Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

A Deep Reinforcement Learning Approach to Dynamic Pricing in Competitive Ride-Hailing Platforms

[1] Dr. Shahzaib Khan

[2] Rabia Nadeem

How to cite this article:

Khan, S., & Nadeem, R. (2026). A Deep Reinforcement Learning Approach to Dynamic Pricing in Competitive Ride-Hailing Platforms. Journal of Technological Information, Management & Engineering Sciences, 1(1), 104-121.

Received	Accepted	Published
05 October 2020	09 November 2020	14 December 2020

Abstract

Ride-hailing markets are dynamic, with passenger demand, driver availability, traffic conditions, weather, events, and rival-platform prices changing over time. Static pricing rules and short-term demand-response approaches might not be able to capture these coupled dynamics, and unconstrained price optimization could raise issues of fairness, transparency and affordability. The paper suggests a deep reinforcement learning (DRL) approach for dynamic pricing in a competitive ride-hailing market. The pricing problem is cast as a sequential decision process where the agent observes the market conditions, chooses bounded fare multipliers or fare changes, and is rewarded according to the platform contribution, trip completion, service reliability, driver participation, and customer outcomes. The proposed architecture is composed of state representation via spatiotemporal demand forecast, actor–critic policy with constraints, rival-response model and safety measures to restrict extreme price fluctuation. Training is done in a simulation environment before careful evaluation in the "offline" phase and staged deployment. The research design involves comparing the DRL policy with fixed pricing, a rule-based surge pricing policy, a contextual bandits policy, and a model-predictive control policy for revenue, fulfillment, wait time, cancellation, price volatility, and distributional fairness metrics. The focus is special placed on non-stationarity, multi-agent competition, causal evaluation, and bias in historical data. Rather than purporting to demonstrate the empirical performance benefits, the paper provides a framework for measuring the potential for constrained DRL to enhance marketplace efficiency, while maintaining user trust and regulatory compliance.

Keywords: *deep reinforcement learning, dynamic pricing, ride-hailing, surge pricing, marketplace competition, actor–critic, fairness, and simulation.*

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan

[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)

This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

1. Introduction

Ride hailing provides on-demand access to vehicles to passengers and drivers to passengers through digital platforms that are constantly estimating demand, allocating vehicles and calculating fares. The price of a ride is not the same as it is at a traditional retail outlet, but, rather, varies by location, time, distance, traffic conditions, availability of the driver to accept the ride request and the availability of other services that can provide similar transportation. Demand can vary significantly in the short term following a concert, during the rain, at commuting periods or in the event of disruption in public transport. Supply is also responsive: drivers switch between zones and log on and off and select among platforms depending on anticipated earnings and waiting time. Pricing is not just a one-time fare determination, but a decision of control that can affect the shape of the whole marketplace in the future.

Dynamic or surge pricing is applied to incentivize supply to meet demand, is common and is used to ration scarce capacity. But a price rise can turn people off, can result in cancelation, can cause people to switch to another platform or could lead to loss of trust if they don't understand why. On the other hand, if prices are too low during a surge in demand, it may result in long queues and delayed orders. Any pricing strategy that would be useful should take into account short term financial results as well as the quality of services provided and the participation of both sides of the market in the longer term.

Deep reinforcement learning provides a framework for learning sequential decisions in environments where actions change the next state, and rewards. Deep neural networks enable modeling of many complex state variables and reinforcement learning enables optimization of a policy over repeated decisions. However, the application of DRL to ride-hailing pricing is challenging: the environment is non-stationary, observations are incomplete, competitors have strategic behaviour, previous pricing policies are confounded in the historical data, and harmful exploration can have direct impact on customers. These constraints call for a framework that entails learning alongside clear limits of operation and ethics.

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

This paper proposes a research design on the constrained DRL-based pricing. It is not the assertion that reinforcement learning is necessarily better than current best practices, but rather a general framework for comparing reinforcement learning to competitive baselines in a realistic marketplace setting.

1.1 Research objectives

Develop dynamic pricing as a sequential decision problem for market-place, driver, passenger, and competitor states.

Create an architecture for DRL that incorporates forecasts, strategic competition, bounded price actions and operational constraints.

Establish a protocol and metrics for evaluating the financial performance, service reliability, stability, and fairness for a simulation.

Outline governance, privacy and deployment controls required for appropriate use of learned pricing policies.

1.2 Research questions

How to model the spatiotemporal and competitive dynamics that drive ride-hailing fares?

What are the conditions under which the constrained DRL can be better than the rule-based surge pricing and other benchmarking methods in simulated markets?

3. What should a pricing policy do regarding revenue/fulfillment and affordability, price volatility and distributional effects?

4. What evaluation and governance process is required to enable a learned policy to be put into a real marketplace?

2. Literature Review

In this course, you will explore dynamic pricing and two-sided marketplaces in greater depth.

Dynamic pricing refers to the ability to adjust prices based on fluctuations in demand, capacity and market conditions. In the ride-hailing business, fares are a revenue source and also a means of coordination. Higher prices may deter some requests, and encourage drivers to pick up more loads in a high demand area. The resulting response is endogenous: A change in price changes demand and supply, which then changes the conditions that justify the change in price. This feedback

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

loop differentiates marketplace pricing from situations where demand is somewhat independent of the seller's activities.

Two side market research highlights the importance of understanding two-sided markets and considers both sides of the market instead of optimizing one market side alone. Passenger value is determined by price and service availability, driver value is determined by earnings, utilization and time to pickup. The logic behind a fare policy geared toward maximizing immediate fare revenue may actually backfire if it leads to lower passenger retention, or to unstable driver incentives. This should therefore be a long horizon objective that encompasses current results as well as future health of the marketplace.

2.2 Reinforcement learning for sequential control

RL is a learning method where an agent interacts with an environment, sees what state it is in, chooses an action and is rewarded. The aim is to train a policy that gives a max expected total reward. In the case that the state space is large enough for tabular methods to be impractical, deep reinforcement learning is used to approximate value functions or policies with neural networks. One important distinction is that while DQN methods can be used with discrete actions, policy-gradient and actor–critic can be used with continuous/hybrid actions. Proximal Policy Optimization (PPO) has become popular as the objective uses a clip to avoid overly large policy updates during training (Schulman et al., 2017).

The delay of the impact of an action is made attractive by DRL. A fare adjustment can affect driver positioning, availability, user retention and the actual conversion rate. But the design of rewards is key: an agent will maximize its reward function, even if it doesn't consider outcomes that are important to users. Constraint-based optimization, well-designed penalties and external monitoring are key and not just ancillary features.

Explain the concepts of learning in competitive and non-stationary environments.

The transition dynamics can be more challenging when there is competition. A competitor could react to a price modification, modify driver incentives, or employ a different promotion. The environment in which the agent observes the competition evolves over time, which breaks the stationary assumptions in simple

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

reinforcement-learning analyses. While strategic interactions can be represented by multi-agent reinforcement learning and game-theoretic models, they can also lead to instability and make the behavior of the policies more difficult to explain.

It's perfectly fine to assume that everyone won't know each and every policy as a practical design. Instead, a single agent may observe the market and use indicators for this market, which can be ranges of estimated fares for competitors, relative availability, etc., and have some uncertainty estimates. The pricing system should be tested for a number of competitor-response scenarios rather than for a single competitor model. Good resistance to unexpected actions of rivals is an important deployment criterion.

2.4 Fairness, explainability, and responsible pricing

Price discrimination is acceptable when it is due to a legitimate variation in the cost or scarcity of the trip, but can also have an unequal impact on specific neighborhoods or user groups. Geographic pricing may compound with income, disability, and access to alternatives and transportation. If the reward function assigns a high value to a request that is considered high-margin but does not take access into account, a model will be able to replicate historical inequities if historical data indicate disparities in service coverage. It is therefore important that fairness assessment assess outcomes at relevant locations and time periods and does not unnecessarily collect sensitive personal information.

Explainability is also an operational value. There may be reasons that are understandable to customers and regulators for a fare, such as the high local demand for the service or low supply of nearby drivers. Feature summaries, policy constraints, policy price change logs, and counterfactual analysis are all methods that can support explanations for a complex policy. Explanations should refer to the real factors employed by the system, not just backfill explanations that are not accurate to the model.

2.5 Research gap

There are several concerns that are usually treated separately in existing studies of reinforcement learning, revenue management and marketplace optimization, and the pricing problem for ride-hailing services must account for them: the two-

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

sided nature of the market, strategic competitor response, spatial movement of drivers, safe action constraints, off-policy evaluation, and distributional effects. To address that integration, the present paper proposes a simulation-based architecture as well as an evaluation plan that addresses a set of multiple objectives.

3. Problem Formulation

The region of the service be divided into zones numbered by i and the decision intervals be denoted by t (for instance, 5 minutes interval). Every time the platform registers a state s_t, it contains estimates of demand, available drivers, the number of active trips, the pickup time, the travel time, the number of cancellations, the weather data, the local events, the traffic, the conditions of the competitors, and the price history. The policy defines a price action a_t for each eligible zone or zone–trip class. The action can be either as a bounded multiplier m_i,t on a base fare, or as a bounded change in multiplier compared with its previous value.

The environment changes then as passengers accept or reject rides, the drivers accept or reject, reposition, complete trips and also update their own offers. This platform earns a reward r_t. Some state variables are unobserved and measurements are subject to noise, which makes the problem a more realistic partially observable Markov decision process (POMDP). Using the temporal information can be possible by having a recurrent encoder or a short history window to the policy.

3.1 State representation

A small state may have the following feature groups:

Demand: recent requests, conversion rate, estimated unmet demand and probabilistic demand forecasts by zone and time horizon.

Supply: Idle Drivers, Driver arrivals, Active-trip distribution, Driver acceptance rates and estimated repositioning time.

Service quality: Estimation of pickup times, trip completion, cancellation, unserviced trips.

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

Weather, traffic, public events, calendar effects and interruptions to other modes of transport.

Competition: knowledge of competitor prices, promotions, service availability, and uncertainty of competitor reactions.

Policy history: exposure to price caps or fairness constraints, current fare multiplier, and recent changes in the multiplier.

The features are to be aggregated at an appropriate spatial and temporal resolution. For the core control loop, the precision of the passenger identity is not required; if it is not needed for the core control loop then it is not a concern for the zone-level or trip-class aggregates.

3.2 Action space

The easiest is a fixed list of permitted multipliers, e.g. {0.9, 1.0, 1.1, 1.2, 1.35, 1.5} with local legal and product restrictions. Alternatively, a continuous action may emit a range of multipliers within a specified range. The final action is to be passed through a safety layer which imposes minimum and maximum fares, has a cap on how quickly the rate can change, does not allow for any invalid quotes and has the ability to apply any emergency or accessibility specific rules as required. Discrete actions can be easier to run a review on, continuous actions can be more difficult but provide more precise control – and require more careful handling of constraints.

3.3 Reward design

A candidate reward is a weighted average of contribution margin, completed-trip value, fulfillment, and service quality minus penalties for cancellation, excessive delays, volatility and unequal outcomes:

The error term of the regression model is $r_t = \alpha \cdot Margin_t + \beta \cdot CompletedTrips_t + \gamma \cdot Fulfillment_t - \delta \cdot Wait_t - \varepsilon \cdot Cancel_t - \zeta \cdot Volatility_t - \eta \cdot FairnessGap_t$

The weights α through η are weights that represent policy priorities and will have to be determined by stakeholder review and sensitivity analysis, rather than tuned to maximize revenue. Reward terms should be normalized to prevent a large-scale metric from overwhelming all others. Certain requirements can also be expressed as hard constraints, rather than penalties that the agent reasonably

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

could break if other reward components are large enough, or even a legal ceiling, like a price ceiling.

Long-horizon performance may be summarized as the discounted return G_t = Σ_k γ^k r_(t+k), where γ is a parameter that determines the relative importance of future performance. A large discount factor makes it easier for the agent to consider retaining its agent or future supply, thus may make training harder or more sensitive to model error.

4. Proposed Hybrid DRL Framework

The proposed framework consists of five interrelated modules: a state and forecast module, a policy learner module, a marketplace simulator module, a safety and fairness module, and a monitoring module. This hybrid design separates the predictive part from the policy decisions to enhance debugging and facilitate the replacement of the forecasting model without redesigning the learning system.

To create a forecast and feature layer.

Demand, supply, pickup time and cancellation forecasts are structured estimates of market conditions for the near future. Short-term forecasts can be produced via gradient-boosted trees, temporal convolutional networks or recurrent models that are trained on the historical time series of zone-level forecasts. Forecast uncertainty should be carried through, such as by using quantile predictions or ensembles, as there should be different policy behaviour in cases where the demand forecasts are reliable versus when they are very uncertain. The forecasts are not facts, but inputs for the agent, and the policy should also be guided by what has actually happened recently.

4.2 Actor–critic policy

For the main policy an actor-critic approach is suggested. The actor assigns the state representation to a distribution over the price actions, and the critic approximates the expected return. Because clipped policy updates can yield a reasonably stable baseline for policy optimization, PPO can be applied in simulation training. While performing zone-specific actions the actor could share an encoder throughout the zone. Sparse areal data may be used for parameter

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

sharing to enhance learning, but the model should not consider the neighborhoods in the sparse area to be interchangeable in terms of supply conditions.

4.3 Competitor-response model

The environment simulator should include several of the competitive behaviors: fixed price, reactive matching, aggressive discounting, delayed response, and stochastic price changes. The probability that rivals may adjust prices or stock levels may be predicted by a learned response model, given the state of the market. Such estimates can be incorrect, so parameters of the competitors should be randomized and stress situations should be used during training. The platform shouldn't rely on private competitor data or coordinate illegally – competitive signals must be from a legitimate, authorized source.

The importance of safety and fairness layer 4.4

A deterministic layer performs compliance checks on fare caps, rate-of-change limits, service rules, and other constraints prior to an action being placed at the fare engine. The price distribution and service outcomes are assessed separately by a fairness monitor by zone and by time. If the learned policy breaches a guardrail, the system can be clipped, return to a validated baseline or stop automated control. The controls need to be tested when operating under load and should be recorded as an integral part of the production system.

The system has to be monitored and be supervised by human.

Revenue and cost, as well as fulfillment, wait time, cancellations, quote-to-book conversion, driver acceptance, price dispersion, and customer complaints should be tracked by a dashboard. Alerts should detect unusual geographical patterns, missing data, model drift, or sudden changes. Humans should be able to stop the policy and look at the decisions and then resume a known safe pricing policy. Audiability of model version, state features, action, guardrail interventions, and outcome summaries should be recorded, and retention and access as per privacy requirements.

5. Research Methodology

5.1 Research design

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

The paper introduces a computational experimental design instead of a field trial completed. Requests, driver movement, trip duration, passenger choice and competitor response will be modeled using a discrete-event simulator. Use historical aggregates that are suitably anonymized and documented assumptions should be used for calibration. Synthetic parameters can be used for preliminary stress testing, but will not show the true-impact test. Before making policy comparisons, the simulator should be tested against observable aggregate patterns.

5.2 Environment and transition model

For every zone and time period, it is possible to model the demand arrivals as time-varying count distribution whose intensity is dependent on calendar, weather, events, fare, and service availability. A discrete-choice model can be used to represent passenger choice, where the probability of booking is dependent on the fare offered, wait time, reliability of the offer, and perceived alternatives offered by competitors. Expected earnings, distance to pickup, and repositioning costs can be considered in the driver's decision. Trip time should be dependent on distance, traffic and time of day. Where possible these components are calibrated separately and are subject to uncertainty analysis.

If the simulator is too simplistic, then a policy that takes advantage of unrealistic assumptions is rewarded. For instance, if the model takes the assumption that drivers will immediately pull from the price increase, the learned policy might end up using surge pricing too much. Demand elasticity, driver responsiveness, competitor adaptation and the time lag between the price change and the supply response should all be changed during the sensitivity analysis.

5.3 Training procedure

6. Ensure that training, validation and test periods are chronological to minimize the risk of leakage from future conditions.

6. Realistic distributions of demand, driver availability, and traffic are swam into the environment at the start of a session, and randomised throughout training.

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Use multiple random seeds and early stopping to train actor-critic policy to maximize the constrained reward.

Test and freeze the design using the validation set only, tuning reward weights and guardrails.

9. Review the frozen policy for unknown periods, extreme demand situations, data failure and other competitor alternatives.

2. Conduct ablation studies, excluding the competitor model, forecasts, volatility penalty, or fairness conditions, to estimate the contribution of each module.

5.4 Baselines

The baselines should provide a minimum of four points. Fixed pricing is when fares are set constant. (2) Rule-based surge pricing raises fares when a ratio of demand to supply is above certain levels. (3) A contextual bandit makes its choice of price from a set of price options based on the current context, but does not explicitly optimize long horizon transitions. (4) MPC requires a finite planning horizon and predictions to make decisions. A fifth baseline can be a guided policy that learns to model the historical pricing. Equal information should be provided and tuned fairly for baselines.

5.5 Evaluation metrics

There is no single measure that reflects the quality of the marketplace. Evaluation should include mean and distributional results, confidence intervals for seeds, and peak and non-peak performance.

Dimension Metric Purpose

Gross booking value Measures economic outcomes, does not equate gross revenue with profit.

Marketplace efficiency Fulfillment rate; completed trips; idle-driver time Measures matching efficiency and use of available capacity.

Passenger experience Median and 90th percentile service quality for pick up time; cancellation rate.

Price stability Mean absolute price change; Multiplier variance Detects erratic or oscillating prices.

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Driver outcomes Earnings per online hour; acceptance; repositioning distance Evaluate the supply side participation and burden.

Fairness and access Price and fulfillment gaps across zones and time periods Tests whether costs or service degradation are grouped together.

Robustness Performance under demand shock, changes in competition, missing data Evaluates resistance beyond normal circumstances.

5.6 Statistical analysis

Report mean outcomes and uncertainty for independent seeds and test scenarios for each policy. Bootstrapping confidence intervals can measure uncertainty regarding the difference between policies. Scenarios are paired, so where possible the same demand traces and random seeds should be used for the same demand across methods. Do report effect sizes as well as significance. The results ought to be broken down by region, time, demand strength and applicable accessibility requirements. Use preregistration of primary outcomes and analysis choices to avoid multiple comparisons and post hoc metric selection.

6. Algorithmic Workflow

The proposed cycle for decision is outlined below. The safety layer is still on in simulation and will be on in any subsequent deployment.

Review zone-level demand, supply, service quality, context, competitor indicators and recent price trending.

12. Forecasts for demand, pickup-time and supply with uncertainties.

13. Use the actor policy to encode the current state and to choose candidate price actions.

Implement hard fare bounds, rate of change bounds, fairness rules, and operational constraints.

Display allowable fare rates and run a simulation or observe the passenger/driver reactions.

Compute the reward for margin, completed trips, fulfillment, waits, cancellations, volatility and fairness indicators.

Keep privacy-preserving transition data; only update the policy in controlled training cycles.

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

Track guardrail intervention and performance drift and restore to a validated baseline if needed.

6.1 Pseudocode

Set up actor policy πθ, critic Vφ, replay store, logging store, and valid fallback policy

For every training session:

Random simulator parameters: demand, supply and competitor.

For every interval of time t:

Observe the state of the market and predict features: s_t ← observe market state, forecast features

a_t ← sample from πθ(a | s_t)

Instruct a_safe to apply price caps, change limits, and fairness constraints to a_t.Tell a_safe to apply price caps, change limits and fairness constraints to a_t.

Perform a_safe in simulator.

Observe next state s_(t+1) and reward r_t and constraint indicators

Store transition (s_t, a_safe, r_t, s_(t+1))

Update critic and actor using constrained PPO objective

Evaluate policy on validation scenarios and save eligible checkpoint

Select checkpoint based on a set of pre-registered multi-metric criteria

Test out frozen checkpoint on scenarios and stress cases held out

Only deploy in shadow mode, limited pilot, monitoring and rollback controls.

The student will analyze and discuss the work.The student will analyze and discuss the work.

This paper details a proposed framework and does not provide work of the imagination, or assert that DRL is superior to current systems. Here are some hypotheses that need to be tested. First, one would expect that a long-horizon policy would work better at fulfilling demand when there are high surges in the demand, if it could predict the movement of supply instead of just to the current demand-to-driver ratio. Second, the benefit might be reduced if there are predictable demand and supply, or if repositioning drivers takes time. Third, if a

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

policy is trained with an inaccurate simulator, it might do worse in real life than a simple rule-based baseline.

The comparison, likely will be based on the forecast quality and the consistency of passenger and driver reaction. Forecast errors may result in a policy that overprices a zone that does not actually have the demand, and underprices a zone that has insufficient supply. The key is to make minimal changes in a strong policy when uncertainty is high, and to test the policy on the down side, rather than the middle ground. In case the agent continuously jumps back and forth among consecutive price tiers, then the action smoothing or volatility cost should be explicitly introduced.

However, revenue is not enough to demonstrate success. An increase in the price can increase revenue per completed trip, but it can also decrease the number of completed trips, the time passengers spend waiting for a ride, or the burden it puts on passengers who have other options for transportation. Hence, results should be taken as a set of trade-offs. If one policy is better at enhancing the margin but is worse on the affordability/service access aspect, then a policy judgment must be made, not a purely technical statement that the policy is optimal.

The perceived benefit may also change based on competitor reaction. A competitor that can match price hikes could cancel out the draw shifting impact and leave customers with price hikes throughout the market. If a competitor is bidding down, it may cause an agent with a revenue mentality to raise prices or promotions in a costly cycle. Delayed and asymmetric reactions should be considered when designing stress tests and rewards should not be designed to tacitly coordinate or match competitor prices without independent justification.

7.1 Ablation and sensitivity analysis

The following should be one by one removed from the ablation experiments: forecasts, competitor-response estimates, fairness penalties, volatility penalties, and parameter sharing across zones. For sensitivity analysis, the weights of the rewards should be changed and the elasticity assumptions should be changed. If the policy changes significantly with only small weights changes, this may not be a

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

Journal of Technological Information, Management & Engineering Sciences (2020),1(2), 104-121

good system, and this system may not be suitable for autonomous deployment. A helpful outcome could be that a less complex constrained policy is more stable, understandable and dependable than a more complex DRL model.

7.2 Validity threats

There are threats to internal validity due to simulator misspecification, imperfect calibration, and differences in information available to competing baselines. Limited external validity if results are generalizable from one city, season or regulatory context to another. The selection bias in the observed response to prices may be due to historical reasons. Frequent testing can also cause overfitting of the final test set. These risks can be mitigated by chronological splits, multiple cities and seasons, pre-specified analyses, independent simulator validation and prospective shadow evaluation.

8. Ethical, Regulatory and Operational considerations

8.2 Impacts on livelihoods

An evaluation of a pricing policy should be conducted based on differential impacts on neighborhoods and time, especially in situations where there are few alternatives. Average prices can mask uneven service price trends or price spikes. Acceptable disparity thresholds should be established within a process of governance, and the reasons for observed disparities should be discussed and documented and exceptions identified. Fairness does not mean the same price in all zones, it means differences are measured to clear and fair criteria and do not cause undue harm.

8.2 Transparency and contestability

Fares for users should be clearly explained before accepting a ride, and if applicable, material pricing factors should be explained. The system must maintain the quoted fare for the agreed validity period and not impose an unannounced surcharge on the fare after it has been accepted, unless such a surcharge has been announced and is legally allowed. Error detection can be accomplished through complaint channels and internal review procedures that can provide additional information to metrics. Explanations should not reveal the

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

competitive information or give an idea of a certainty for the demand of an inherently uncertain product.

8.3 Privacy and data governance

The system should only store data necessary for pricing and evaluation, and should always be aggregated, access should be limited and schedules set for data retention. Personal information that could be sensitive should not be a direct pricing feature. Fairness audits of protected or sensitive groups should be planned to include sufficient legal review, access controls and re-identification protection. The training data, simulator assumptions, policy versions, and evaluation reports should be recorded for the purpose of reproducing results.

8.4 Deployment controls

A responsible rollout agenda should be implemented gradually: offline evaluation and then shadow mode (no effect on live fares), then a small-scale pilot with limits, and finally mass-scale deployment after an independent review. The team should identify the rollback conditions prior to launch, as they may be, abnormal price spread, cancelation, poor wait times, guardrail failures, or unexplained geographic differences. There should be a focus on both technical and customer metrics. If there is a change in the reward, in the state features or in the policy architecture, the validity should be renewed.

8.5.1 Health and safety of others and the property

Prices systems should be examined to ensure conformance of the rules of consumer protection, transport, privacy, competition and the rules on emergency pricing. Competitor data should be used from legitimate and legal sources. A reinforcement-learning agent should not be allowed to collude with competing systems, or to set design goals that are intended to deplete the competition. Legal and compliance review should be jurisdiction-specific as there are different rules in different jurisdictions for surge pricing and disclosure.

9. Restrictions and Further Study

The proposed framework has some drawbacks. First, the evaluation of a transport disruption via simulation cannot fully capture real passenger behaviour, driver heterogenicity and the social context of a transport disruption. Secondly, weights

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

reward represent normative judgments about balancing profit, access and service quality. Third, competitor behaviour may be partially hidden and a learned response model can become out-of-date. Fourth, a common policy might not reflect local circumstances in cities with varying travel habits, regulations and infrastructure.

Future research will include testing offline reinforcement learning approaches that do not rely on risky online exploration, causal approaches to estimating the price elasticity, and distributionally robust approaches that consider uncertainty regarding demand and competitor response. Multi-objective reinforcement learning may reveal a new space of trade-offs, not a space that was obscured by one scalar reward. Other studies should additionally explore interpretable policies, human-in-the-loop overrules, fairness metrics that are suitable for transport access, and future research in a variety of cities. Field evaluation independent and should report the results as carefully as positive outcomes.

10. Conclusion

This paper suggests a hybrid deep reinforcement learning approach for competitive ride-sharing platforms that can be implemented in dynamic pricing. The framework is essentially a sequential problem of marketplace control, incorporating demand and supply forecasts, an actor–critic policy, competitor-response scenarios, hard operational constraints, fairness monitoring, and staged deployment. It has an evaluation protocol that compares DRL to fixed pricing, rule-based surge pricing, contextual bandits and model-predictive control in terms of financial, operational, customer, driver, fairness, and robustness outcomes.

The main message is methodological: DRL is not a ready-made decision-support tool, it should be evaluated in the context of realistic uncertainty and strong safeguards to establish its value, not simply taken as an automatic substitute for existing pricing rules. The system should be successful in increasing the coordination of the market place, while maintaining the following: affordability, transparency, privacy, respect for competition law, and reliability of services. It is only through offline validated results, independent review, controlled pilots and

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.

ongoing monitoring that the appropriate use of the proposed approach for a specific platform and jurisdiction can be confirmed.

References

Busoniu, L., Babuska, R., De Schutter, B. (2008). A thorough survey of multiagent reinforcement learning. IEEE Transactions on Systems, Man, Cybernetics, Part C: Applications and Reviews, 38(2) 156-172.

Chen, M. K., & Sheldon, M. (2016). Dynamic pricing in a labour market: Surge pricing and flexible working on Uber. In Proceedings of the 2016 ACM Conference on Economics and Computation (EC '16), 2016.

den Boer, A. V. (2015). Historical background, recent research, and future directions in dynamic pricing and learning. A. J. Young: A survey of operations research and management science, Vol. 20, No. 1, pp.1-18

Fiez, T., & Ratliff, L. J. (2019). A supermartingale analysis of push-pull algorithm for distributed optimization in networks. Background on distributed learning (use of multi-agent learning concepts should be adjusted for the marketplace).

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Survey of reinforcement learning. Journal of Artificial Intelligence Research, 4, 237-285.

Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Pérolat, J., Silver, D., & Graepel, T. (2017). Game-theoretic perspective on multiagent reinforcement learning. Improved Neural Information Processing Systems, 30.

The film "Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. " (2017). Algorithms called proximal policy optimization algorithms. arXiv:1707.06347.

Sutton, R. S., & Barto, A. G. (2018). An introduction to reinforcement learning. MIT Press.

Taylor, T. A. (2018). On-demand service platforms. R. Manufacturing & Service Operations Management 20(4), 704-720.

Varian, H. R. (1980). A model of sales. The American Economic Review, 70(4), 651–659.Wang, Z., & Ye, J. (2018).

[1] Department of Computer Science, University of Central Punjab, Lahore, Pakistan
[2] Department of Software Engineering, University of Management and Technology, Lahore, Pakistan

Copyright (c) 2020 Author(s)
This work is licensed under a Creative Commons Attribution 4.0 International License.