Triple-BERT: Do We Really Need MARL for Order Dispatch on Ride-Sharing Platforms?

Zijian Zhao, Sen Li
zzhaock@connect.ust.hk

Department of Civil and Environmental Engineering
Hong Kong University of Science and Technology

April 23, 2026

Outline

1. Introduction: Trip-Vehicle Dispatch Task
2. Methodology: Proposed Centralized SARL Framework
3. Experiment: Real-World Ride Sharing Scenario
4. Conclusion

Problem Setup

- **Large-Scale Trip-Vehicle (Order) Dispatch**:
  - ride hailing & ride sharing (pooling)
  - food delivery
  - emergency medical service
- **Key Factors in Ride Sharing**:
  - spatial relationship between trips' origin-destination (OD) and vehicles' location
  - spatiotemporal relationship among en-route trips (for bundling)

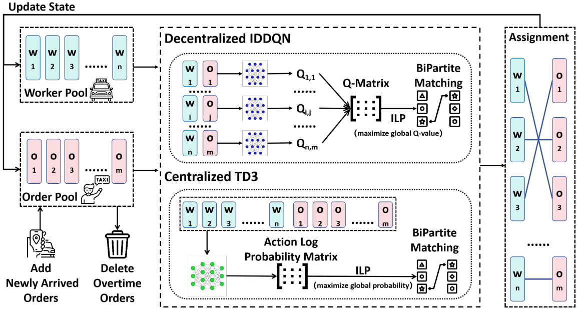

Figure 1: Workflow of Trip-Vehicle Dispatching

Challenge: Large Observation & Action Space

❑ **Curse of Dimensionality (CoD) Challenges**

- Observation Space: vehicle state & trip state

$$|S_t| = b_w n + b_o m_t$$

  - S_t: state (observation); b_w: vehicle state size; b_o: trip state size
  - t: time step; n: vehicle amount; m_t: trip amount

- Action Space: assign which trip to which vehicle

$$\begin{aligned}|A_t| &= \sum_{k=0}^{m_t} \mathsf{C}(m_t,k)\mathcal{P}(n,k) \geq \sum_{k=0}^{m_t} \mathsf{C}(m_t,k)(n-k+1)^k \\ &\geq \sum_{k=0}^{m_t} \mathsf{C}(m_t,k)(n-m_t+1)^k = (n-m_t+2)^{m_t} \geq 2^{m_t} \quad (n \geq m_t \geq 0)\end{aligned}$$

  - A_t: action; C: combinations; $\mathcal{P}$: permutations
  - When the number of vehicles n is 1,000 and the number of trips m_t is 10, the expression $(n-m_t+2)^{m_t}$ evaluates to $992^{10} \approx 10^{30}$.

Previous Work: MARL based Methods

❑ **MARL Challenges**

- Independent MARL
  - unstable environment
  - poor cooperation
- Centralized Training Decentralized Execution (CTDE) / Centralized MARL
  - sample scarcity
  - credit assignment challenge
  - CoD challenge (mixture / critic neural network)

Table 1: Comparison of Different Order Dispatching Methods: * represents the modified version.

Method	**DeepPool**	**BMG-Q**	**HIVES**	**Enders et al.**	**CEVD**	**Triple-BERT (ours)**
Type	Independent		CTDE		Centralized	
RL Algorithm	IDDQN	IDDQN	QMIX	MASAC	VD*	TD3*
Multi-Agent	✓	✓	✓	✓	✓	×
Network Backbone	MLP	GAT	MLP+GRU	MLP+Attention	MLP	BERT

Outline

1. Introduction: Trip-Vehicle Dispatch Task
2. Methodology: Proposed Centralized SARL Framework
3. Experiment: Real-World Ride Sharing Scenario
4. Conclusion

Overview

❑ **SARL Challenges:**

- State Space CoD:
  - A BERT-based network for extracting features and relationships from trips and vehicles.
- Action Space CoD:
  - A novel action decomposition mechanism that factorizes the joint action probability into virtual selection probabilities for each vehicle-trip pair.
  - A novel multiplicative network that reduces computational complexity from multiplicative to additive.
- Data Scarcity:
  - MARL-based pre-training to enhance the encoder's general feature extraction capacity.
  - SARL fine-tuning to derive a more powerful policy.

Neural Network Architecture

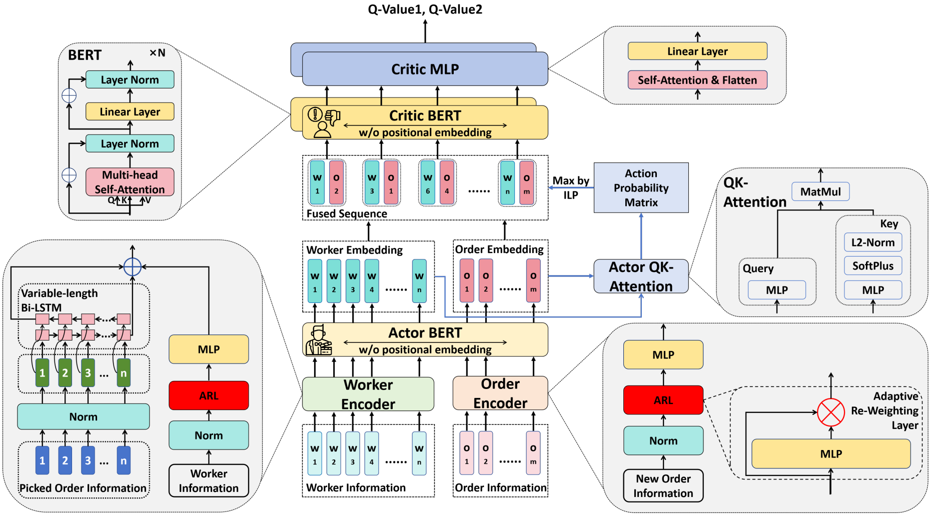

Figure 2: Network Architecture

Neural Network Architecture

❑ **QK-Attention: Transfer Multiplication Complexity to Addition Complexity**

- QK-Attention for Order Assignment (Zhao et al. 2025)

$$\text{QK-Attention}(\overline{w}_{i,t}, \bar{o}_{j,t}) := \mathsf{f}(\overline{w}_{i,t}; \theta_f) \cdot \mathsf{g}(\bar{o}_{j,t}; \theta_g)^T \approx \mathsf{F}(\overline{w}_{i,t}, \bar{o}_{j,t}; \theta_F) \in \mathbb{R}$$

  - $d << |\mathsf{f}| \approx |\mathsf{g}| < |\mathsf{F}|$; $|f|$: function computation complexity; d: hidden dimension
  - $\overline{w}_{i,t}$: embedding of vehicle i state; $\bar{o}_{j,t}$: embedding result of trip j state

- Normalized QK-Attention

$$\text{QK-Attention-Norm}(\overline{w}_{i,t}, \bar{o}_{j,t}) := \mathsf{f}(\overline{w}_{i,t}; \theta_f) \cdot \frac{\text{Softplus}(\mathsf{g}(\bar{o}_{j,t}; \theta_g))^T}{||\text{Softplus}(\mathsf{g}(\bar{o}_{j,t}; \theta_g))||_2}$$

  - mitigate the parameter redundancy problem, inspired by Dueling DQN

Training Process

- **Stage 1: IDDQN Pre-training**
  - Update by Minimizing TD-Error

$$L_Q = \mathbb{E}_{\pi^Q_\Phi}\left[\mathcal{Q}^{DQN}_{\pi^Q_{\Phi^-}}(s_{i,t+1}, r_{i,t+1}; \Phi^-) - \mathrm{Q}^{DQN}_{\pi^Q_\Phi}(s_{i,t}, a_{i,t}; \Phi)\right]$$

$$\mathcal{Q}^{DQN}_{\pi^Q_{\Phi^-}}(s_{i,t+1}, r_{i,t+1}; \Phi^-) = r_{i,t+1} + \gamma \mathrm{Q}^{DQN}_{\pi^Q_{\Phi^-}}(s_{i,t+1}, \arg \max_{\kappa_{i,t+1} \in \psi_{i,t+1}} \mathrm{Q}^{DQN}_{\pi^Q_\Phi}(s_{i,t+1}, \kappa_{i,t+1}; \Phi); \Phi^-)$$

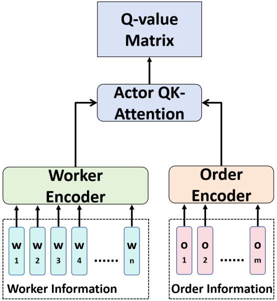

Figure 3: Network Architecture in Stage 1

Training Process

- **Stage 2: TD3 Fine-tuning**
  - Critic: Update by Minimizing TD-Error

$$L_C = \sum_{i=1,2} \mathbb{E}_{\pi^T_{\Theta}} \left[\mathcal{Q}^{TD3}_{\pi^T_{\Theta^-}}(S_{t+1}, R_{t+1}; \Theta^-) - \mathrm{Q}^{TD3}_{\pi^T_{\Theta},i}(S_t, A_t; \Theta) \right]$$

$$\mathcal{Q}^{TD3}_{\pi^T_{\Theta^-}}(S_{t+1}, R_{t+1}; \Theta^-) = R_{t+1} + \gamma \min_{i=1,2} \mathrm{Q}^{TD3}_{\pi^T_{\Theta^-},i}(S_{t+1}, \mathrm{Actor}(S_{t+1}; \Theta^-, \xi); \Theta^-)$$

    - $\Phi \subset \Theta$: policy network parameters; Φ^-, Θ^-: target network parameters
    - ψ: feasible action sets; π^Q, π^T: policy of IDDQN and TD3
  - Actor: Action Decomposition

$$\pi^T_{\Theta}(A_t | S_t) = \mathrm{z}\Big(\prod_{i,j \in \mathrm{h}(A_t)} \mathscr{P}_{i,j,t} \Big)$$

    - $\mathscr{P}_t$: virtual action choosing probability generated by QK-Attention
    - $\mathrm{h}(A_t) = \{(i,j) | \text{if vehicle } i \text{ chooses order } j \text{ in } A_t\}$
    - $\mathrm{z}(\cdot)$: probability mapping function (increasing function)

Training Process

- **Stage 2: TD3 Fine-tuning**
  - Actor: Action Optimization (Policy Gradient)

$$
\begin{aligned}
\nabla_\Theta \mathrm{J}(\Theta) &\propto \mathbb{E}_{\pi^T_\Theta}\left[\left(\mathrm{Q}^{TD3}_{\pi^T_\Theta}(S_t, A_t) - B\right)\nabla_\Theta \log \pi^T_\Theta(A_t|S_t)\right] \\
&= \mathbb{E}_{\pi^T_\Theta}\left[\left(\mathrm{Q}^{TD3}_{\pi^T_\Theta}(S_t, A_t) - B\right)\nabla_\Theta \log \mathsf{z}\left(\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t}\right)\right] \\
&= \mathbb{E}_{\pi^T_\Theta}\left[\left(\mathrm{Q}^{TD3}_{\pi^T_\Theta}(S_t, A_t) - B\right)\frac{d\mathsf{z}(\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t})}{d\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t}} \frac{\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t}}{\mathsf{z}(\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t})} \nabla_\Theta \log \prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t}\right] \\
&= \mathbb{E}_{\pi^T_\Theta}\left[\left(\mathrm{Q}^{TD3}_{\pi^T_\Theta}(S_t, A_t) - B\right)\mathcal{E}_{\mathsf{z}(x),x}\big|_{x=\prod_{i,j\in \mathsf{h}(A_t)} \mathscr{P}_{i,j,t}} \nabla_\Theta \sum_{i,j\in \mathsf{h}(A_t)} \log \mathscr{P}_{i,j,t}\right]
\end{aligned}
$$

- $\mathcal{E}$: elasticity ($\mathcal{E}_{y,x} = \frac{d\log y}{d\log x}$); B: baseline

Training Process

- **Stage 2: TD3 Fine-tuning**
  - Actor: Action Optimization (Policy Gradient)
    - If we can view $\mathcal{E}_{\mathsf{z}(x),x}|_{x=\prod_{i,j\in\mathsf{h}(A_t)}}$ as a positive constant (restrict the policy in a structural space), then:

$$\nabla_\Theta \mathrm{J}(\Theta) \propto \mathbb{E}_{\pi_\Theta^T}\left[\left(\mathrm{Q}_{\pi_\Theta^T}^{TD3}(S_t,A_t) - B\right)\nabla_\Theta \sum_{i,j\in\mathsf{h}(A_t)} \log \mathscr{P}_{i,j,t}\right]$$

  - Actor: Action Choice
    - Evaluation: select the action with highest probability (utility) greedily
    - Training: add random noise to $\mathscr{P}_t$ and choose the action with highest probability

$$\arg\max_{A_t\in\psi(S_t)} \pi_\Theta^T(A_t|S_t) = \arg\max_{A_t\in\psi(S_t)} \mathsf{z}\Big(\prod_{i,j\in\mathsf{h}(A_t)} \mathscr{P}_{i,j,t}\Big) = \arg\max_{A_t\in\psi(S_t)} \sum_{i,j\in\mathsf{h}(A_t)} \log \mathscr{P}_{i,j,t}$$

Outline

1 Introduction: Trip-Vehicle Dispatch Task

2 Methodology: Proposed Centralized SARL Framework

3 Experiment: Real-World Ride Sharing Scenario

4 Conclusion

Experiment Setup

❑ **Real-World Ride Hailing Dataset**

- Training Set: Yellow Taxi Data from Manhattan, New York City
- Test Set: Yellow Taxi Data from Queens, New York City

❑ **Reward: Comprehensive Function**

$$\mathcal{R}(s_{i,t}, a_{i,t}) = \begin{cases} \beta_1 + \beta_2 p_{i,t}^{in} - \beta_3 p_{i,t}^{out} - \beta_4 \chi_{i,t} - \beta_5 \rho_{i,t} \ , & |a_{i,t}| = 1 \\ 0 \ , & |a_{i,t}| = 0 \end{cases}$$

- $p_{i,t}^{in}$: income from customers; $\chi_{i,t}$: estimated overtime of all en-route trips
- $p_{i,t}^{out}$: payment to vehicles; $\rho_{i,t}$: added travel time of all en-route trips

Experiment Result

❑ Training Process

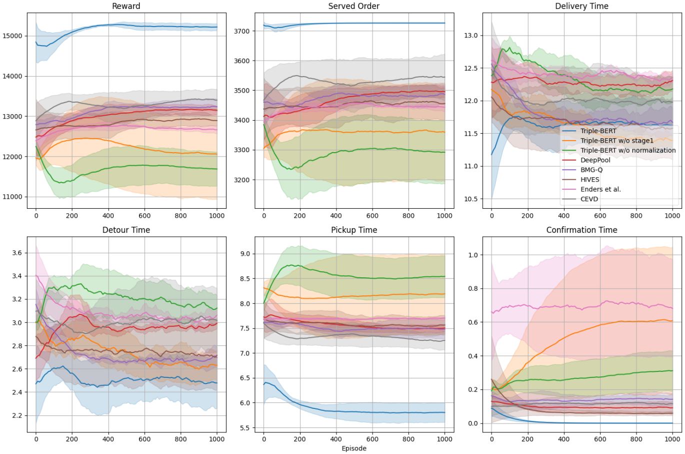

Figure 5: Training Process

Experiment Result

❑ Evaluation on Testing Set

- Our method primarily optimizes pickup time, enabling vehicles to serve more orders within a given time window, thereby achieving higher rewards.
- However, the increased number of served orders inevitably leads to longer detour and delivery times due to more frequent order bundling.

Table 2: Performance in Manhattan FHV Data

Method	Reward	Service Rate	Delivery Time	Detour Time	Pickup Time	Confirmation Time
DeepPool	11258.56±2578.68	0.76±0.18	13.93±0.93	2.54±1.48	10.19±0.78	0.09±0.07
BMG-Q	11899.39±2804.65	0.78±0.17	13.27±1.00	2.44±1.54	9.39±0.43	0.15±0.11
HIVES	11183.12±2458.29	0.78±0.18	13.72±1.48	2.87±1.90	9.77±0.68	**0.05±0.03**
Enders et al.	10512.65±2744.34	0.78±0.17	14.12±0.48	3.06±0.43	9.92±0.55	1.37±0.99
CEVD	12556.74±3303.93	0.80±0.13	**12.33±0.72**	**2.25±1.33**	8.02±1.27	0.09±0.08
Triple-BERT	**14329.74±4627.26**	**0.88±0.11**	13.07±0.61	2.78±0.92	**7.02±0.88**	0.34±0.32

Outline

1. Introduction: Trip-Vehicle Dispatch Task

2. Methodology: Proposed Centralized SARL Framework

3. Experiment: Real-World Ride Sharing Scenario

4. Conclusion

Conclusion

❑ We introduce the first centralized SARL framework, Triple-BERT, for order dispatch in ride-sharing scenarios.

❑ We address the challenges of CoD and data scarcity in conventional methods through a BERT-based network architecture, a novel action decomposition mechanism, and a MARL pre-training approach.

❑ Our method consistently outperforms previous MARL solutions, achieving approximately 14% improvement in overall reward across various scenarios.

Q&A

Thanks for your attention!