Supplementary Materials to 'Learning to Optimize Industrial-scale Dynamic Pickup and Delivery Problems'

Xijun Li, Jiawen Lu and Weilin Luo

January 21, 2021

1 Parameter settings

The recommended parameters and hyperparameters of different DRL methods are listed in below tables.

Table 1: Hyperparameters used in DRL algorithms

Hyperparameter	DGN & ST-DDGN	DQN & AC
# attention layers	2	-
# attention heads	8	-
NE (# vehicles < 50)	3	-
NE (# vehicles ≥ 50)	10	-
# neurons of layers in MLP (initial)	(512, 128)	(1024,256,1)
# neurons of layers in MLP (final)	(64,1)	-
activation	ReLu	
initializer	random normal	

Table 2: Parameter setting in policy training

Parameter	Recommended value
discount factor (γ)	0.95
instant reward factor (α)	0.01
cost of using a vehicle (μ)	300
operation cost per kilometers (δ)	0.8
updating period of target network ($\mathcal{T}$)	36
learning rate	10^{-4}
batch size	64
# training episodes ($maxEpisode$)	700
capacity of replay memory	10^5
greedy factor (ϵ)	0.95
optimizer	Adam
gradient clipping parameter (clipnorm)	1

2 Mathematical Model of PDP

The static version of DPDP can be defined on a complete directed graph $G = (N, A)$, where $N = W \cup P \cup D$ is the set of nodes and $A = \{(i, j)|i, j \in N\}$ is the set of arcs. $W = \{w_i|i = 1, ..., m\}$ is the set of all depots.

For each depot, K_{w_i} denotes the set of vehicles starting from depot $w_i \in W$, of which each vehicle starts from depot w_i and must return to the depot at the end of its path. $P = \{1, 2, ..., n\}$ and $D = \{n+1, n+2, ..., 2n\}$ represents the sets of pickup and delivery nodes respectively. An arc (i, j) denotes the directed connection from node i to node j. Here each delivery order o_i is associated with an earliest time t_e^i and a latest time t_l^i for vehicle to serve, a pickup node $i \in P$, a delivery node $(n+i) \in D$ and amount of cargoes to be delivered q_i where $q_i > 0$ if $i \in P$ and $q_i = -q_{i-n}$ if $i \in D$. Let s_i denote the service (loading or unloading) duration at node i, with $s_i > 0$ if $i \in P \cup D$ and $s_i = 0$ if $i \in W$. Each node $i \in P \cup D$ is associated with a time window $[e_i, l_i]$, where e_i and l_i respectively represent the earliest and latest time at which the service for node i must begin. Intuitively, both e_i and l_i are related to t_e^i and t_l^i, where $t_e^i = e_i \leq l_i = t_l^i$ always holds for node $i \in P \cup D$. A fleet of K homogeneous vehicles with maximum loading capacity Q is available. A non-negative transportation distance distance $d_{i,j}$ and a non-negative travel time $t_{i,j}$ are associated with arc $(i, j) \in A$. To minimize the total transportation cost, the transportation distance of each arc leaving from depots, i.e., arc (w, i) such that $w \in W, i \in P$, is set to be large [1]. M is a very large constant. Besides, the triangle inequality is assumed to be respected for both travel distances and times. The decision variables are introduced as follows. For each arc (i, j) and vehicle k, x_{ij}^k denotes a binary variable equal to 1 if vehicle k travels from node i to node j otherwise 0. For each node $i \in N$ and each vehicle $k \in K$, let T_i^k represent the time vehicle k begins service at node i, and let Q_i^k be the load of vehicle k when leaving from node i. The whole mathematical model is given below:

Objective

$$min\ F(x) = \sum_{k \in K} \sum_{i \in N} \sum_{j \in N} d_{ij} \times x_{ij}^k \tag{1}$$

s.t.

$$\sum_{k \in K} \sum_{j \in P \cup D, j \neq i} x_{ij}^k = 1 \quad \forall i \in P \tag{2}$$

$$\sum_{k \in K} \sum_{j \in P \cup D, j \neq i} x_{ji}^k = 1 \quad \forall i \in D \tag{3}$$

$$\sum_{j \in N, j \neq i} x_{ji}^k - \sum_{j \in N, j \neq i} x_{ij}^k = 0 \quad \forall i \in N, \forall k \in K \tag{4}$$

$$\sum_{i \in N, i \neq w} x_{w,i}^k = 0 \quad \forall k \notin K_w \tag{5}$$

$$\sum_{j \in P \cup D, j \neq i} x_{i,j}^k - \sum_{j \in P \cup D, j \neq i} x_{j,n+i}^k = 0 \quad \forall i \in P,\ \forall k \in K \tag{6}$$

$$\sum_{i \in V, i \neq w} x_{w,i}^k \leq 1 \quad \forall k \in K_w \tag{7}$$

$$\sum_{k \in K_w} \sum_{i \in P} x_{w,i}^k \leq |K_w| \quad \forall w \in W \tag{8}$$

$$\sum_{k \in K_w} \sum_{i \in D} x_{i,w}^k \leq |K_w| \quad \forall w \in W \tag{9}$$

$$\sum_{k \in K_w} \sum_{i \in P} x_{w,i}^k = \sum_{k \in K_w} \sum_{i \in D} x_{i,w}^k \quad \forall w \in W \tag{10}$$

$$Q_j^k \geq Q_i^k + q_j - Q(1 - x_{ij}^k) \quad \forall i \in N \quad i \neq j, \forall j \in P \cup D, \forall k \in K \tag{11}$$

$$Q_{n+i}^k = Q_i^k - q_i \quad \forall i \in P, \forall k \in K \tag{12}$$

$$0 \leq Q_i^k \leq Q \quad \forall i \in N, \forall k \in K \tag{13}$$

$$T_j^k \geq T_i^k + t_{ij} + s_i - M(1 - x_{ij}^k) \quad \forall i \in N \quad i \neq j, \forall j \in P \cup D, \forall k \in K \tag{14}$$

$$T_{n+i}^k \geq T_i^k + t_{i,n+i} + s_i \quad \forall i \in P, \forall k \in K \tag{15}$$

$$e_i \leq T_i^k \leq l_i \quad \forall i \in N, \forall k \in K \tag{16}$$

$$x_{ij}^k \in \{1, 0\} \quad \forall (i,j) \in A \quad \forall k \in K \tag{17}$$

Equation (1) is the objective function where the total travel distance is minimized. The reason why we only optimize the total travel distance is that the travel cost is directly proportional to travel distance. Constraints (2) and (3) ensure that each node is served exactly once. Constraint (4) is the flow conservation. Constraint (5) makes sure that vehicles cannot depart from those depots to which the vehicles do not belong. Constraint (6) ensures that for every delivery order, its pickup and delivery nodes are served by the same vehicle. For each depot w, constraints (7) - (10) ensure that the number of vehicle leaving from depot w cannot exceed the maximum number of vehicle depot w has and that each vehicle leaving from depot w must return to w at the end of its path. Constraints (11) - (13) calculates the load variables according to arc used in the solution and make sure that the maximum loading capacity of vehicle is respected. Analogously, constraints (14) and (16) calculates the time variables and ensure that time windows for service are respected at each node i. Constraint (14) ensures that for each delivery order, its pickup node is served before corresponding delivery node. Besides, the Last In, First Out (LIFO) rule is achieved in constraints (11) and (14).

3 Comparisons between KL and JS divergence

There are two reasons why we employ Jensen-Shannon (JS) divergence rather than KL-divergence: 1) the JS divergence is symmetric in calculation, i.e., $D_{JS}(p||q) = D_{JS}(q||p)$, and 2) that the experimental results show that JS divergence performs slightly better than KL divergence in our solution. As shown in the Fig. 1 and Fig. 2, our solution with JS divergence could get lower number of used vehicle and total cost on most of the testing instances.

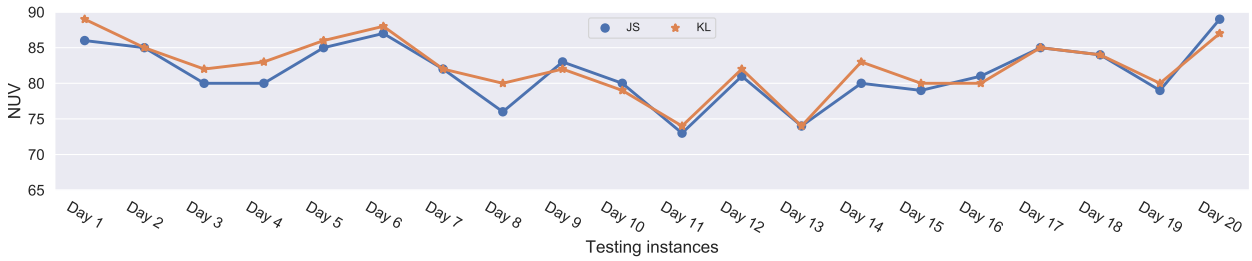

Figure 1: Comparison between JS and KL divergence on NUV.

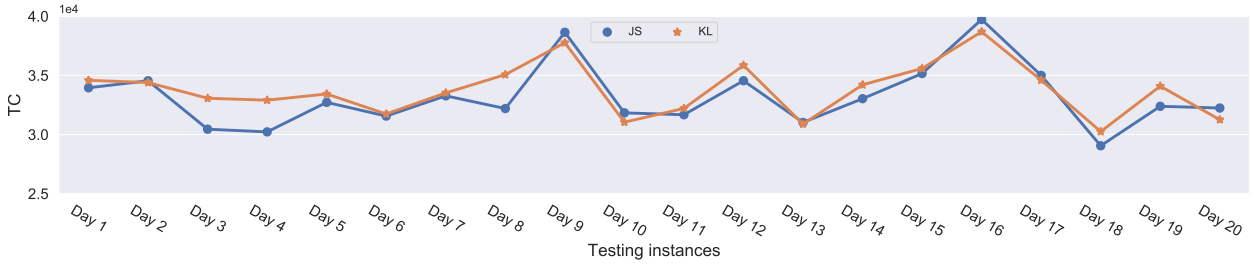

Figure 2: Comparison between JS and KL divergence on TC.

References

[1] Marilène Cherkesly, Guy Desaulniers, and Gilbert Laporte. "Branch-price-and-cut algorithms for the pickup and delivery problem with time windows and last-in-first-out loading". In: *Transportation Science* 49.4 (2014), pp. 752–766.