A Modular Framework for the Freight Management Problem Encountered by Third-Party Logistics Providers

Presented by : Amin Abbasi Pooya

School of Business, University of Kansas

THE UNIVERSITY OF KANSAS

Brief Overview

1. **Introduction to Logistics**

2. **The General Framework for the 3PL Freight Management Problem**

3. **One Instantiation of the Problem**

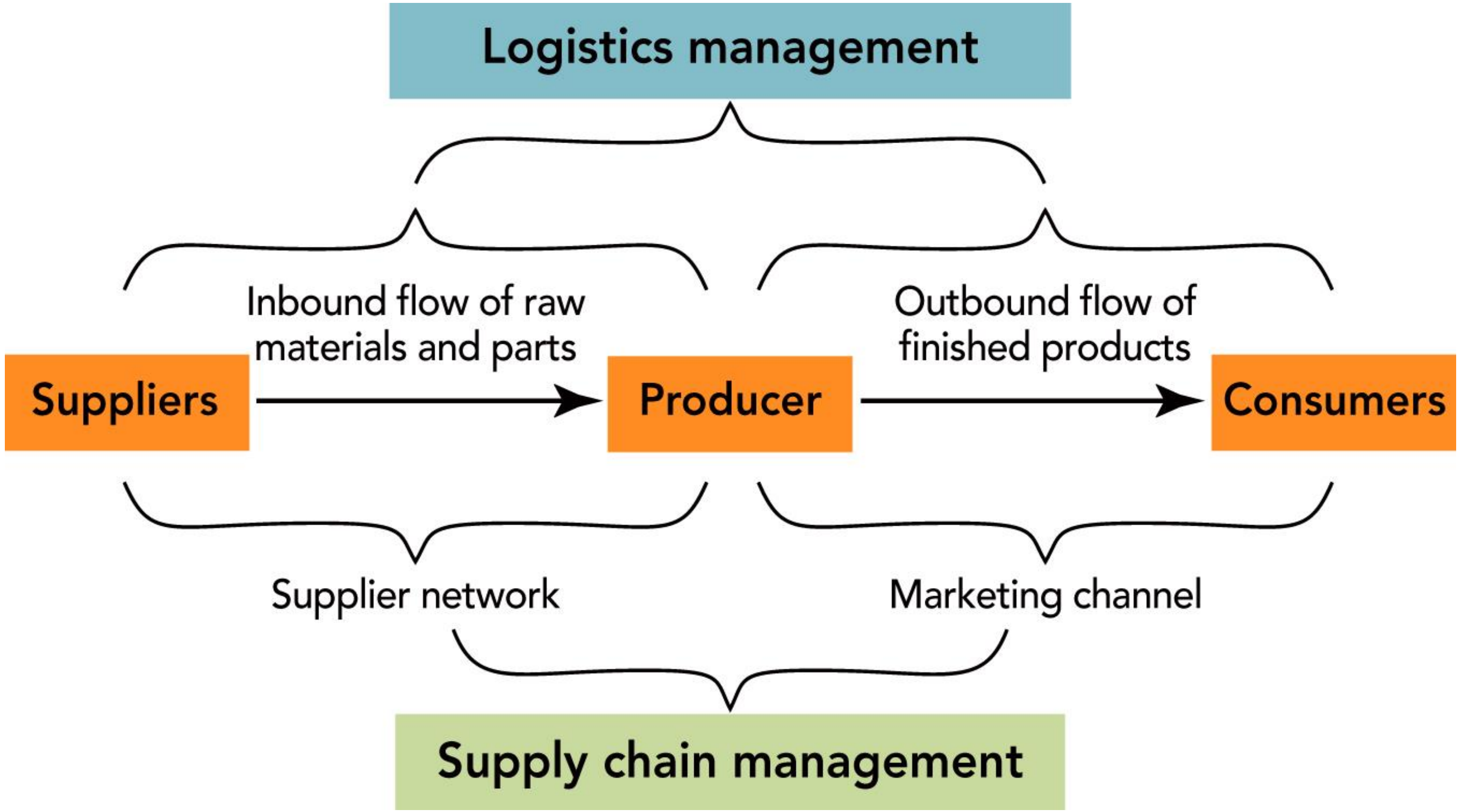
4. **Reinforcement Learning**
    1. Q-Learning

5. **Results**

6. **Conclusions and Future Research**

Introduction to Logistics

Logistics



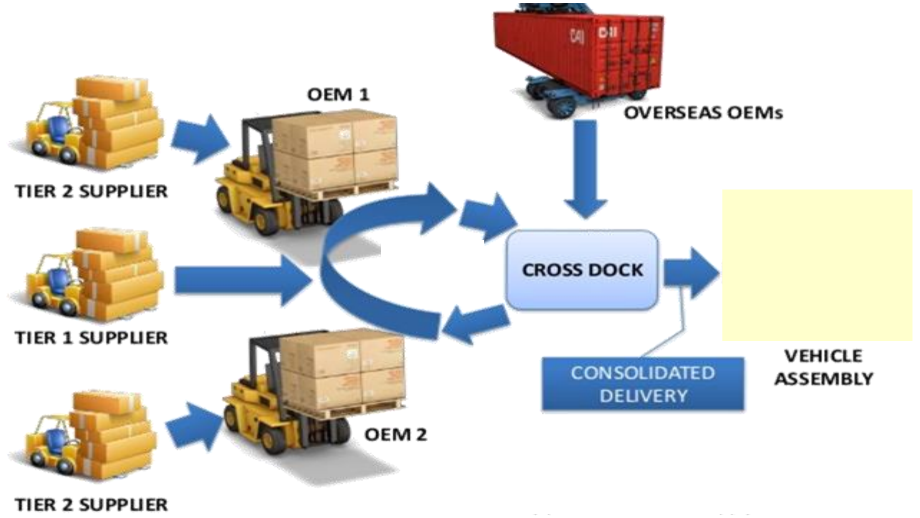
Marketing (2015), Kerin, Hartley, Rudelius

Third-Party Logistics (3PL) Providers

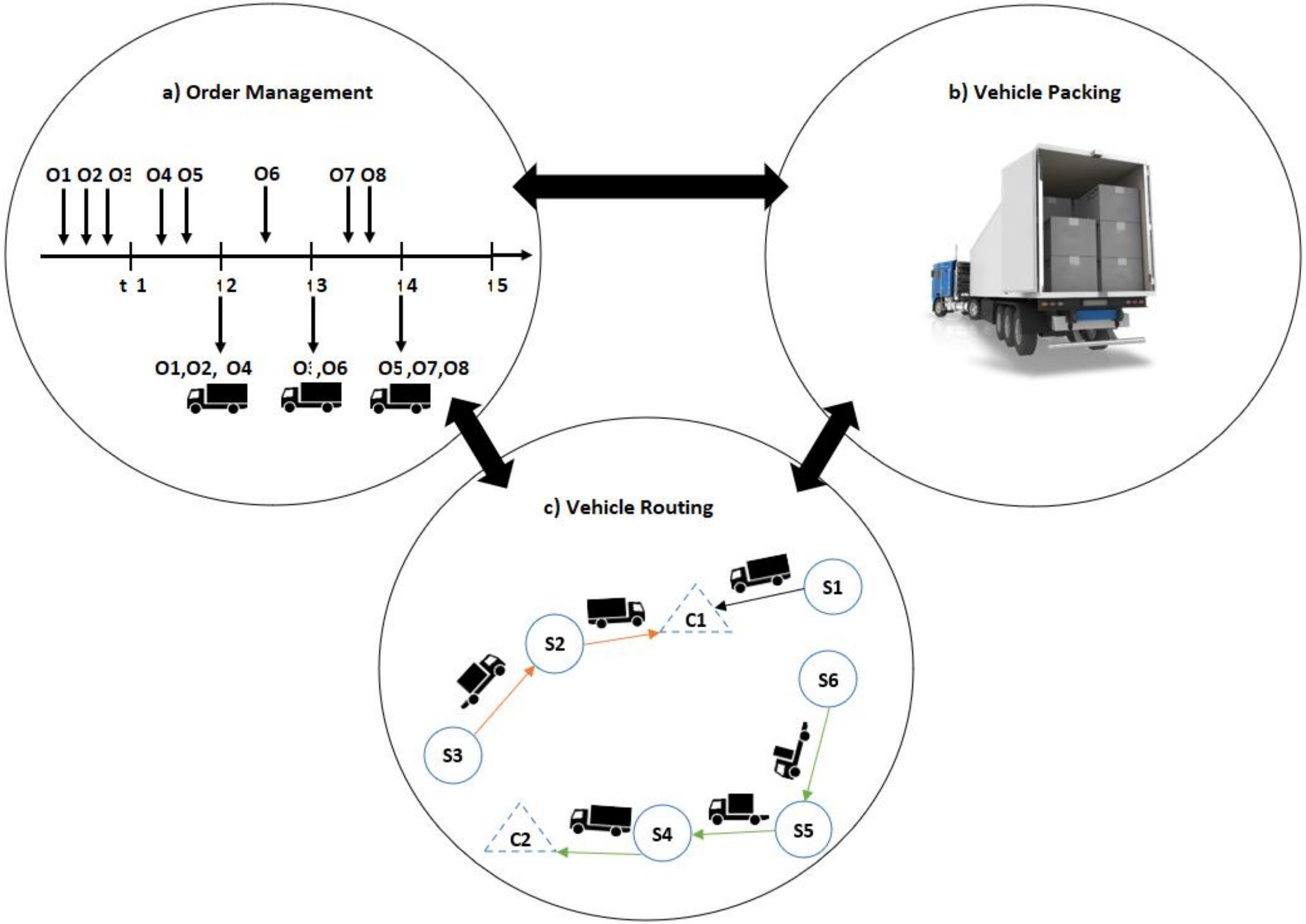
- Employing 3PL companies : focus on the production of goods.

- A 3PL provider : an external entity that is responsible for management, control, and delivery of logistics activities for a shipper (Hertz, Alfredsson, 2003).

- Example of Toyota’s inbound logistics

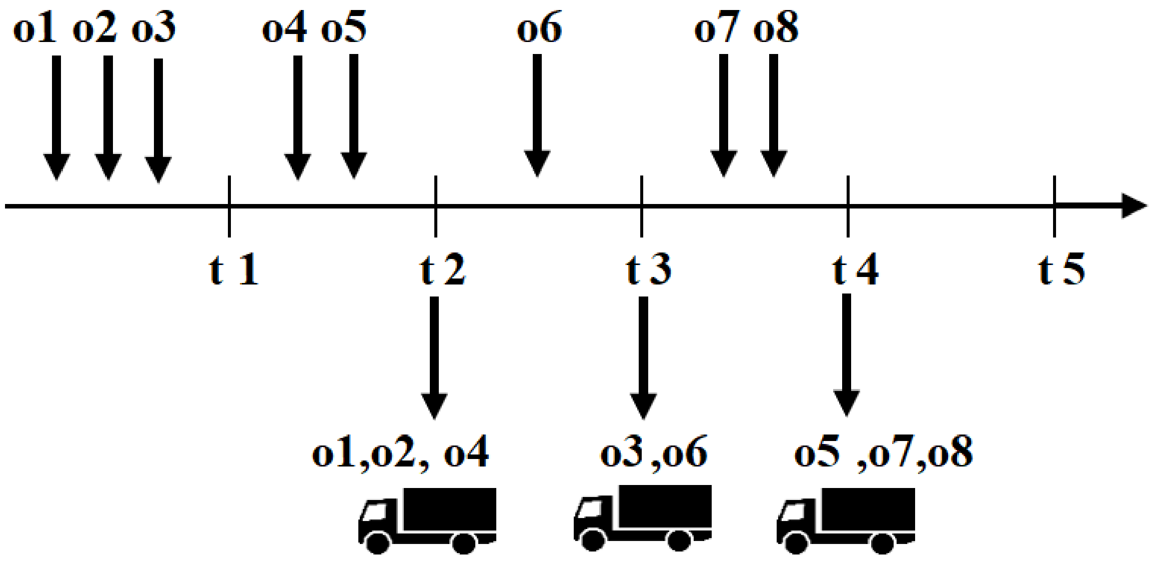


Three Main Activities of a 3PL



Order Management

- Consolidation strategies (vehicle, terminal, inventory)
- Load splitting
- Holding costs associated with late pickup or early delivery
- Order/supplier compatibility



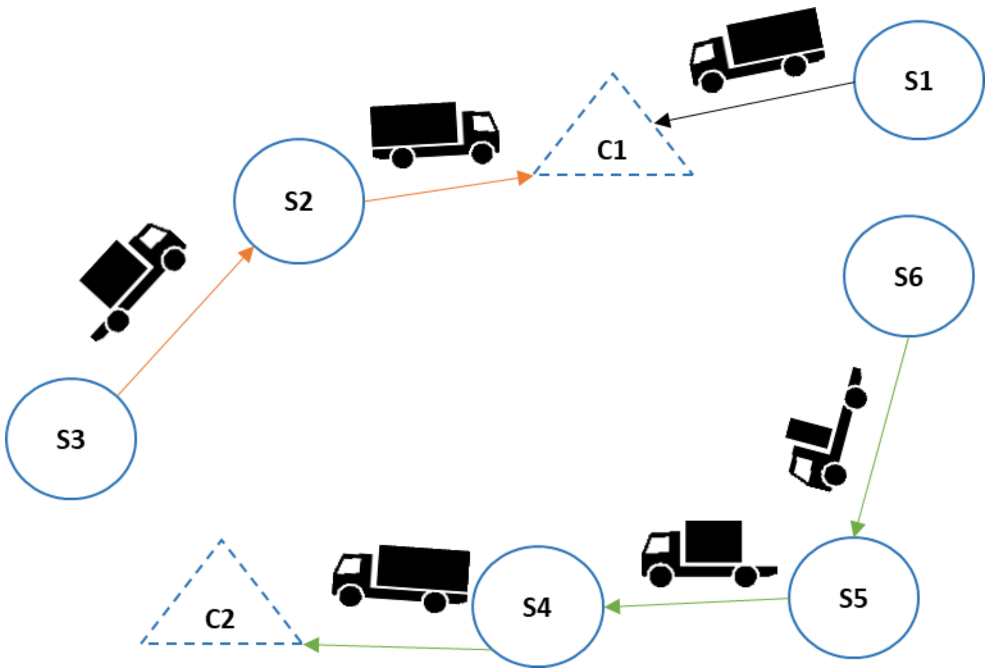
Vehicle Packing

- Volume capacity
- Stackability
- Stability
- Loading/unloading considerations

https://www.freightcenter.com/services/ltl-freight

Vehicle Routing

- Deterministic/stochastic travel time
- Service time at each node
- Time window
- Speed limits on roads
- Maximum number of stops per route



3PL FM Problem Formulation

Notation

- $C = \{1,2, \dots, n_c\}$: the set of n_c customers
- $S = \{1,2, \dots, n_s\}$: the set of n_s suppliers
- $P = \{1,2, \dots, n_p\}$: the set of n_p products
- $V = \{1,2, \dots, n_v\}$: the set of n_v vehicles
- Ψ : the set of considerations
- Φ : a set containing three arbitrary data structures containing relevant order assignment information (Φ_A), packing information (Φ_P), and routing information (Φ_R)

Problem

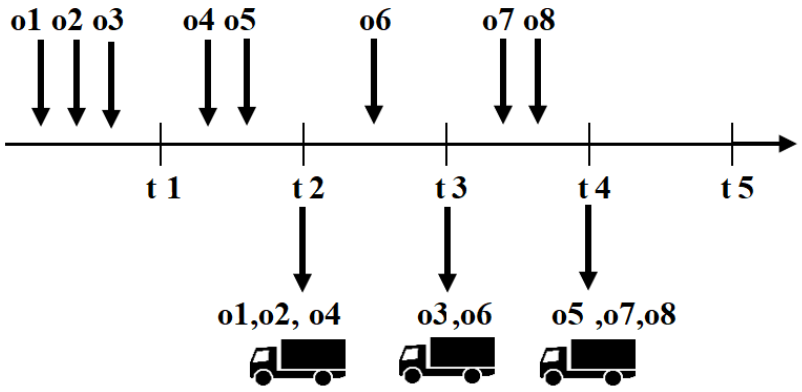
Customer i ($i \in C$) places an order $o_{ijp}^{(t)}$ of product p ($p \in P$) from the supplier j ($j \in S$) at time $t \in T$, where $T = \{1,2, \dots, n_t\}$. The order must be delivered to customer i by the due date $t'_{o_{ijp}^{(t)}}$ (denoted by t_o).

3PL FM Problem Formulation

$$\boldsymbol{\Phi}_{A}^{(\mathbf{t})} \leftarrow \boldsymbol{ASSIGN}_{\sigma}^{\omega}\,(\boldsymbol{O}^{(t)}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi})$$

an arbitrary procedure that assigns orders in $O^{(t)}$ to vehicles under a particular consolidation strategy σ ($\sigma \in \Sigma$) using the solution strategy ω ($\omega \in \Omega$)

- Σ : consolidation strategy (vehicle, inventory, or terminal consolidation)
- Ω : solution strategies such as dynamic programming, heuristic/metaheuristic methods, and machine learning methods
- Ψ : set of considerations (weight capacity, load splitting, holding costs associated with late pickup or early delivery, order/supplier compatibility, and repackaging considerations)



3PL FM Problem Formulation

$$\boldsymbol{\Phi}_{\boldsymbol{P}}^{(\mathbf{t})} \leftarrow \boldsymbol{PACK}^{\boldsymbol{\omega}}\left(\boldsymbol{\Phi}_{\mathbf{A}}^{(\boldsymbol{t})}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi}\right)$$

an arbitrary procedure for arrangement of packages in vehicles using the solution strategy ω ($\omega \in \Omega$)

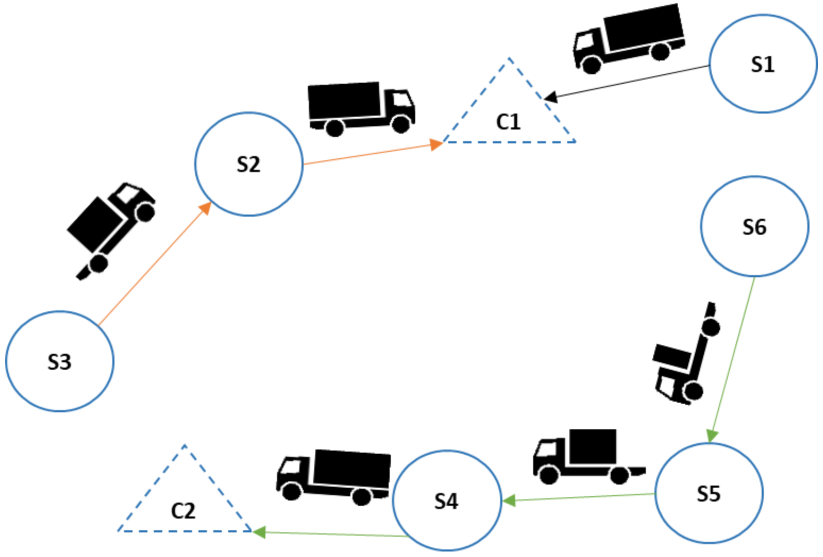
- Ω : solution strategies such as dynamic programming, heuristic/metaheuristic methods, and machine learning methods
- Ψ : set of considerations (volume capacity, stackability, stability, loading/unloading considerations)

3PL FM Problem Formulation

$$\boldsymbol{\Phi}_R^{(\mathbf{t})} \leftarrow \boldsymbol{ROUTE}^{\omega}\,(\boldsymbol{\Phi}_{\mathbf{A}}^{(t)}, \boldsymbol{\Phi}_P^{(t)}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi})$$

an arbitrary procedure that creates a set of routes that ensure each vehicle visits the origin and destination of the orders it carries.

- Ω : solution strategies such as dynamic programming, heuristic/metaheuristic methods, and machine learning methods
- Ψ : set of considerations (deterministic/stochastic travel time, service time at each node, time window, speed limits on edges, and number of stops per route)



General Algorithm

Algorithm 1 Algorithm for solving the 3PL freight management problem

Input: $C, S, P, V, T, \sigma, \omega, \Psi$

Initialize $O^{(t=0)}, \Phi^{(t=0)}$

for $t = 1, \ldots, T$ **do**

$\quad O^{(t)} \leftarrow INCOMING^{(t)} \bigcup O^{(t-1)}$

$\quad \Phi_A^{(t)} \leftarrow ASSIGN_\sigma^\omega(O^{(t)}, \Phi^{(t-1)}, \Psi)$

$\quad O^{(t)} \leftarrow O^{(t)} \setminus Orders(\Phi_A^{(t)})$

$\quad \Phi_P^{(t)} \leftarrow PACK^\omega(\Phi_A^{(t)}, \Phi^{(t-1)}, \Psi)$

$\quad \Phi_R^{(t)} \leftarrow ROUTE^\omega(\Phi_A^{(t)}, \Phi_P^{(t)}, \Phi^{(t-1)}, \Psi)$

$\quad \Phi^{(t)} \leftarrow \{\Phi_A^{(t)}, \Phi_P^{(t)}, \Phi_R^{(t)}\}$

$\quad COST^{(t)} \leftarrow f(\Phi^{(t)})$

end for

The 3PL Freight Management Problem with Fixed Routing Schedule and 1-Dimensional Packing

One Instantiation

$$\boldsymbol{ASSIGN_{\sigma}^{\omega}}\,(\boldsymbol{O}^{(t)}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi})$$

$\boldsymbol{\Psi}$: Weight capacity, demand distribution (split-load not allowed)

$\boldsymbol{\omega}$: Q-Learning

$\boldsymbol{\sigma}$: Vehicle consolidation

- Direct vs. consolidated shipment

One Instantiation

$$\boldsymbol{PACK^{\omega}}\ (\boldsymbol{\Phi}_{\mathbf{A}}^{(t)}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi})$$

$\boldsymbol{\Psi}$ **:** One-dimensional volume constraints

$\boldsymbol{\omega}$ **:** Q-Learning

One-dimensional Packing

Product	Package Dimensions	Volume
1	l_1, w_1, h_1	$l_1 * w_1 * h_1$
2	l_2, w_2, h_2	$l_2 * w_2 * h_2$
…	…	…
$\boldsymbol{n_p}$	l_p, w_p, h_p	$l_p * w_p * h_p$

One Instantiation

$$ROUTE^{\omega}\left(\boldsymbol{\Phi}_{\mathbf{A}}^{(t)}, \boldsymbol{\Phi}_{P}^{(t)}, \boldsymbol{\Phi}^{(\mathbf{t-1})}, \boldsymbol{\Psi}\right)$$

$\boldsymbol{\Psi}$: Weight capacity, delivery due date (t_o)

$\boldsymbol{\omega}$: Fixed routing schedule

Fixed Routing Schedule

Vehicle	Route	Available day
1	$s1 \rightarrow s2 \rightarrow c3 \rightarrow c5$	2
2	$s3 \rightarrow s5 \rightarrow s4 \rightarrow c1 \rightarrow c2 \rightarrow c4$	1
…	…	…
n_v	$s3 \rightarrow s1 \rightarrow s2 \rightarrow c1 \rightarrow c3 \rightarrow c5$	4

Direct Shipment Vehicle

$n_v + 1$	All suppliers and customers	All days

Some Details

- Arrival time and due date of each order (t_o)

- Customer demands follow a distribution : Generated and discretized to a finite set of numbers (quantization), called levels of demand (A three-level example would be High, Medium, and Low)

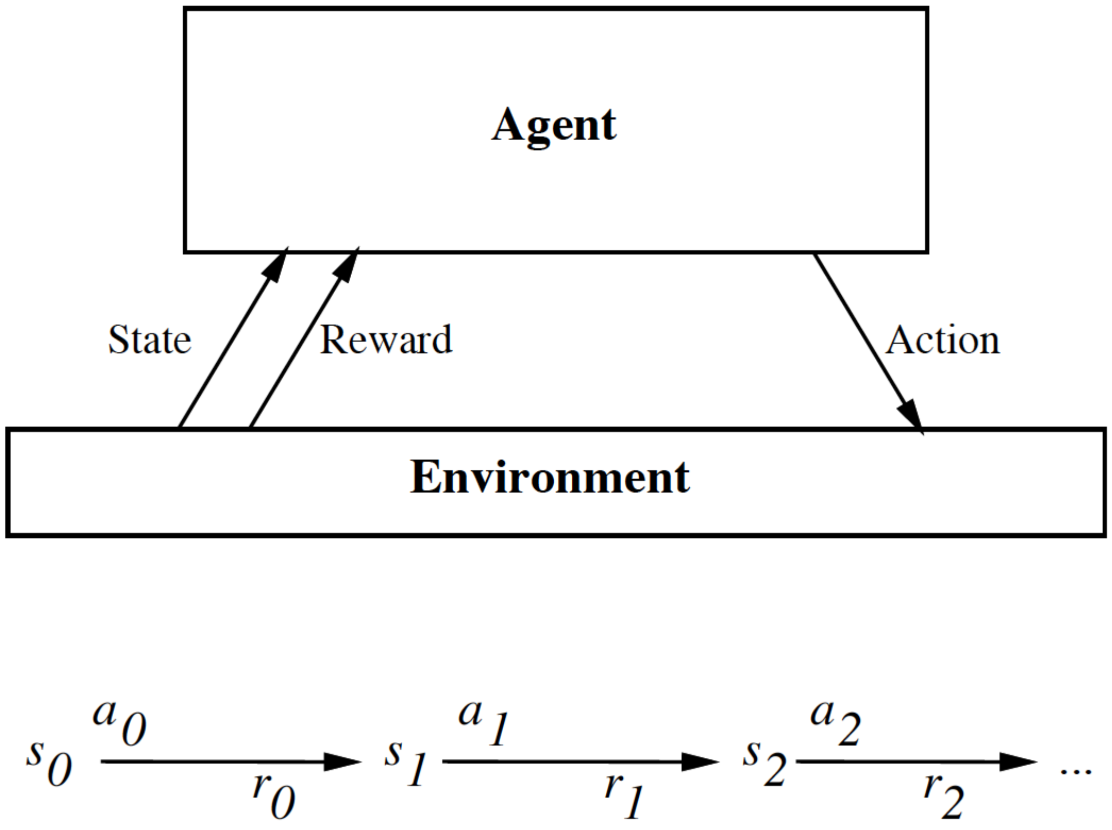
- There are vehicles of the same capacity (volume and weight) to carry the orders, and orders have different volumes and weights.

Reinforcement Learning

Reinforcement Learning

- (State,Action,Transition Func.,Reward)

- Goal?
  - Finding the optimal policy (mapping from state to action)



Reinforcement Learning

- Limited knowledge of the environment
  - Can only act in the world and observe states and reward
- Other factors
  - Actions have non-deterministic effects (which are initially unknown)
  - Rewards / punishments are infrequent
    - Often at the end of long sequences of actions
    - World is large and complex
- Nevertheless, agent must decide what actions to take

Q-Learning

The general form of problem

- $Max\, r = R^{(t)}$

$$s.t.\ \mathbb{G}\left(S^{(t)}\right) \leq 0$$

where $R^{(t)}$ is the reward function at time t, $\mathbb{G}\left(S^{(t)}\right)$ is the set of constraints corresponding to the considerations in the set Ψ.

Q-Learning

State

- The combination of customer i $(i \in C)$, product p $(p \in P)$, supplier j $(j \in C)$, and level of demand d $(d \in \{1,2,\dots,n_d\})$ at each time period t $(t \in T)$, i.e.
  - $S_{i,p,j,d}^{(t)} = (ipjd)^{(t)}$

- Assuming there are $n_{\{st\}}$ possible combinations of these four components (customer, product, supplier, demand level $(cpsd)$), and define
  - $S_n^{(t)} = (cpsd_n)^{(t)} \qquad (n = 1,\dots,n_{st})$

Action

- The vehicle $v \in V$ a $cpsd$ is assigned to.

- Action set (A) is $A = V \cup \{n_v + 1\}$ where $n_v + 1$ is an extra vehicle is added to the set of vehicles to represent a direct shipment vehicle.

Q-Learning

Q-Table

- Q values: expected cumulative rewards
- $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma \max_{a} Q(S',a) - Q(S,A)]$
- Discount factor γ $(0 < \gamma < 1)$: the weight of the future reward
- Learning rate α $(0 < \alpha < 1)$: to balance between exploration and exploitation

$State \setminus v$	1	2	$\ldots$	n_v	n_v+1
$(cpsd_1)^{t_1}$					
$(cpsd_1)^{t_2}$					
$\ldots$					
$(cpsd_2)^{t_1}$					
$(cpsd_2)^{t_2}$					
$\ldots$					
$(cpsd_{n_{st}})^{t_1}$					
$(cpsd_{n_{st}})^{t_2}$					
$\ldots$					

Q-Learning

Reward

$$R^{(t)} = -(TC^{(t)} + \pi^{(t)} + \beta^{(t)})$$

- $TC^{(t)}$: total cost of transportation
- $\pi^{(t)}$: the penalty for infeasibilities in terms of the vehicle capacity, due date, unsatisfied demand, or assigned to a vehicle that does not pass the corresponding origin and destination of that $cpsd$
- $\beta^{(t)}$: the amount of barrier that is added for satisfied constraints

Q-Learning

Reward

$$R^{(t)} = -\left(TC^{(t)} + \pi^{(t)} + \beta^{(t)}\right)$$

- $TC^{(t)} = \sum_{o=1}^{n_o} \sum_{v=1}^{n_v+1} c_{o,v} \mathbb{1}_{o,v}$
- $\pi^{(t)} = \left(1 + \sum_{l \in VC^{(t)}} \mathbb{G}_l'(S^{(t)})\right)^{\xi}$
  - $VC^{(t)} = \{\mathbb{G}(S^{(t)}) \mid \mathbb{G}(S^{(t)}) > 0\}$ (violated constraints)
- $\beta^{(t)} = \sum_{l \in SC^{(t)}} \left(-\log(-\mathbb{G}_l'(S^{(t)}))\right)$
  - $SC^{(t)} = \{\mathbb{G}(S^{(t)}) \mid \mathbb{G}(S^{(t)}) \leq 0\}$ (satisfied constraints)

Q-Learning

Q-Learning

s_n: $\mathrm{cpsd_n}$

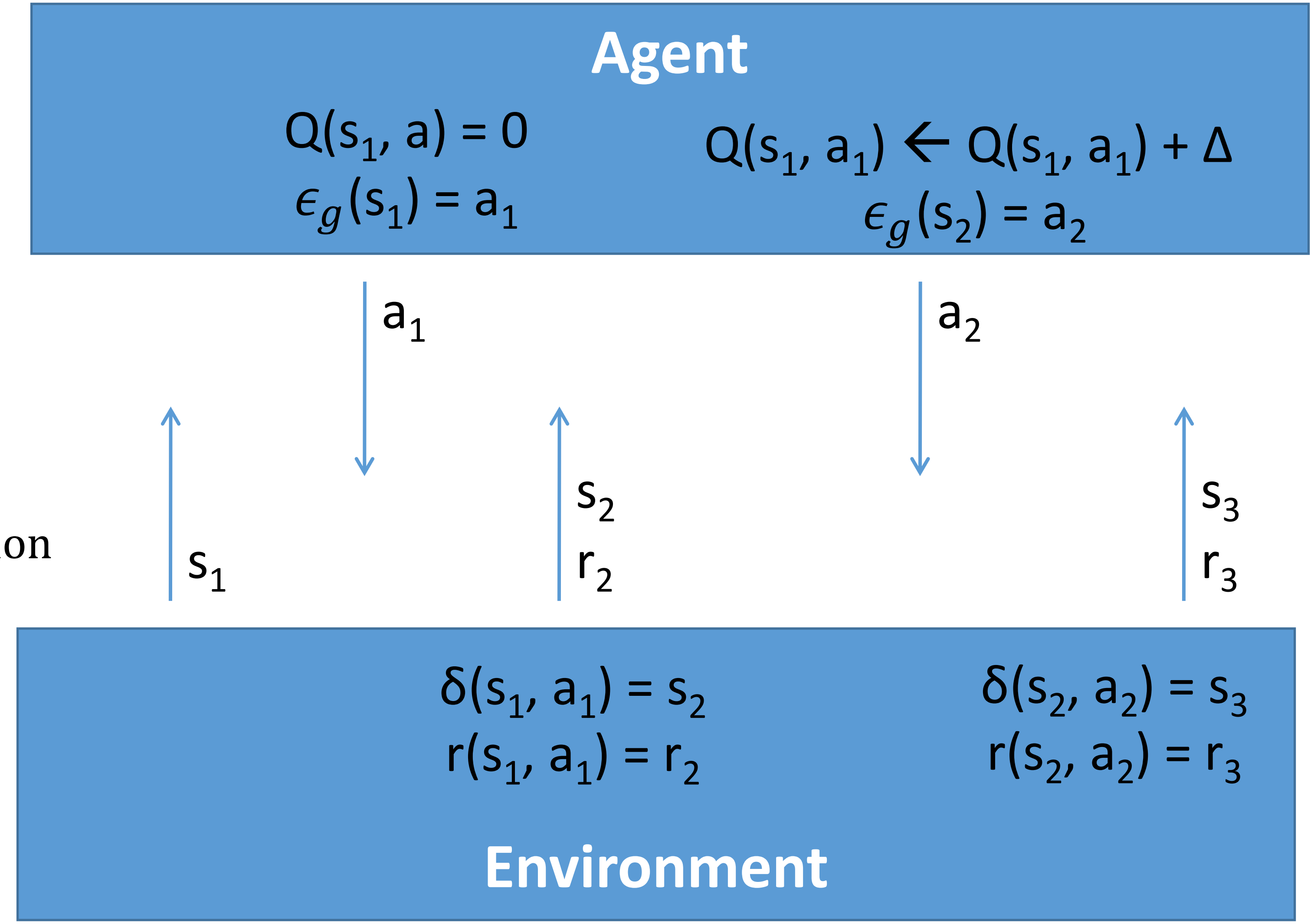
a_v: vehicle

δ: transition function

ϵ_g: ϵ − greedy

Δ: change in Q − value

based on the Bellman equation



Results

Instance Generation and Parameter Settings

- A random instance with

  $n_{cps} = 9$

  $n_t = 7$

  $n_v = 6$

  $n_d = 5$

- Parameters

  $\alpha = .2$

  $\gamma = .8$

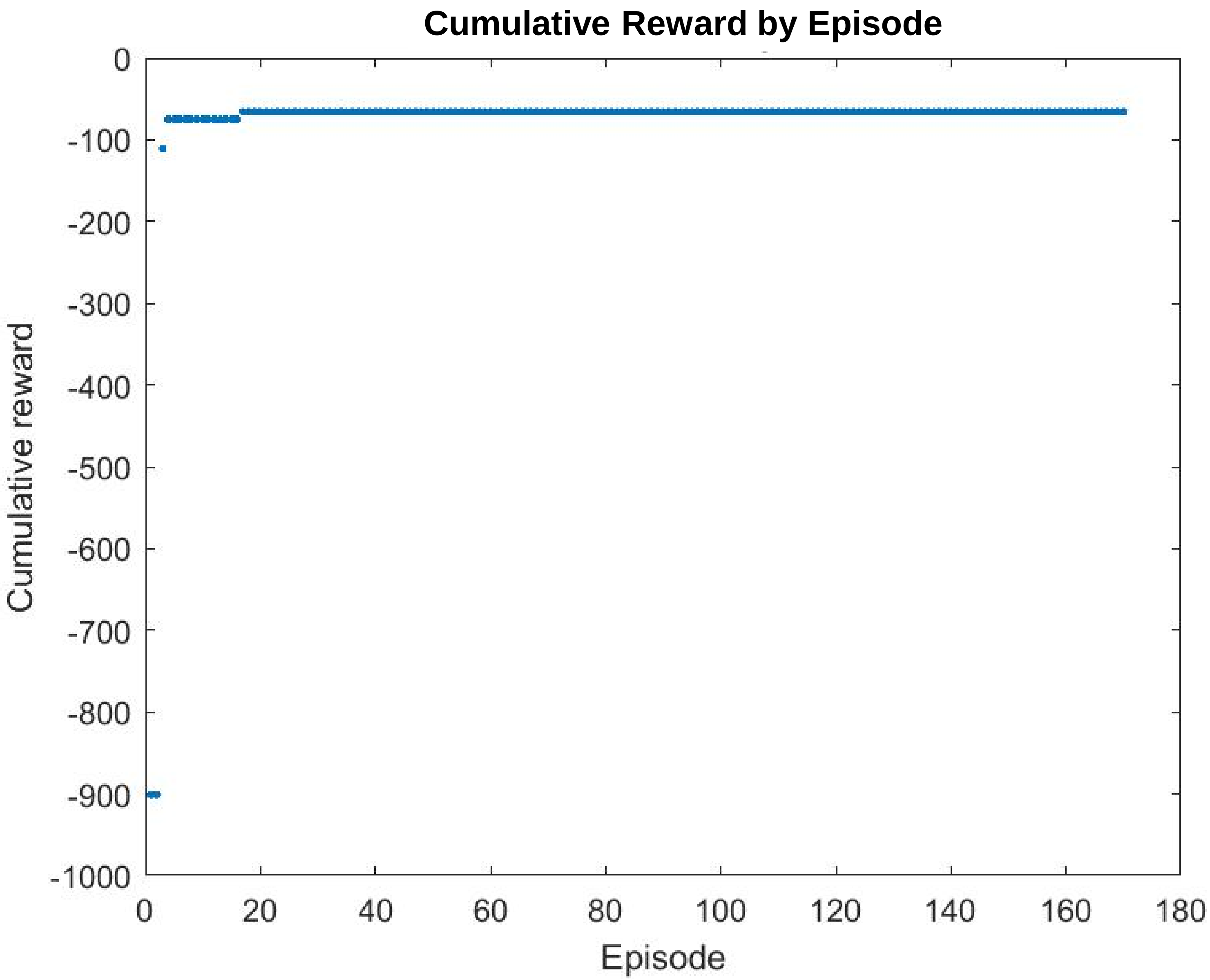
  $\epsilon = .9$

  $1000\ episodes$

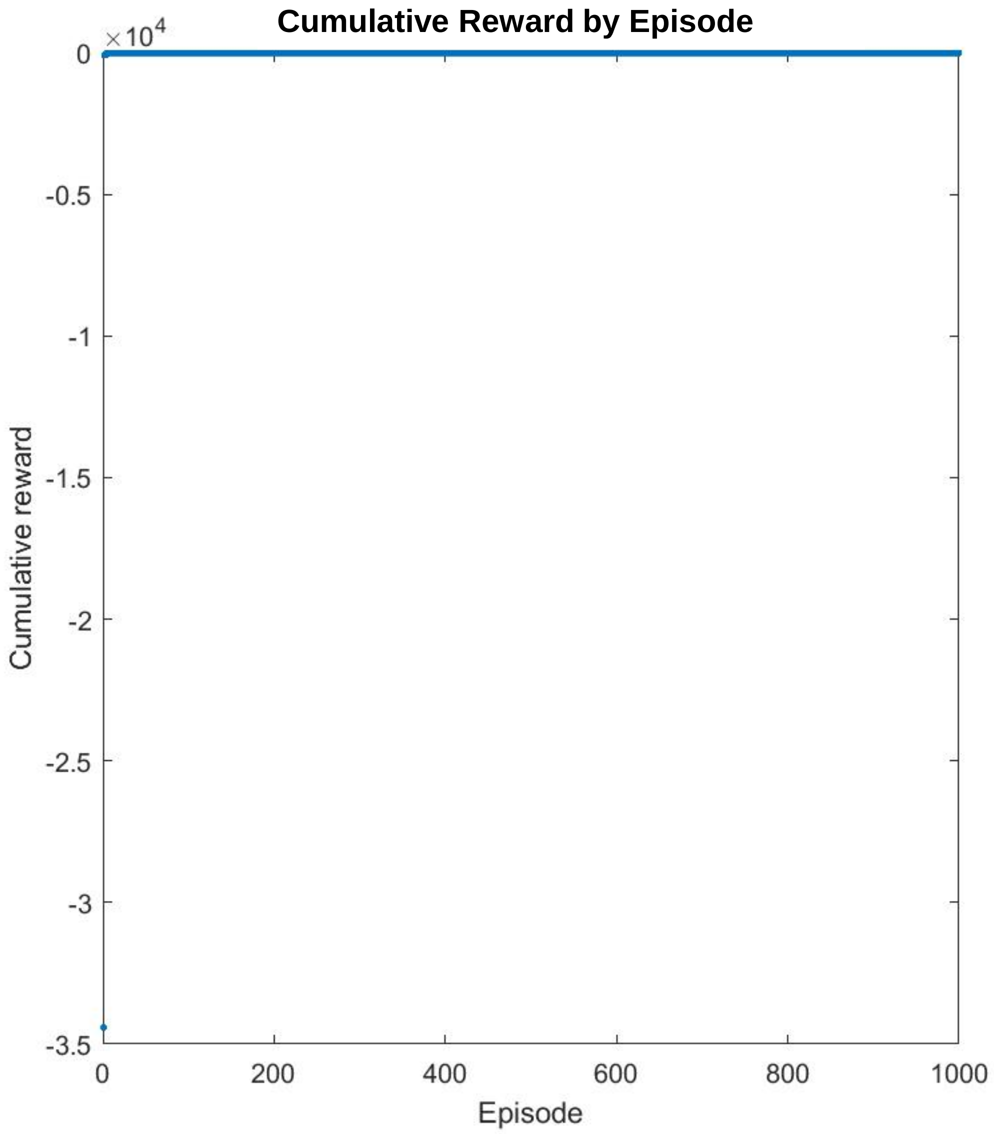
Parameter	Value
Demand for cps_n	$N(\mu_n = 1100 - (n-1) * 100, \sigma_n = 50)$
Order's arrival time	$U(1, [.6 * n_t])$
Delivery time length	$U(1, [.4 * n_t])$
Cost	$U(1,4)\ for\ CS$ $U(10,20)\ for\ DS$
Time	$U(.2,.5)\ for\ DS$ $U(1,2)\ for\ CS$
Weight & volume	$Based\ on\ the\ demands$

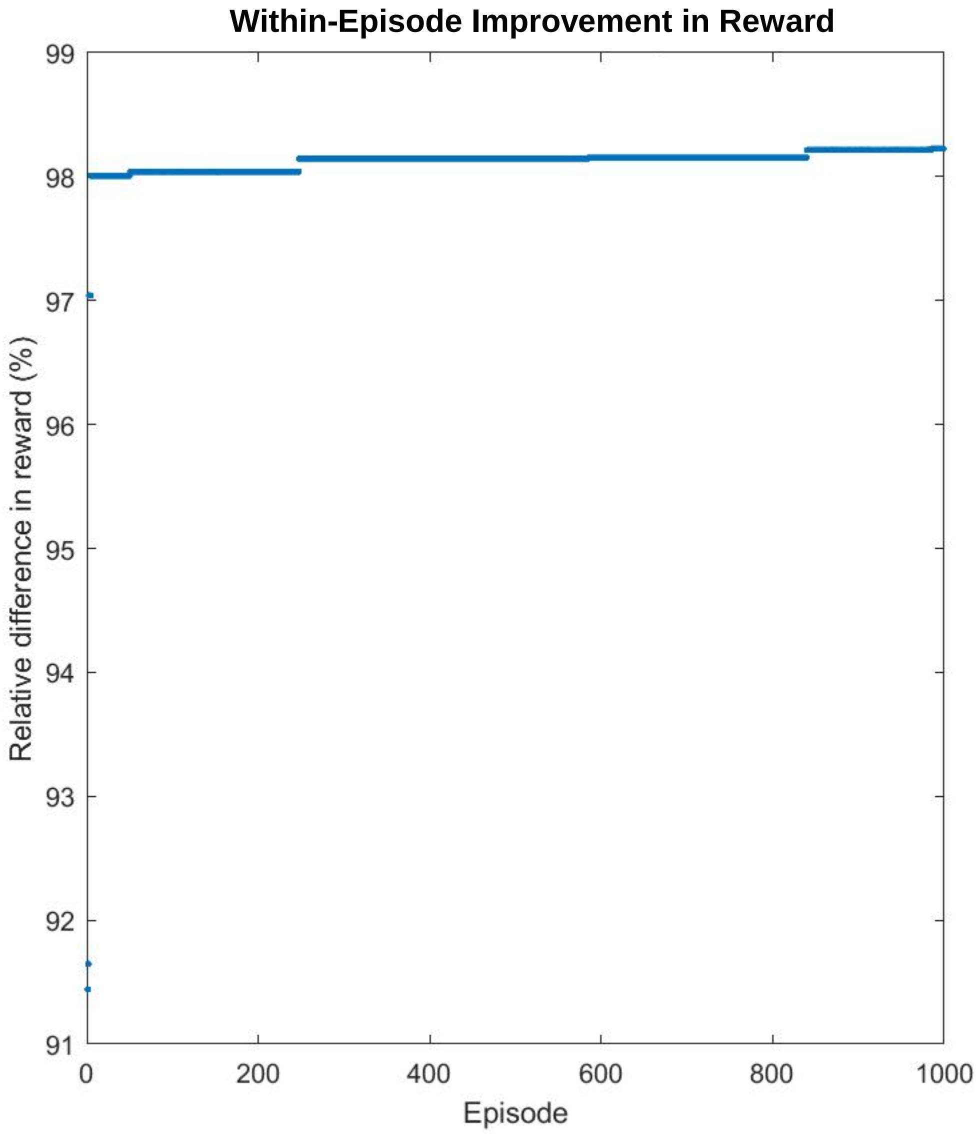
$$Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma \max_a Q(S',a) - Q(S,A)]$$

Results – Episodes 2-170



Results


Cumulative Reward by Episode
×10^4
0
-0.5
-1
-1.5
-2
-2.5
-3
-3.5
Cumulative reward
0
200
400
600
800
1000
Episode


Within-Episode Improvement in Reward
99
98
97
96
95
94
93
92
91
Relative difference in reward (%)
0
200
400
600
800
1000
Episode

Conclusions and Future Research

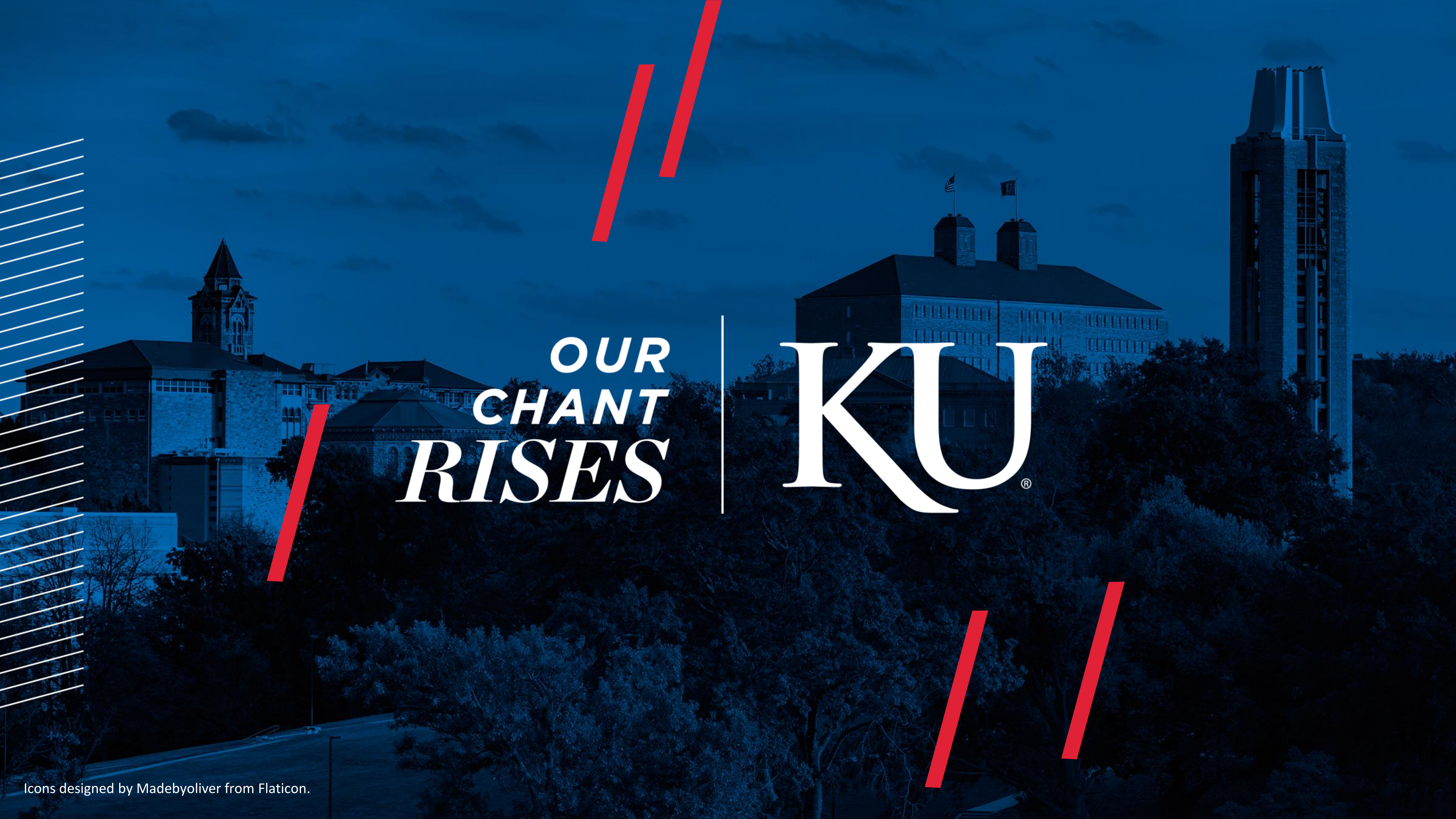
Conclusions and Future Work

Summary

- A modular framework is proposed for the 3PL freight management problem
- Q-Learning algorithm is shown to be able to solve an instantiation of the problem

Future Work

- Refining the framework
- Extending to continuous state space and/or action space (continuous demand, various consolidation strategies, 3-dimensional constraints)
- Using Deep Q-Learning



Icons designed by Madebyoliver from Flaticon.