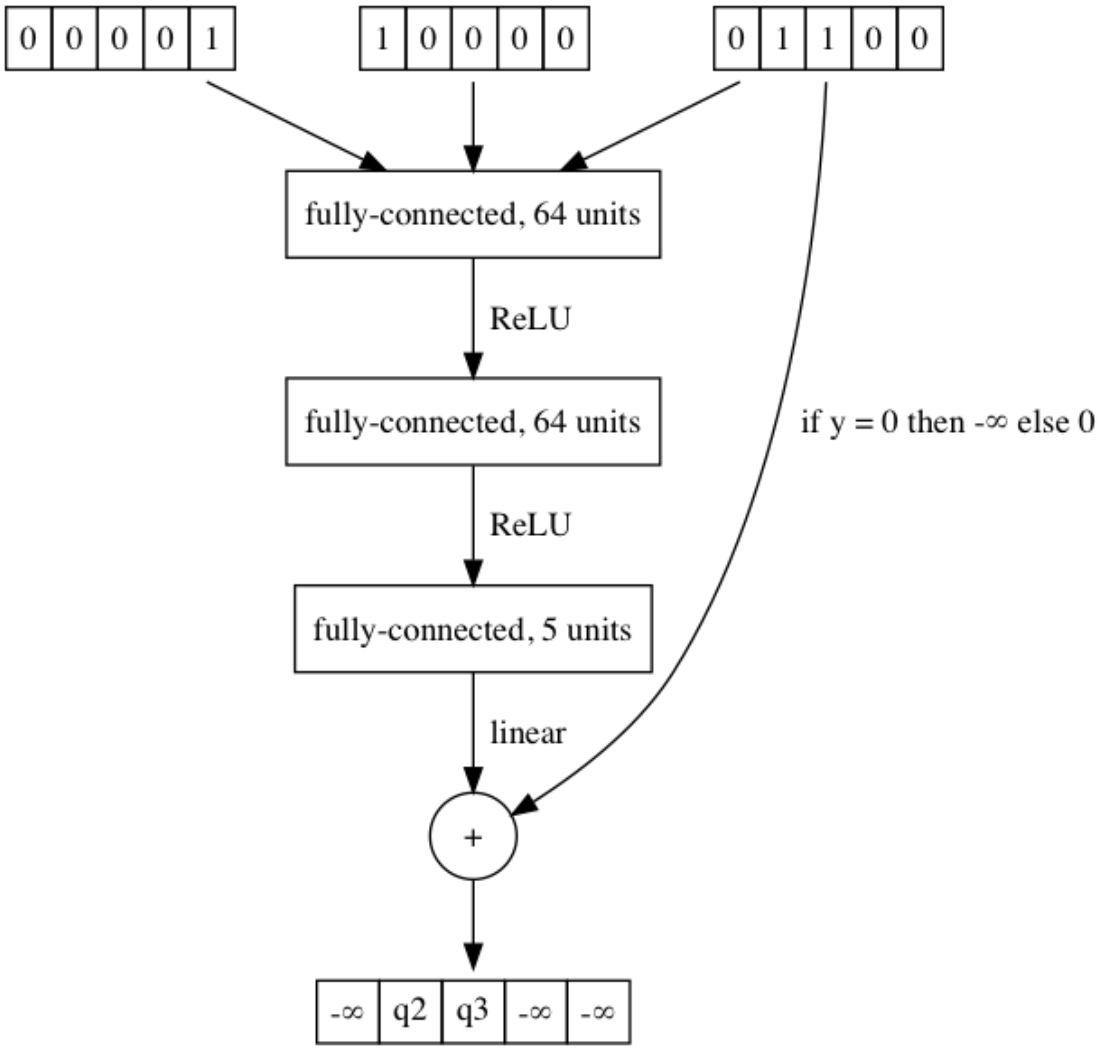
$$Q_x(d, y) \leftarrow (t_{finish} - t_{start}) + \max_{z \in \{V | (y,z) \in E\}} Q_y(d, z)$$

Базовая архитектура нейросети

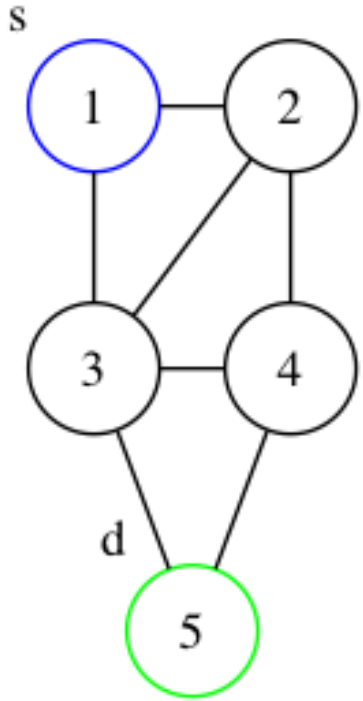
Вход нейросети $o = (d, s, y_1 ... y_n, o')$, где:

- d — узел назначения, s — текущий узел, $y_1 ... y_n$ — номера соседей, o' — любая дополнительная информация
- Выходы нейросети $a_1 ... a_n$ — оценки $Q(o, a)$ для всех узлов сети ($-\infty$ для узлов, не являющихся соседями)
- Кодируем номера унитарным кодом, чтобы избежать корреляции результатов
- Используем RMSProp для оптимизации

Базовая архитектура нейросети



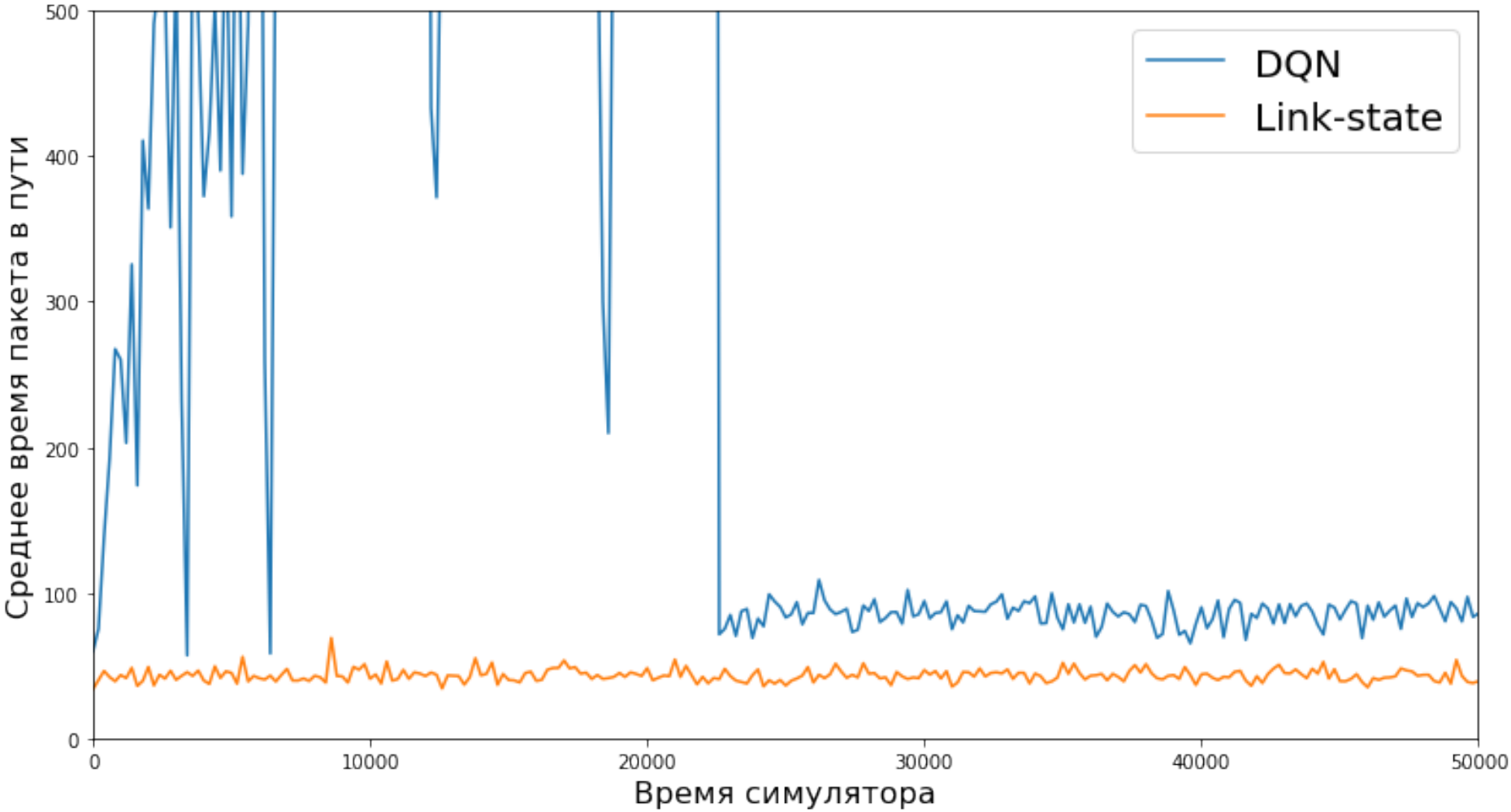
Пример сети для работы в графе из 5 вершин



Проблема нестационарности

- Q-обучение, вообще говоря, не сходится к оптимальному решению в случае бесконечного числа состояний
- Другие агенты меняют свое поведение — среда нестационарна, вследствие чего experience replay (Mnih et al., 2015) не работает
- При обучении с нуля на равномерной низкой нагрузке нейросети не могут найти оптимальные пути

Проблема нестационарности



Решение проблемы нестационарности

- Предобучение сети (bootstrapping)
  - Собираем данные работы алгоритма кратчайших путей
  - Обучаем одну нейросеть на данных от всех роутеров
  - Предобученная сеть повторяет работу алгоритма кратчайших путей
- Отказ от experience replay во время работы
  - Условия работы в сети меняются
  - Старый опыт перестает быть актуальным
  - Если обращать на него внимание, адаптивность к изменяющимся условиям страдает