Reinforcement Learning for Intensity Control: An Application to Choice-Based Network Revenue Management

Huiling Meng
CUHK

Ningyuan Chen
University of Toronto

Xuefeng Gao
CUHK

Abstract

Intensity control is a type of *continuous-time* dynamic optimization problems with important applications in Operations Research. In this study, we adapt the reinforcement learning framework to intensity control using choice-based network revenue management as a case study, which is a classical problem in revenue management that features a large state space, a large action space and a continuous time horizon. We show that the inherent discretization from jump points, a key feature of intensity control, eliminates the need to discretize the time horizon upfront, which was believed to be necessary because most reinforcement learning algorithms are designed for discrete-time problems. This facilitates computation and significantly reduces discretization error. We lay the theoretical foundation for policy evaluation and develop policy-gradient-based actor-critic algorithms for intensity control. A comprehensive numerical study demonstrates the benefit of our approach versus state-of-the-art benchmarks.

1 Introduction

Many dynamic optimization problems in Operations Research are intensity control problems, featuring *continuous time* and a discrete state space. Two notable areas are control problems in queueing [2, 3] and dynamic pricing/assortment problems in revenue management [5, 11]. Despite extensive study, most problems remain challenging to solve in practice because the state space—such as combinations of remaining inventory—can be extremely large, rendering optimal solutions intractable.

Reinforcement learning (RL) provides a computational framework for general dynamic optimization problems. However, standard RL algorithms are designed for discrete-time Markov decision processes (MDPs). A common practice for applying RL to continuous-time problems is to first discretize the time horizon uniformly. This approach presents a difficult trade-off: a fine grid is needed for accuracy but leads to prohibitive computational costs and numerical instability. Even worse, *there are no guidelines for selecting an appropriate grid size*, and the performance of RL algorithms can be highly sensitive to this choice [12].

In this paper, via the classical application of choice-based network revenue management (NRM) (see [11] for a recent review), we provide a framework to implement RL algorithms for intensity control problems, *without the need for upfront time discretization*. The key insight is to leverage the event-driven discretization induced by the jump times of the system's sample paths under a given policy. For example, in the focused application, the system state (inventory) changes and the reward is generated only when a customer arrives, and these random yet finite arrival times provide an adaptive discretization of the time horizon.

The main contributions of our study are threefold. First, we adapt policy evaluation (PE) and policy gradient (PG) in the standard discrete-time RL framework to the *continuous time*, and combine

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: MLxOR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making.

them to develop model-free actor-critic algorithms. With the *adaptive discretization* procedure, our algorithm can be implemented with reduced approximation errors, while maintaining numerical stability and computational efficiency. Second, we establish a theoretical underpinning for the continuous-time PE and PG methods by extending the *martingale approach*—originally developed for RL in controlled diffusion processes [6, 7]—to intensity control problems with discrete states. Third, we conduct numerical experiments demonstrating that the proposed actor-critic algorithm outperforms state-of-the-art benchmarks, including the CDLP policy [8] and the ADP policy [16], despite being a model-free approach. In particular, one of the experiments features a state space of size 11^{100} and an action space of size 2^{200}, demonstrating the scalability and practical applicability of our framework.

There is extensive literature related to this work. The choice-based NRM problems have been studied by [4, 13, 17, 8, 16, 15], most of which focus on providing efficient approximate solution algorithms, with provable performance guarantees. While some studies including [16, 9] design algorithms based on approximate dynamic programming (ADP), which is an important concept and approach in RL, they focus on the discrete-time formulation and value function approximations. In contrast, we study the continuous-time formulation and general RL algorithms including exploration and the policy gradient method. Methodologically, our paper builds on recent advances in continuous-time RL with *continuous* state and action spaces [14, 6, 7], where dynamics are modeled by controlled diffusion processes and rewards accrue continuously. In contrast, we focus on intensity control of point processes with *discrete* states and actions, piecewise-constant paths, and rewards collected only at jump times. This leads to significant differences in our theoretical analysis and algorithm design. Moreover, the classical uniformization method for infinite-horizon continuous-time MDPs [10, Chapter 11] is not directly applicable to our finite-horizon intensity control problem.

2 Problem Formulation and Methodology

We consider a network revenue management problem over a finite selling horizon $[0, T]$. The firm controls m resources with initial inventory $c = [c_1, \ldots, c_m]^\top$, and it offers n products at fixed prices $p = [p_1, \ldots, p_n]^\top$, each consuming certain resources upon sale. Let $A = [a_{ij}]_{m\times n}$ be the consumption matrix, where a_{ij} is the amount of resource i used by selling one unit of product j. Consumer arrivals follow a Poisson process with rate λ. Upon arrival, given the product assortment $S \subset \{1, 2, \ldots, n\}$ offered by the firm at the moment, the customer purchases product $j \in S$ with probability $P_j(S)$, yielding a revenue of p_j for the firm. With probability $P_0(S)$, the customer makes no purchase ($j = 0$) and the firm earns no revenue. The choice probabilities $P_j(S) \in [0, 1]$ satisfy $\sum_{j\in S\cup 0} P_j(S) = 1$ and $P_j(S) = 0$ for $j \notin S$, although they may be unknown to the RL algorithms. The firm's decision problem is to find a dynamic policy that offers assortment S_t at time t that maximizes the expected total revenue over the selling horizon $[0, T]$.

We formulate the problem in the language of optimal intensity control. The system state $X_t = [X_{1,t}, \ldots, X_{m,t}]^\top$ is the remaining inventory levels, with state space denoted by $\mathcal{X}$. The action S_t is the assortment offered by the firm at time t, with action space denoted by $\mathcal{A}$. Let $N_t = (N_{1,t}, \ldots, N_{n,t})^\top$ be a vector of controlled Poisson processes with intensities $(\lambda P_1(S_t), \ldots, \lambda P_n(S_t))$, representing the cumulative number of the n products sold by time t. The remaining inventory is then given by $X_t = c - AN_t$.

In this study, we focus on policy-based RL and consider a class of admissible randomized Markov policies Π. Each policy $\pi \in \Pi$ is a mapping from $[0, T] \times \mathcal{X} \times \mathcal{A}$ to $[0, 1]$, and $\pi(\cdot \mid t, x)$ is a probability distribution on $\mathcal{A}$. Under policy π, when a customer arrives, the firm draws an assortment from the distribution $\pi(\cdot \mid t, x)$ given the current time t and state x. Hereafter, we use the superscript π to denote processes generated under policy π; namely, X_t^π, S_t^π and N_t^π.

To encourage exploration, we follow [7] and consider the entropy-regularized value function

$$J(t, x; \pi) = \mathbb{E}\bigg[\int_{(t,T]} p^\top \mathrm{d}N_s^\pi + \gamma \int_t^T \mathcal{H}(\pi(\cdot \mid s, X_{s-}^\pi))\mathrm{d}s \mid X_t^\pi = x\bigg],$$

where the first term corresponds to the cumulative reward accrued over $(t, T]$ in the classical setting, while the second term is an entropy bonus $\mathcal{H}(\pi(\cdot \mid t, x)) = -\sum_{S\in\mathcal{A}} \pi(S \mid t, x) \log \pi(S \mid t, x)$. The parameter $\gamma \geq 0$ serves as a temperature that controls the degree of exploration.

The task of RL is to find a policy $\pi^* \in \Pi$ that attains $J^*(t, x) = \sup_{\pi\in\Pi} J(t, x; \pi)$ for all t, x. We address this task by focusing on two model-free objectives: policy evaluation (PE) and policy improvement via the policy gradient (PG) method.

Policy Evaluation

For a given policy π, PE aims at the employment of a (numerical) procedure to determine $J(t,x;\pi)$ as a function of (t,x) without any knowledge of the customer arrival rate or the choice probabilities. It is often achieved through function approximations, where $J(t,x;\pi)$ is approximated by a parametric family of functions $\{J^\theta(t,x):\theta\in\Theta\}$.

While Monte Carlo methods are a standard approach for offline policy evaluation in discrete time RL, this section develops a *continuous-time counterpart* to the gradient Monte Carlo method. We achieve this by formulating a valid loss function in the continuous-time setting as

$$L(\theta)=\frac{1}{2}\mathbb{E}\Big[\int_0^T\Big(\int_{(t,T]}p^\top \mathrm{d}N_s^\pi+\gamma\int_t^T\mathcal{H}(\pi(\cdot\mid s,X_{s-}^\pi))\mathrm{d}s-J^\theta(t,X_t^\pi)\Big)^2\mathrm{d}t\Big],$$

which tracks the error between the realized reward along sample paths and the estimated value function. We first note that by taking the derivative of $L(\theta)$ we obtain a continuous-time updating rule for θ which is parallel to that of gradient Monte Carlo. More importantly, we establish the validity of our proposed loss function from a theoretical standpoint, as formalized in Theorem 1, which states that minimizing $L(\theta)$ is equivalent to minimizing the mean-squared value error (MSVE). It is worth noting that, although the MSVE is considered an ideal loss function, minimizing it does not directly result in a feasible algorithm since the true function $J(\cdot,\cdot;\pi)$ is not known.

Theorem 1 *It holds that* $\arg\min_\theta L(\theta)=\arg\min_\theta\frac{1}{2}\mathbb{E}[\int_0^T|J(t,X_t^\pi;\pi)-J^\theta(t,X_t^\pi)|^2\mathrm{d}t]$.

Note that the integrals in $L(\theta)$ either amount to a finite sum of values at the jump points, such as $\int_{(t,T]}p^\top\mathrm{d}N_s^\pi$, or share a common general form $\int_0^T z(t,X_t^\pi)\mathrm{d}t$. For a realized state sample path $\{x_t:t\in[0,T]\}$ with jump times $\{\tau_l\}_{l=1}^L$, the integral can be written as $\int_0^T z(t,x_t)\mathrm{d}t=\sum_{l=0}^L\int_{\tau_l}^{\tau_{l+1}}z(t,x_{\tau_l})\mathrm{d}t$, where $\tau_0\coloneqq 0$ and $\tau_{L+1}\coloneqq T$. In contrast to a pre-specified discretization, this essentially provides an *adaptive scheme* that discretizes the time horizon for each trajectory *without discretization error*. Moreover, if the univariate function $z(t,x_{\tau_l})$ takes a simple form in t, such as a polynomial, the integral $\int_{\tau_l}^{\tau_{l+1}}z(t,x_{\tau_l})\mathrm{d}t$ can be evaluated exactly, without the need for numerical procedures. Even when $z(t,x_{\tau_l})$ is not analytically integrable, the one-dimensional integral can still be approximated with high accuracy using numerical integration.

Policy Gradient

For a given admissible policy, we next seek to improve it using the PG method. In particular, we formulate the method to rely only on observable data and the value function of the current policy. Consider a parametric family of admissible policies $\{\pi^\phi(\cdot\mid\cdot,\cdot):\phi\in\Phi\}$, our objective is to determine $\arg\max_{\phi\in\Phi}J(0,c;\pi^\phi)$, which requires computing the policy gradient $\nabla_\phi J(0,c;\pi^\phi)$.

Theorem 2 *Under mild assumptions on π^ϕ, the policy gradient admits the following representation:*

$$\begin{aligned}\nabla_\phi J(0,c;\pi^\phi)=\mathbb{E}\Big[&\sum_{j=1}^n\int_{(0,T]}\nabla_\phi\log\pi^\phi(S_t^{\pi^\phi}\mid t,X_{t-}^{\pi^\phi})[J(t,X_{t-}^{\pi^\phi}-A^j;\pi^\phi)-J(t,X_{t-}^{\pi^\phi};\pi^\phi)+p_j]\mathrm{d}N_{j,t}^{\pi^\phi}\\&+\gamma\int_0^T\nabla_\phi\mathcal{H}(\pi^\phi(\cdot\mid t,X_{t-}^{\pi^\phi}))\mathrm{d}t\Big].\end{aligned}\tag{1}$$

With $J(t,x;\pi^\phi)$ approximated by $J^{\theta^*}(t,x)$ obtained from the PE step, all the terms inside the expectation in (1) become computable from observed trajectories under the current policy π^ϕ. When dealing with the samples, the computation involves a finite sum of values at the jump points and an integral of the entropy gradient, where the latter is addressed via the adaptive discretization scheme discussed in the PE step to reduce the approximation error. Unlike the policy gradient for diffusion processes [7], whose implementation requires an artificial discretization of time and successive action randomization at all grid points, our policy gradient formula benefits from the inherent structure of the jump process and *only requires randomized actions at customer arrival times*.

By combining the PE and PG modules in an iterative manner, we obtain the actor-critic algorithms. In each iteration, based on the current parameters (θ,ϕ), trajectories are generated under the current policy π^ϕ. This collected data is then used to update the value parameters θ by minimizing the loss function $L(\theta)$, and the policy parameters ϕ via the policy gradient (1). This process is then repeated with the updated parameters.

3 Numerical Experiments

In this section, we evaluate the performance of our proposed algorithms on medium- and large-scale choice-based network revenue management problems. We consider three different combinations of value and policy approximations as follows.

- *Linear-Pair*: $J^\theta(t,x) = \sum_{r=0}^{d} \theta_{(0,r)}(1-\frac{t}{T})^r + \sum_{i=1}^{m}(\sum_{r=0}^{d} \theta_{(i,r)}(1-\frac{t}{T})^r)x_i$ and $\pi^\phi(S \mid t,x) = \frac{\exp\{\frac{1}{\gamma}\sum_{r=0}^{d}(\sum_{1\le j,\, j'\le n} \phi_{(j,j',r)}\delta_j(S)\delta_{j'}(S))(1-\frac{t}{T})^r\}}{\sum_{\bar{S}\in\mathcal{A}(x)}\exp\{\frac{1}{\gamma}\sum_{r=0}^{d}(\sum_{1\le j,\, j'\le n} \phi_{(j,j',r)}\delta_j(\bar{S})\delta_{j'}(\bar{S}))(1-\frac{t}{T})^r\}}$, where $\delta_j(S)=1$ if $j\in S$ and 0 otherwise.
- *Linear-RO*: $J^\theta(t,x)$ is the same as in Linear-Pair, $\pi^\phi(S^{[k]} \mid t,x) = \frac{\exp\{\frac{1}{\gamma}\sum_{l=0}^{d}\phi_{(k,l)}(1-\frac{t}{T})^l\}}{\sum_{\bar{k}=1}^{K}\exp\{\frac{1}{\gamma}\sum_{l=0}^{d}\phi_{(\bar{k},l)}(1-\frac{t}{T})^l\}}$, where $S^{[1]},\ldots,S^{[K]}$ are all the revenue-ordered assortments [1].
- *2-NNs*: Two fully connected neural networks $J^\theta : \mathbb{R}^{1+m} \mapsto \mathbb{R}$ and $W^\phi : \mathbb{R}^{1+m} \mapsto \mathbb{R}^n$ are adopted. The former serves as the value approximator, while the policy approximator is constructed from the latter via $\pi^\phi(S \mid t,x) = \prod_{j\in S}\frac{e^{\frac{1}{\gamma}[W^\phi(t,x)]_j}}{1+e^{\frac{1}{\gamma}[W^\phi(t,x)]_j}}\prod_{j\notin S}\frac{1}{1+e^{\frac{1}{\gamma}[W^\phi(t,x)]_j}}$.

Our benchmarks include UNIF-RAND, GREEDY, CDLP [8] and ADP [16]. The UNIF-RAND policy selects each available assortment with equal probability, and the GREEDY policy always selects the assortment with the highest expected revenue. Since the ADP method operates in discrete time, we use ADP-Δt to denote the ADP policy with discretization step size Δt.

A Medium-Sized Airline Network

We consider an airline network with 6 flight legs (resources) and 9 itineraries (products). The selling horizon is $T = 200$ and the initial inventory is $c = [12, 20, 16, 20, 12, 16]^\top$. Customer choice probabilities are assumed to follow the MNL model [8].

As summarized in Table 1, our algorithm performs well under three different approximation schemes. The Linear-Pair policy achieves the best results, outperforming all benchmarks. Although the performance of ADP-1 is competitive, ADP policies are highly sensitive to the level of time discretization. For instance, with a suboptimal discretization of $\Delta t = 0.5$, our algorithm surpasses ADP-0.5 by a margin of up to 17.2%. This further demonstrates the advantage of our continuous-time framework, as it avoids issues with upfront time discretization.

A Large Network

We consider a network with $m = 100$ resources and $n = 200$ products. Each resource has an initial capacity of $c_i = 10$ and the selling horizon is set to $T = 2,000$. This configuration results in a problem with a state space of size 11^{100} and an action space of size 2^{200}, mirroring the scale of real-world problems.

At this scale, the Linear-Pair and Linear-RO methods become computationally infeasible due to the enormous policy space. Similarly, the ADP method is intractable. Therefore, we only implement the 2-NNs policy and other benchmarks. In this example, the CDLP method provides a theoretical upper bound of 197,785 on the optimal expected revenue. Table 2 shows that the 2-NNs policy exhibits a small performance gap of 0.13% from the upper bound, indicating that it nearly achieves the optimal solution. The computational time for the 2-NNs experiment was approximately 33.6 hours, which is reasonable given the scale of the example.

Table 1: Simulation results for medium network

Policy	Avg. Rev.	Rel. Perf. (%)
Linear-Pair	677.356	100.00
Linear-RO	673.595	99.44
2-NNs	676.369	99.85
UNIF-RAND	595.603	87.93
GREEDY	605.402	89.38
CDLP	651.294	96.15
ADP-1	675.566	99.74
ADP-0.5	560.899	82.81
ADP-0.1	640.854	94.61

Table 2: Simulation results for large network

Policy	Avg. Rev.	Gap (%)
2-NNs	197,533	0.13
UNIF-RAND	141,260	28.58
GREEDY	161,205	18.49
CDLP	173,502	12.28

References

[1] Gerardo Berbeglia and Gwenaël Joret. Assortment optimisation under a general discrete choice model: A tight analysis of revenue-ordered assortments. *Algorithmica*, 82:681–720, 2020.

[2] Pierre Brémaud. *Point Processes and Queues: Martingale Dynamics*. Springer-Verlag, New York, 1981.

[3] Hong Chen and David D Yao. Optimal intensity control of a queueing system with state-dependent capacity limit. *IEEE Transactions on Automatic Control*, 35(4):459–464, 1990.

[4] Guillermo Gallego, Garud Iyengar, Robert Phillips, and Abhay Dubey. Managing flexible products on a network. *Working Paper*, 2004.

[5] Guillermo Gallego and Garrett Van Ryzin. A multiproduct dynamic pricing problem and its applications to network yield management. *Operations research*, 45(1):24–41, 1997.

[6] Yanwei Jia and Xun Yu Zhou. Policy evaluation and temporal-difference learning in continuous time and space: A martingale approach. *Journal of Machine Learning Research*, 23(154):1–55, 2022.

[7] Yanwei Jia and Xun Yu Zhou. Policy gradient and actor-critic learning in continuous time and space: Theory and algorithms. *The Journal of Machine Learning Research*, 23(1):12603–12652, 2022.

[8] Qian Liu and Garrett Van Ryzin. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management*, 10(2):288–310, 2008.

[9] Yuhang Ma, Paat Rusmevichientong, Mika Sumida, and Huseyin Topaloglu. An approximation algorithm for network revenue management under nonstationary arrivals. *Operations Research*, 68(3):834–855, 2020.

[10] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[11] Arne K Strauss, Robert Klein, and Claudius Steinhardt. A review of choice-based revenue management: Theory and methods. *European journal of operational research*, 271(2):375–387, 2018.

[12] Corentin Tallec, Léonard Blier, and Yann Ollivier. Making deep Q-learning methods robust to time discretization. In *International Conference on Machine Learning*, pages 6096–6104. PMLR, 2019.

[13] Kalyan Talluri and Garrett Van Ryzin. Revenue management under a general discrete choice model of consumer behavior. *Management Science*, 50(1):15–33, 2004.

[14] Haoran Wang, Thaleia Zariphopoulou, and Xun Yu Zhou. Reinforcement learning in continuous time and space: A stochastic control approach. *J. Mach. Learn. Res.*, 21(198):1–34, 2020.

[15] Dan Zhang. An improved dynamic programming decomposition approach for network revenue management. *Manufacturing & Service Operations Management*, 13(1):35–52, 2011.

[16] Dan Zhang and Daniel Adelman. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science*, 43(3):381–394, 2009.

[17] Dan Zhang and William L Cooper. Revenue management for parallel flights with customer-choice behavior. *Operations Research*, 53(3):415–431, 2005.