Offline Vehicle Routing Problem with Online Bookings: A Novel Problem Formulation with Applications to Paratransit

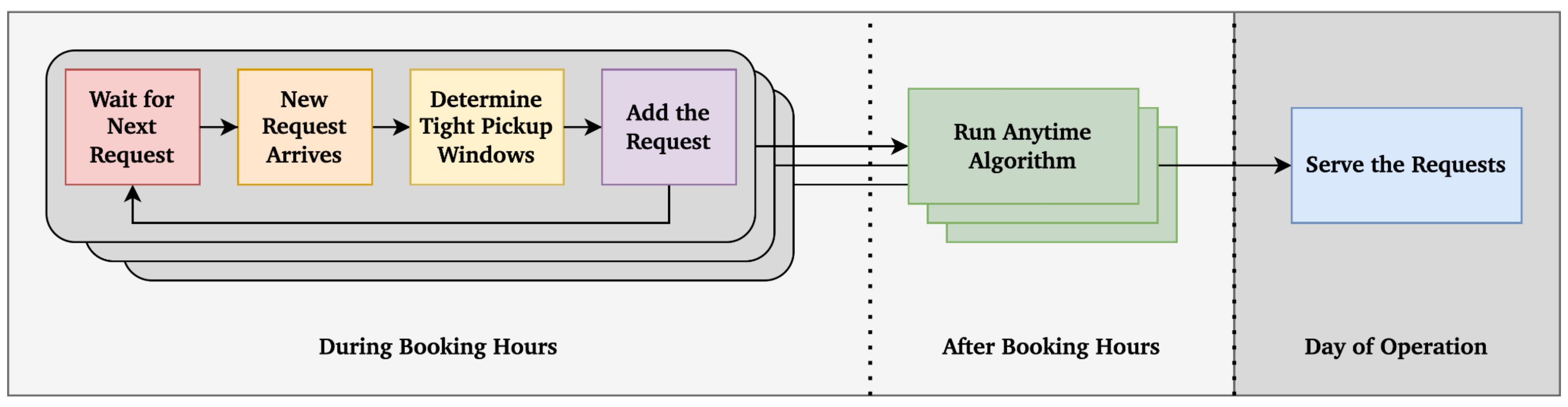
Amutheezan Sivagnanam[1], Salah Uddin Kadir[1],
Ayan Mukhopadhyay[2], Philip Pugliese[3], Abhishek Dubey[2],
Samitha Samaranayake[4], Aron Laszka[5]

[1]University of Houston, [2]Vanderbilt University, [3]Chattanooga Area Regional Transport Agency, [4]Cornell University, [5]Pennsylvania State University

This material is based upon work sponsored by the National Science Foundation under Grant CNS-1952011 and by the Department of Energy under Award DE-EE0009212

Motivation

We focus on the problem of ***online day ahead booking,*** *where passengers* ***provide flexible broad windows (e.g., 3 hours)*** *that need to be* ***tightened (e.g., 30 minutes)*** *in real-time.*
Applications*: Paratransit, Food Delivery, etc.*



Existing Vehicle Routing Problem (VRP)

Offline VRP	Online VRP
All the requests are known	Request arrives in real time
Allow large running time	Make quicker decision

Problem Statement

How to select tight pickup windows at the time of booking, based on both requests booked so far as well as expectation of future requests?

GOAL: Learning optimal policy that determines tight pickup window during online decision making, that minimizes total travel time and number of vehicle routes required

DATA: Trip data that contains details of pickup and dropoff locations, and broad pickup windows

We partnered with **Chattanooga Area Regional Transportation Authority (CARTA)** and obtained the real-world paratransit trip data

Model and Problem Formulation

Online Booking Problem

Inputs:
Existing trips $(\langle \mathrm{T}_1, \ldots, \mathrm{T}_{i-1}\rangle, \langle \mathrm{w}_1, \ldots, \mathrm{w}_{i-1}\rangle)$
Current vehicle routes $(\mathrm{R}^{(i-1)})$
Incoming request (T_i, W_i)

Outputs:
Tight pickup window (w_i)
Updated vehicle routes $(\hat{\mathrm{R}}^{(i)})$

Constraints:
Tight pickup windows should be within the broad pickup windows

$$W_i^{\text{start}} \leq w_i^{\text{start}} \leq w_i^{\text{end}} \leq W_i^{\text{end}}$$

Offline Vehicle Routing Problem

Inputs:
Trips $(\langle \mathrm{T}_1, \ldots, \mathrm{T}_i\rangle, \langle \mathrm{w}_1, \ldots, \mathrm{w}_i\rangle)$
Candidate vehicle routes $(\hat{\mathrm{R}}^{(i)})$

Outputs:
Optimized vehicle routes $(\mathrm{R}^{(i)})$

Constraints:
Must serve all the trip requests
Pickup and dropoff must ensure time window and capacity constraints
Duration of vehicle route must be less than D^{maxroute}

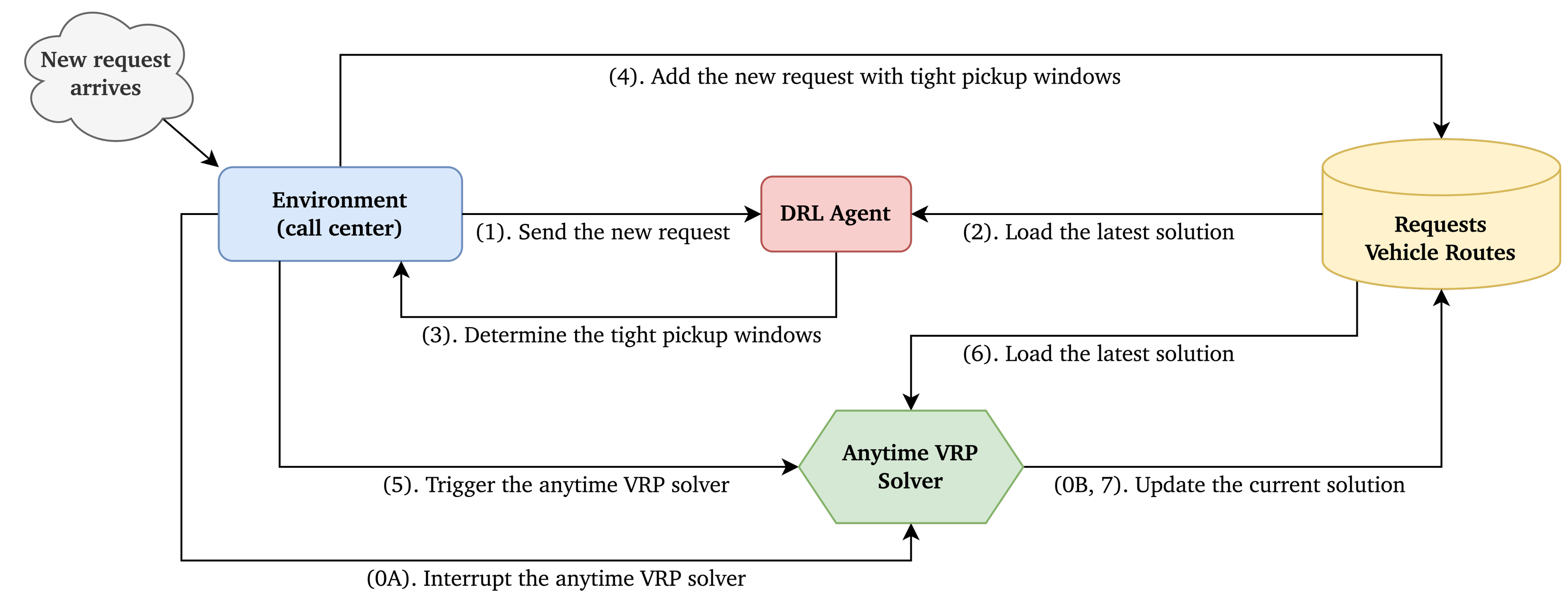
Objective:
Select tight windows such that ***the optimal cost*** *of resulting offline VRP is* ***minimized***

$$argmin_{\mu}\mathbb{E}_{(\mathbf{T},\mathbf{W})\sim\mathscr{D}}\left[VRP^*(\mathbf{T},\mathbf{w})\,\middle|\,(w_i,\hat{\mathbf{R}}^{(i)})=\mu\left(\langle T_1,\ldots,T_{i-1}\rangle,\langle \mathbf{w}_1,\ldots,\mathbf{w}_{i-1}\rangle,W_i,\mathbf{R}^{(i-1)}\right),\ \mathbf{R}^{(i)}=\alpha\left(\langle T_1,\ldots,T_i\rangle,\langle \mathbf{w}_1,\ldots,\mathbf{w}_i\rangle,\hat{\mathbf{R}}^{(i)}\right)\right]$$

Challenges:
We need to determine the tight pickup windows during online booking **within a limited time frame (i.e., 1 - 2 seconds)**
- considering **the large number of trips** booked so far
- minimizing **the number of vehicle routes** required

Solution Approach

DRL Agent	Anytime VRP Solver
determines the tight pickup windows whenever a new request arrives	perform the offline VRP optimization based on the requests arrived so far



State-Action: State space is ***enormous*** and ***complex***, to overcome the complexity we map the ***state-action*** into following feature vectors
- Busyness of the area at given time of the day
- Additional travel time/distance
- Tightness of requests within the vehicle route
- Expectation of future requests ($|\mathrm{T}| - i$)

Reward Shaping: Rewards from environment are delayed (e.g., by 140 to 200 steps); instead, we compute reward as the difference in cost (i.e., total travel time and vehicle routes required) between before and after tightening pickup windows

$$\tilde{c}_i = VRP^*(\mathbf{T}, \langle w_1, \ldots, w_{i-1}, w_i, W_{i+1}, \ldots, W_{|\mathbf{T}|}\rangle) - VRP^*(\mathbf{T}, \langle w_1, \ldots, w_{i-1}, W_i, W_{i+1}, \ldots, W_{|\mathbf{T}|}\rangle)$$

Data Collection and Experimentation Setup

Collect historical paratransit trip data from **Chattanooga**, **Tennessee**.
6 months of data, with an average of **140** trips per day

Obtain the travel time and travel distance data using **OSMNx** and **NetworkX**

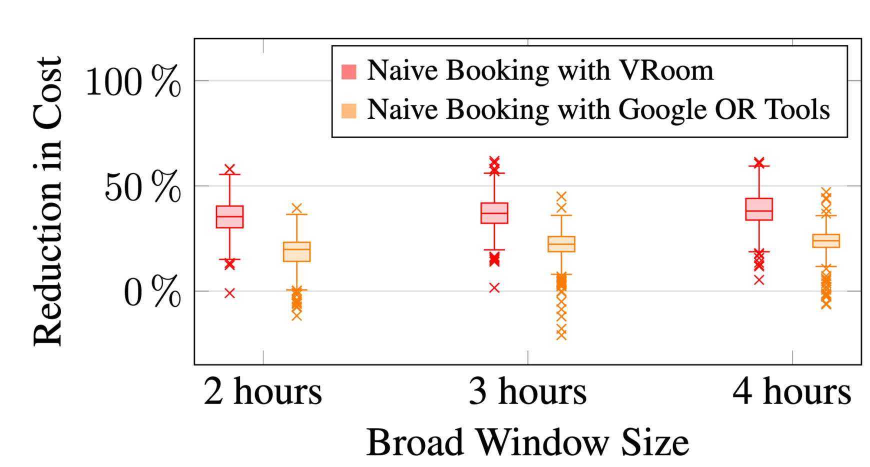
Implement the entire framework in Python **3.8**
(used **TensorFlow** for Neural Networks)

Run the experiments in **Intel Xeon E5-2680 28-core CPU** and **128 GB of RAM**

Vehicle capacity: **9** passengers, Maximum route duration: **10** hours,
Tight windows: **30** minutes, Broad windows: **3** hours

Evaluation

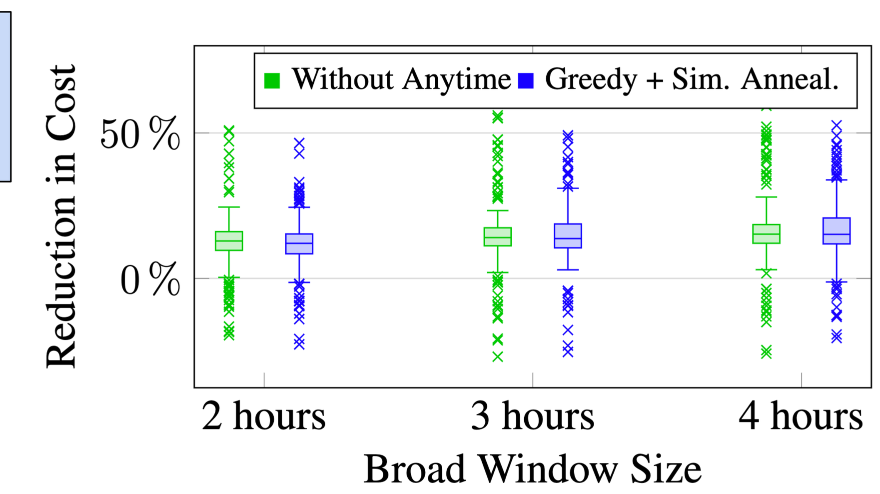
The trained DRL agent is able to **reduce the cost** by selecting tight pickup windows during online booking **compared to naive selection** (i.e., always choose the middle of broad windows)



Training Duration: 1 - 2 days per episode

Decision time: 0.25 - 2 seconds

The DRL agent **supported by the anytime algorithm performed better than** a DRL agent **without anytime algorithm**



Conclusion

- The novel problem formulation for online booking by **minimizing the total travel time** and **total vehicle routes**
- **Significant reduction** in costs due to selecting tight pickup time windows using trained DRL agent instead of a naive selection
- The trained DRL agent determines the tight pickup windows **within a fraction of seconds**
- The DRL agent trained running the anytime algorithm between requests **performs better** than the agent trained without

Contact:
Amutheezan Sivagnanam (amutheezan.com)
Aron Laszka (aronlaszka.com)

smarttransit.ai