사전 발표자료 – 강민수

1

세미나 전 사전 발표 자료입니다

2

세웠던 목표는 교통 공학 용어 정리 및 학습, Sumo 네트워크 제작 및 TraCI 함수에 대해서 어떤 것들이 있는지 찾아보았습니다. 강화학습에 대한 내용을 기반으로 하므로 과거 기억을 되살려보려고 노력하였습니다. 이후, 논문 리뷰와 현재 SUMO 기반 신호최적화 repository에서의 action, reward state가 어떻게 구성되어 있는지 확인하였습니다. 마지막으로는 이를 기반으로 개선점 및 실험을 어떻게 진행해보면 좋을지 제안 구성을 고민하였습니다.

3

TOD는 고정 시간으로 사용자 입력 시간에 따라 매일 반복하여 신호를 제어하는 것으로, 변동이 적은 교차로에서 사용하면 효율적이라고 합니다. 단점은 주기적으로 사용자의 사전 조사 및 검증이 필요하고, 복잡한 도로나 시간대에 따라 첨두 시간대를 갖는 경우 매우 효율이 낮습니다. 신호시간은 적합 주기와 녹색 시간, offset을 준비한 후 설정된 신호 시간으로 운영됩니다. 가로 축과 지역 별로 유사 교통 패턴을 갖는 경우, 교차로 group을 편성하여 연동 시스템을 구축하여 사용하기도 합니다. offset값은 교차로 양끝간 거리 나누기 주행차량 속도로서 inflow에 있는 차량이 다음 교차로에서 inflow로 가는데까지 걸리는 시간 값을 의미합니다. 여기서 주행차량 속도는 주로 요구되는 속도에 따라 다르게 설정합니다.

4

신호 시스템을 구축하는데에 있어서 고려해야할 사항입니다. 신호 교차로간 거리, 도로 운영, 접근로의 상태, 차량의 도착 특성이나 시간대에 따라 교통량이 변화하므로 이를 고려해야합니다. 앞으로의 연구에 있어서 까다로운 점은 도로 운영에서 신호등이 없는 횡단 보도로 인한 예상치 못한 신호의 사용에 변동이 있을 수 있다는 점입니다. 또한, 도착 특성이 접근로의 상태와 시간대에 따라 변화하므로 이를 반영할 수 있으면 더 현실화에 가까워질 수 있을 것이라 생각합니다.

5

Spill back은 신호 교차로에서 꼬리물기, 교통량 과다 등의 원인으로 inflow intersection까지 차량이 존재하여 다음 신호의 차량이 이동하기 어려워 지는 현상이며, 이러한 상황은 현재 교통 공학에서 풀기 어려운 난제로 남아있다고 합니다.

6

이러한 문제들을 강화학습으로 풀기 위해 2x2 와 3x3 synthetic grid를 제작하면서 기본적인 sumo의 사용 방법을 다시 익혔습니다.

7

TraCI 함수를 확인하여 어떤 식으로 Agent가 될 Traffic light system이 구성되는지 확인하였습니다. setPhase 함수를 통해 next step에서 가져야할 phase를 설정할 수 있씁니다. setPhaseDuration 함수는 지정된 phase의 시작시점으로부터의 phase를 결정할 수 있습니다. setProgramLogic은 1주기에 대해서 어떤 시간에 어떤 phase set으로 동작할지 설정해주는 함수입니다.

8

TLS program은 기본적인 신호등의 색과 특수 색이 존재하며, 특징 잡아야할 것 만 표시하였습니다.

9

TLS state 부여방식은 북동남서 방향으로 lane의 방향별로 부여하며, 그림의 경우 한 출발 edge에서 lane수가 3개일 때, 7가지의 state가 존재하여 총 intersection에 28가지의 phase가 존재한다. Junction 형태를 설정하는 경우 이 개수를 조절할 수 있고, Connection설정으로 Intersection에서의 merge 형태를 고를 수 있다.

10

Traffic의 State와 Action을 가져오기 위한 함수입니다. 내용 읽기

11

Traffic의 Observation Data를 가져오기 위한 Detector의 함수입니다. 사용법과 존재 유무를 확인하기 위해서만 작성하였습니다.

12======================================================

제가 리뷰할 논문은 ‘Toward A Thousand Lights: Decentralized Deep Reinforcement Learning for Large-scale Traffic Signal Control’ 이며, AAAI2020에 게재되었습니다.

13

내용은 지금까지 RL을 신호시스템에 적용하기 위한 challenge와 그것들을 어떻게 극복하려고 했는지 다루고, 관련 선행 연구들에 대해서 말씀드리겠습니다. 저도 처음 하는 내용인 만큼 다소 용을 길게 작성하였습니다. 그리고 나서 Approach 부분에서 어떻게 실험을 구성하였고, Super Agent 기반 DQN을 어떻게 적용하였는지와 Model을 어떻게 구성하였는지에 대해서 말씀드리겠습니다. 마지막으로는 추후 실험을 위해서 Improvement proposal 부분을 설명드리고 마치도록 하겠습니다.

14

Traffic congestion은 빠른 도시화로 인해 차량이 늘어남에 따라 증가해왔습니다. 그 결과 온실가스 배출량의 23%를 차지하고, traffic control system에 의한 배출량은 그 중에서 40%를 차지할 정도로 많이 증가해왔습니다. 이에 따라 신호시스템의 최적화와 dynamic adaptive control system에 대한 수요가 꾸준히 증가해왔고, 최근에는 강화학습을 사용하여 traffic signal control을 하려는 시도가 이어져오고 있습니다. 기존에는 Pre-timed control을 통해서 실제 데이터를 기반으로 사전에 정의된 signal plan에 의해서 운용되어 왔습니다. 이후 발전을 통해 queue length 기반으로 판단하는 Actuated control, sensor에 의해 current traffic situation을 판단하는 adaptive control system이 있습니다. 최근에는 optimization-based control을 통해서 사람의 경험에 덜 의존하고 관측된 데이터에 의해서 plan을 결정했지만, dynamic traffic에 대처하지는 못했습니다.

15

기존 연구에서는 single-intersection에서 진행해왔지만, 교차로들 끼리 highly correlated되어 있고, 실제에 적용하기 위해서는 large-scale network에서 실험을 진행해야 하므로 small region에서의 시뮬레이션 보다 넓은 network에서 실험할 수 있는 구조가 필요합니다. 전통적인 방식들은 dynamic하게 반응이 불가능하므로 이를 사용하는 것 또한 어렵다고 합니다. RL을 실제 환경에 적용하기 위해 관심을 가질 키워드로 3가지를 제시하였습니다. 첫째는 scalability로서 large scale에서도 학습을 효율적으로 진행하며, global optimization goal을 동시에 학습을 할 수 있어야 합니다. 둘째로는 coordination으로 가까운 위치에 있을수록 signal timing의 최적화가 연동될 때, 더 높은 효율을 보일 수 있습니다. 마지막으로는 Data feasibility로서 강화학습은 exploration과 exploitation을 통해 다양한 상황을 학습합니다. 하지만, 일어나지 않을 상황을 학습하면 실제 환경에 도움이 되지 않으므로 이를 Dataset으로서 학습하지 않도록 배제해야 함을 의미합니다.

16

이러한 키워드를 가지고 과거의 방식들을 살펴보면, 전통적인 방식에서 multi-intersection control은 동일한 cycle length를 기반으로 구성하여 모든 신호가 균일하게 설정하였습니다. 특히, 선택된 차량 흐름에 대해서는 offset을 조정하여 그 부분만 소통이 원활하게 조정하였습니다. 차량이 출발하면 next intersection의 inflow에 도달할 때, green을 on하여 한쪽 차량의 흐름을 원활하게 하기도 하였습니다. 이러한 방식은 few network에서만 적용가능하여 scalability를 만족하지 못하는 것이 단점입니다. 또한, fixed offset에 의해 global optimization을 진행하기 어렵기 때문에 coordination도 만족하지 못했습니다.

17

Global optimization을 진행하여 coordination을 달성하기 위해, max pressure 기법을 적용하였는데 pressure를 # of inflow vehicles - # of outflow vehicles라고할 때, max pressure를 골라 phase를 선정해주는 control방식입니다. Max pressure를 통해 intersection 모두에서 vehicle travel time과 throughput을 global하게 optimization을 진행할 수 있었습니다. 하지만, simple traffic condition에만 의존하였고, 실제에 적용하지는 못했습니다. Single-intersection에서 RL을 적용한 것을 확인해보니 전통적인 transportation method보다 더 나은 성능을 보이는 것을 확인하였고, 이를 scalability를 만족시키기 위해 multi-intersection에 적용하려고 이 논문을 작성하게 되었다고 합니다.

18

Scalability를 위해서 RL은 2가지로 사용할 수 있는데, 모든 intersection을 agent별로 분리해서 각자 학습시키는 방식과 centralize된 optimization 기반으로 multiple coordination을 달성할 수 잇는 방식이 있었습니다. 하지만, 전자는 scale-up은 쉬웠지만, 이웃 교차로와 협업을 하지않으므로 global optimization을 이룰 수 없었습니다. 후자의 경우, coordination은 기존의 multi-agent algorithm으로 달성할 수 있지만, scale-up이 될수록 joint action space로 인해 curse of dimensionality의 위험이 있었습니다. Curse of dimensionality는 판단해야하는 차원이 커질수록 local optimization될 확률이 높아지므로 섣불리 centralized algorithm을 사용할 때, scale-up을 못한다는 단점을 가지고 있다고 볼 수 있습니다. 이렇듯이 scalability와 coordination은 trade-off관계를 가지고 있다고 볼 수 있습니다.

19

이러한 단점을 보완하기 위해, Decentralized Approach를 사용하여 적용하려는 시도가 있었습니다. 하지만, 보통 average wait time과 delay등 주로 사용하는 transportation measurement를 reward함수로 사용하여 각 agent가 optimize되면서 global하게 optimize되는지에 대한 여부는 확인할 수 없었습니다. 그래서 논문에서는 3가지 기준점을 잡고 모델링을 통해 이 문제를 해결하려고 하였습니다. 각 intersection과 연관된 reward system이어야 하고, reward design이 direct coordination을 하기 적합해야 하며, RL이 불필요한 데이터를 가지면 안된다고 기준을 잡고 실험을 진행하였습니다.

20

논문에서는 아까 언급한 Scalability, Coordination, Data feasibility를 극복하기 위한 실험을 진행하였습니다. Simple feature와 real-data를 이용하여 infeasibility를 극복하고자 하였습니다. 또한, Superagent의 특징인 parameter를 공유하고, action을 분리하여 scalable하게 만들려고 하였습니다. 기존 연구에서 사용하던 FRAP 모델을 사용하여 학습 속도를 올리고, exploration space를 줄이고자 구성하였습니다. Pressure 개념을 reward에 도입하여 해당 global throughput을 최대화 시키고 pressure를 줄여 차량의 분포를 균일하게 만드는 것을 목표로 하였습니다.

21

FRAP model은 이 연구팀의 이전 연구에서 만든 모델로서, exploration experience를 획기적으로 줄이기 위한 아이디어였습니다. 4개 교차로에서 phase 종류가 8개라고 할 때, 동일 state에서 8가의 action을 학습해야 완벽한 학습이 되지만, 이를 flipping, rotating을 통해서 동일한 데이터를 가지고 한 번 학습할 때, 8회의 경험이 쌓이는 것이 주요 내용입니다. 이러한 모델링이 실험의 전체에 사용되므로 더 깊게 설명 드리겠습니다. 다른 intersection에도 이를 적용하기 위해서 두 가지 원리를 바탕으로 model을 만들었습니다. Competition principle과 invariance principle입니다. Competition principle은 larger traffic일수록 통행량 증가를 위한 green signal의 수요가 높다는 것이고, invariance principle은 rotating과 flipping을 통한 symmetric한 구성에서도 신호제어는 변함이 없어야 한다는 것입니다. FRAP model에서는 가장 높은 phase score를 predict해야 agent가 그것을 받아서 학습에 사용가능합니다.

22

FRAP은 다음의 3가지 순서로 진행됩니다. Phase demand modeling에서는 현재 phase와 차량의 수를 2개의 fully connected layers를 지나 phase demand로 연산됩니다. 이 때, phase에서는 oppose movement도 연산이 되며 이를 fully connected layers를 통해 나온 값들끼리 더합니다. 더한 값을 d라고 할 때, state 2개 fully connected layers끼리의 답을 구성한 후, competing 하고자 하는 phase demand와 concatenate를 시킵니다. 이후, 이러한 embedding은 8x7xL로 구성이 되는데 8개의 phase에서 나를 제외한 phase7개를 고르므로 다음과 같이 표시됩니다. 이를 1x1 filter 로 k번 CNN을 하는데, 그 이유는 비교하는 phase의 정보 안에 주변 intersection의 정보가 있기 때문에 size를 키우지 않았으며, filter는 20개로 고정하였습니다. Phase competition mask를 embedding하여 이를 K번 CNN 을 통과 시킨 후, element-wise multiplication을 진행하면 pairwise competition result가 나온다. Result의 row가 1개의 phase에 대한 나머지 모든 phase와의 competition score일 것이므로 이를 더해서 가장 높은 score의 phase를 선택하여 RL agent가 행동으로 선택하게 됩니다.

23

이 모델에서는 DQN을 기반으로 모델링을 진행하였습니다. DQN은 Q-learning 즉 value-based deep reinforcement learning이며, FRAP 모델을 기반으로 target network와 behavior(local) network를 두고 학습을 진행합니다. 상호 Q 값 비교를 통해 MSE를 기반으로 loss를 개선해 나가며, target network를 일정 주기마다 update를 하여 비교한다는 특징을 가지고 있습니다. 또한 update에 replay를 사용하여 이전에 있던 data를 기반으로 학습합니다. 하지만, 학습을 단순히 동시에 진행하면 이전 data와 현재 data간의 correlation이 높아 학습이 원활하게 진행되지 않으므로, replay로부터 batch size만큼 uniform random sampling을 진행하여 학습합니다.

24

DQN을 기반으로 Observation은 current phase p와 pressure demand를 기반으로 FRAP에 넣어 Q를 만듭니다. Action은 매 10초마다 1 time step을 가지고 진행하며, 8개의 candidate phases를 고르지만 모든 phase를 반드시 고르지는 않습니다. Reward는 Max Pressure control 처럼 Pressure가 높을수록 penalty를 부여하는 형식으로 reward함수가 구성되어 있습니다. 여기서 DQN을 super-agent라고 부르는 이유는 각 action을 agent가 각각 결정하지만, 모든 observation parameter를 공유하기 때문이며 이를 하나의 network로 구성할 수 있으므로 single agent의 한 종류인 super-agent라고 부릅니다. 즉, curse of dimensionality를 줄이기 위해 동시에 결정해야 하는 action space의 dimension을 줄이고, 각각 Q값을 갖기 때문에 action을 다르게 취하는 형태가 됩니다. ( 확률적으로 생각하면, 동시에 여러 개중에 하나를 선택하는 것보다 적은 개수에서 하나를 선택하되 다른 주머니에서 하나를 또 고르는 것이 정확한 선택을 할 수 있고 최적화 하기 쉽다)

25

그러므로 super-agent에서 설명한 것처럼 parameter sharing은 모든 observation data에 대해서 실시하고 각 intersection은 모든 intersection의 data를 갖는다. 실험에서 setting은 4x4 data를 기반으로 시뮬레이션을 진행했습니다. 각 phase 이후에 3초의 노란 신호와 2초의 all red를 부여하여 안전성을 높였습니다. Bi-directional과 dynamic flow한 속도를 더하기 위해서 첨두 시간 여부에 따라서 Flat과 Peak를 구분하였으며, 속도 분산을 0.6과 0.3으로 각각 구분하여 부여하였다. 실제 데이터와 유사하게 하기 위해서 turning ratio는 0.6일 때 직진, 0.3일 때는 우회전, 0.1은 좌회전을 진행하게 구성하였습니다.

26

이후 실제 scale로 확대하여 real data인 Manhattan의 taxi driver data를 활용하여 진행하였습니다. 대신에 scale을 맞추기 위해 scale factor를 곱하여 실행하였으며, 총 2510개의 traffic system이 존재하는 상태에서 진행하였습니다. 여기서 성능을 판가름하는 기준은 Travel time과 Throughput을 기준으로 삼았습니다. Travel time은 모든 차량의 평균 이동 시간을 의미하며, signal control method의 performance를 확인하기 위해 사용하였고, 낮을수록 높은 성능을 의미합니다. 그리고 Throughput은 trip이 완료된 수를 의미하며, 높다면 control 전략이 괜찮다는 것을 의미하므로 높을수록 좋습니다.

27

세팅에 대한 실험 결과는 다음과 같습니다. MPLight