A Robust Test for the Stationarity Assumption in Sequential Decision Making

Jitao Wang [1] **Chengchun Shi** [2] **Zhenke Wu** [1]

Abstract

Reinforcement learning (RL) is a powerful technique that allows an autonomous agent to learn an optimal policy to maximize the expected return. The optimality of various RL algorithms relies on the stationarity assumption, which requires time-invariant state transition and reward functions. However, deviations from stationarity over extended periods often occur in real-world applications like robotics control, health care and digital marketing, resulting in suboptimal policies learned under stationary assumptions. In this paper, we propose a model-based doubly robust procedure for testing the stationarity assumption and detecting change points in offline RL settings with certain degree of homogeneity. Our proposed testing procedure is robust to model misspecifications and can effectively control type-I error while achieving high statistical power, especially in high-dimensional settings. Extensive comparative simulations and a real-world interventional mobile health example illustrate the advantages of our method in detecting change points and optimizing long-term rewards in high-dimensional, non-stationary environments.

1. Introduction

Reinforcement learning (RL, Sutton & Barto, 2018) has become increasingly popular in various fields, including video games (Mnih et al., 2015; Shao et al., 2019), robotics (Kober et al., 2013; Kilinc & Montana, 2022), mobile health (Liao et al., 2020; Luckett et al., 2020; Shi et al., 2022b) and ridesharing (Xu et al., 2018; Shi et al., 2022a). However, the fundamental online learning paradigm of RL, which requires the agent to iteratively and extensively collect experience by interacting with the environments, makes these algorithms inapplicable to real-world domains such as autonomous driving (Shalev-Shwartz et al., 2016; Kiran et al., 2021) and precision medicine (Murphy, 2003; Qian & Murphy, 2011; Zhu et al., 2017; Raghu et al., 2017; Wang et al., 2018; Kosorok & Laber, 2019; Qi et al., 2020; Liu et al., 2020; Cai et al., 2021; Nie et al., 2021; Li et al., 2022), where online data collection can be costly and risky. To address this challenge, offline RL algorithms have emerged in recent years. These algorithms allow an offline RL agent to learn from a static dataset containing transition data collected by a behavior policy without additional online interactions with the environment (Fujimoto et al., 2019; Kumar et al., 2019; Levine et al., 2020; Jin et al., 2021; Uehara & Sun, 2021; Xie et al., 2021; Shi et al., 2022c; Bai et al., 2022; Rezaeifar et al., 2022; Liao et al., 2022; Zhou et al., 2022; 2023).

Motivations Most of the state-of-art RL algorithms rely on the assumption that the underlying Markov Decision Process (MDP) is stationary, requiring the system dynamics to be temporally invariant. However, this assumption can be highly restrictive in many real-world problems, such as traffic signal control (Padakandla et al., 2020; Alegre et al., 2021) and mobile health (Liao et al., 2020; Li et al., 2022). In traffic control, for example, traffic flow rates in different lanes can be distinctive between peak and off-peak hours, and queue lengths can vary significantly depending on different traffic patterns. Since traffic flow patterns change dynamically over time, the stationarity assumption is likely to be violated. Directly applying an RL algorithm derived under stationarity assumptions may result in a suboptimal policy, potentially causing longer waiting time and traffic congestion.

Another motivating example is the Intern Health Study (IHS, Sen et al., 2010; NeCamp et al., 2020), a one-year mobile health-based mirco-randomized trial targeting first-year training physician in the United States. One goal of this study is to investigate the effectiveness of just-in-time push notifications, including tips and life insights, on improving interns' physical activity, sleep duration and mental health, while minimizing user burden and expense. The study collected data on interns' daily step count and sleep duration through wearable devices (i.e., Fitbit or Apple Watch) and self-reported mood score through daily survey. Non-stationarity is a serious issue in this study, due to the waning intervention effect (i.e., the effect of intervention de-

[1]Department of Biostatistics, University of Michigan, Ann Arbor [2]Department of Statistics, London School of Economics and Political Science. Correspondence to: Chengchun Shi <c.shi7@lse.ac.uk>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

creases or change direction over time). Such phenomena are commonly seen in interventional mobile health applications (Hamari et al., 2014; Shcherbina et al., 2019; Klasnja et al., 2019; NeCamp et al., 2020). Ignoring the non-stationary nature of the interventional effects may lead to learning ineffective or even harmful policies that send inopportune prompts, overburdening users and leading to increased user attrition.

There are two additional technical challenges in dealing with modern sequential decision making problems. First, with the fast development of new information technology, the state vector is typically high-dimensional (Mnih et al., 2015; Arulkumaran et al., 2017; Plaat et al., 2020; Kiran et al., 2021), which poses a challenge to classical nonparametric methods due to the curse of dimensionality. Second, the real-world data can be very complex when the underlying state transition and reward function is highly complex (Mnih et al., 2015; Silver et al., 2018; Mahmood et al., 2018). Although linear function approximation often has sound theoretical guarantees, it has limited approximation capacity in handling nonlinear and complicated environments. On the contrary, modern machine learning methods have achieved significant empirical success in dealing with nonlinearity and high-dimensionality by striking a balance between regularization bias and overfitting. However, the resulting estimator's asymptotic distribution is often difficult to establish, hampering tractable statistical inference. To address these challenges, in this paper, we develop a novel offline procedure for testing the stationarity in the MDP framework, bridging the gap between modern ML/RL methods and valid statistical inference.

Related work Non-stationary RL has been studied extensively in recent year (Da Silva et al., 2006; Auer et al., 2008; Gajane et al., 2018; Cheung et al., 2020; Igl et al., 2020; Padakandla et al., 2020; Fei et al., 2020; Mao et al., 2021; Chen et al., 2021; 2022; Wei et al., 2022; Chen & Luo, 2022; Feng et al., 2022). Several approaches have been proposed to address non-stationarity in RL. For example, Gajane et al. (2018) and Cheung et al. (2020) proposed model-based solutions using slide windows to estimate the MDP models to handle non-stationarity. Auer et al. (2008), Ortner et al. (2020) and Mao et al. (2021) incorporated a "forgetting" strategy by periodically restarting the learning algorithm with estimators built on newly collected data. However, these methods typically assume a discrete state and action space, limiting their applicability in real-world applications involving continuous states and actions. Jin et al. (2018) and Jin et al. (2021) considered the episodic MDP setting and tackle non-stationarity by learning a separate policy for each time point. However, their single-time-based approaches may suffer from low sample efficiency when encountering MDPs with a certain degree of homogeneity (i.e., transition and reward functions are homogeneous across some episodes and horizons), as each policy is learned per time without borrowing information from other time points. Moreover, their methods may encounter difficulties in certain studies with small number of episodes and long horizons (Marling & Bunescu, 2020) due to the limited sample size available per time.

Another related line of research focus on change point detection algorithms as a means to address non-stationarity. The central idea of these detection-based methods involves first detecting environmental changes by testing the stationarity assumption and subsequently applying policy learning algorithms to the stationary segments of the data. Therefore, these algorithms typically require a certain degree of homogeneity in terms of MDP models across episodes and horizons (e.g., piecewise stationary MDP). Several methods have been proposed to test the stationarity assumption of an MDP. Among those available, Hadoux et al. (2014) proposed to use a CUmulative SUM (CUSUM) sequential statistical test (Basseville et al., 1993) for change point detection in MDPs. However, this approach requires prior knowledge of the true MDP models, making it inapplicable in many RL problems where the system dynamics are unknown; Padakandla et al. (2020) developed a Context Q-learning (Context QL) built upon the online Dirichlet change point (ODCP) algorithm (KJ et al., 2021). This approach applies the Dirichlet likelihood test to the state-reward-next-state triplet to test the stationarity assumption. Li et al. (2022) proposed a CUSUM-type test statistic to test the stationarity of the optimal Q-function instead of directly tackling the potential changes in the state transition or reward function. Our method is closely related to their method, although there are a few key distinctions. First, their approach is model-free in the sense that they focused on the Q-function whereas our method is model-based, which directly tests the underlying components of the MDP model. Second, their method requires linear function approximation, which can be less favorable in high-dimensional complex environments. On the contrary, our method harnesses the power of modern ML algorithms to handle high-dimensionality.

Our contributions Methodologically, we propose a novel offline procedure to test the stationarity of MDPs in high-dimensional settings. The proposed test is doubly robust in the sense that it controls the type-I error as long as either the transition function or the marginal state-action distribution function is correctly specified – a key property that allows us to utilize modern ML methods for hypothesis testing. Our proposal shares similar spirits with the double ML (DML) method for causal inference (Chernozhukov et al., 2018), the double RL (DRL) method for policy evaluation (Kallus & Uehara, 2022), and the double generative adversarial networks algorithm for conditional independence testing (Shi et al., 2021b). These methods are developed via the classical semiparametric theory in the statistics literature (Tsiatis,

2006). Nonetheless, it is highly nontrivial to derive a doubly robust test procedure for the stationarity assumption, due to the existence of absolute value operator in the CUSUM statistic which makes the target parameter non-pathwise differentiable (Kennedy, 2022). Therefore, existing technical tools are not directly applicable to constructing the test statistic. Theoretically, we proved the size (i.e., type-I error control) and double robustness property of the proposed test under a general bidirectional asymptotic framework which allows either the number of trajectories or the length of horizon to diverge to infinity. Empirically, we demonstrate the efficacy of our test in terms of change point detection and policy learning in non-stationary environments through four numerical studies and a real-world health example.

2. Preliminaries

2.1. Data and Problem Formulation

Consider a Markov Decision Process $\mathcal{M} \equiv (\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S}\times\mathcal{A}\times\mathcal{S} \to \mathbb{R}_{\geq 0}$ is the transition function, $R : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ is the reward function and $\gamma \in [0, 1)$ is the discount factor. The objective of RL is to learn a policy $\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_{\geq 0}$, which maximizes the expected discounted cumulative reward: $\mathcal{J}(\pi) = \mathbb{E}\left(\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right)$, where s_0 is sampled from an initial state distribution $\rho_0 : \mathcal{S} \mapsto \mathbb{R}_{\geq 0}$ (Sutton & Barto, 2018).

We consider testing stationarity of MDPs in an offline setting, where the dataset is fixed and collected by some behavior policies, without further interaction with the environment. The dataset can be summarized as $\{(S_{i,j}, A_{i,j}, R_{i,j}, S_{i,j+1})_{1\leq i\leq N, 0\leq j\leq T-1}\}$ where $(S_{i,j}, A_{i,j}, R_{i,j}, S_{i,j+1})$ is the experience tuple from subject i at time point j, N is the number of subjects and T is the number of time points. A change point $t_{cpt} \in [1, 2, \cdots, T-1]$ is defined as the location at which the state transition or reward function changes. We assume all data trajectories are i.i.d. so that the change points (if exist) are homogeneous across all subjects in terms of their locations. In other words, all the subjects have the same number and locations of the change points.

Next, we formally formulate the hypothesis testing problem of interest. Without loss of generality, we include the reward R into the set of the next state S', as both transition and reward function depend on state S and action A. Let $p_t(s'|s, a)$ denote the "new" transition function at time t that combines the transition and reward function defined earlier. In this paper, we focus on testing the null hypothesis $H_0 : \{p_t(s'|a, s)\}_t$ *are homogeneous over time* where $t < T$, that is, $p_t(s'|a, s)$ is a constant function of time index t. The alternative hypothesis is given by $H_a : \{p_t(s'|a, s)\}_t$ *has at least one change point*.

To simplify the notation, for any quantity f, let $\widehat{f}$ denote the estimator of f and f^* denote the oracle version of f.

2.2. A Naive Plug-in Estimator

The following quantity forms the basis of the proposed test procedure,

$$\int \Bigg| \int \frac{\sum_{j=t}^{T-1} h(s')p_j(s'|a, s)}{T-t} ds' - \int \frac{\sum_{j=0}^{t-1} h(s')p_j(s'|a, s)}{t} ds' \Bigg| g(a, s)\lambda(da, ds), \quad (1)$$

where g is some reference density function and λ is the Lebesgue or counting measure, depending on whether the state-action pair is continuous or not. The function h in Equation 1 is a test function from a rich function class. There are a variety of choices for h. For example, if $h(s') = r$, we are comparing the difference in expected reward before and after time point t. Alternatively, h can be chosen to depend on the next state as well. We rewrite Equation 1 as follows,

$$\int \Big|\mathbb{E}_{[t,T]}[h(S')|a, s] - \mathbb{E}_{[0,t]}[h(S')|a, s]\Big| g(a, s)\lambda(da, ds), \quad (2)$$

where $\mathbb{E}_{[t_1,t_2]}[h(S')|a, s] = \int h(s')p_{[t_1,t_2]}(s'|a, s)ds'$ for $0 \leq t_1 < t_2 \leq T$ and $p_{[t_1,t_2]}$ is the pooled transition function (or a mixture of transition functions) from time t_1 to $t_2 - 1$,

$$p_{[t_1,t_2]}(S'|s, a) = \frac{1}{t_2 - t_1} \sum_{j=t_1}^{t_2-1} p_j(S'|s, a).$$

Notice that when $t_1 \ll t_2$, it is much easier to accurately estimate the pooled function $p_{[t_1,t_2]}$ than to accurately estimate each individual p_j, due to that we have a larger amount of data in the interval $[t_1, t_2]$.

Under H_0, we have $p_{[t_1,t_2]} = p^*$ for any $0 \leq t_1 < t_2 \leq T$, where p^* denotes the oracle transition function. Therefore, Equation 2 equals 0 for any function h. Under H_a, there exists certain test function h and t such that Equation 2 is strictly positive. It can thus be used to detect non-stationarity. This motivates us to consider the following plug-in estimator for Equation 2,

$$\int \Big|\widehat{\mathbb{E}}_{[t,T]}[h(S')|a, s] - \widehat{\mathbb{E}}_{[0,t]}[h(S')|a, s]\Big| g(a, s)\lambda(da, ds), \quad (3)$$

where $\widehat{\mathbb{E}}_{[t_1:t_2]}$ denotes certain estimator for $\mathbb{E}_{[t_1,t_2]}$ by replacing $p_{[t_1,t_2]}$ with an estimated transition function, denoted as $\widehat{p}_{[t_1,t_2]}$.

Modern ML algorithms (e.g., lasso, neural networks) are particularly well-suited to estimate $p_{[t_1,t_2]}$ even in high-dimensional scenarios. These methods perform well through regularization for variance reduction and balancing regularization bias and overfitting. However, naively plugging these ML estimators into Equation 2 will cause a heavy plug-in bias, which invalids the subsequent testing procedure. On the other hand, kernel smoothers or local polynomial regression with properly chosen bandwidth parameter have small plug-in biases. However, these methods suffer from the curse of dimensionality, which makes them less suitable in high-dimensional settings.

In the next section, we propose a doubly robust estimator for Equation 2 to alleviate the plug-in bias. Specifically, we proposed a sample augmented version of the naive estimator Equation 3 which offers additional protection against potential misspecification of the transition model. The doubly robust property entitles us to enjoy the superior approximation capacity of modern ML methods, as well as to conduct valid statistical inference (e.g., able to control type-I and type-II errors).

3. Method

3.1. A Doubly Robust Estimator

To begin with, we introduce some notations. Define the following random variable

$$\phi_{[t_1,t_2]}(S,A,S';h) = \operatorname{sgn}\left(\Delta(A,S;h,t)\right) \times \left[h(S') - \mathbb{E}_{[t_1,t_2]}[h(S')|A,S]\right] \frac{g(A,S)}{\omega_{[t_1,t_2]}(A,S)},$$

where $\Delta(a,s;h,t) = \mathbb{E}_{[t,T]}[h(S')|a,s] - \mathbb{E}_{[0,t]}[h(S')|a,s]$, $\operatorname{sgn}(\cdot)$ denotes the sign function and $\omega_{[t_1,t_2]}$ is the pooled marginal state-action distribution from time t_1 to t_2-1. Moreover, define

$$\psi_i = \int \left|\Delta(t,h;a,s)\right| g(a,s)\lambda(da,ds) + \frac{1}{T-t}\sum_{j=t}^{T-1} \phi_{[t,T]}(S_{i,j},A_{i,j},S_{i,j+1};h) - \frac{1}{t}\sum_{j=0}^{t} \phi_{[0,t]}(S_{i,j},A_{i,j},S_{i,j+1};h), i=1,\ldots,N.$$

The first line on the right-hand-side corresponds to Equation 2. The last two lines form an augmentation term which is of mean zero when the transition model is correctly specified. When the transition model is misspecified, however, it offers additional protection to reduce the bias of ψ_i resulting from the misspecification. More specifically, we present the double robustness property in the following theorem.

Theorem 3.1 (Double robustness)**.** *Suppose H_0 holds. For any $h \in H$ and $t \in (0, T-1)$, we have*

$$\mathbb{E}\psi_i^* = 0, i = 1,\ldots,N, \tag{4}$$

where ψ_i^* denotes a version of ψ_i with the oracle transition and marginal state-action distribution functions. Moreover, the above equation is doubly robust, i.e., for any $p_{[t,T]}$, $p_{[0,t]}$, $\omega_{[t,T]}$ and $\omega_{[0,t]}$, the following holds as long as either $p_{[t,T]} = p^*_{[t,T]} = p^*, p_{[0,t]} = p^*_{[0,t]} = p^*$ or $\omega_{[t,T]} = \omega^*_{[t,T]}, \omega_{[0,t]} = \omega^*_{[0,t]}$,

$$\mathbb{E}\psi_i = 0, i = 1,\ldots,N. \tag{5}$$

Theorem 3.1 show that $\mathbb{E}\psi_i = 0$ under the null, as long as either the transition or the marginal state-action distribution function is correct specified. An empirical estimator for $\mathbb{E}\psi_i$ can thus be constructed to detect the deviation from the null. Specifically, a large value of the estimator suggests that the alternative hypothesis is likely to hold. To construct this estimator, we propose to use some flexible modern ML methods (e.g., random forest and neural network) to estimate $(p_{[t,T]}, p_{[0,t]})$ and $(\omega_{[t,T]}, \omega_{[0,t]})$, and use $(\widehat{p}_{[t,T]}, \widehat{p}_{[0,t]})$ and $(\widehat{\omega}_{[t,T]}, \widehat{\omega}_{[0,t]})$ to denote the corresponding estimators. Consider the following sample estimator,

$$\frac{1}{N}\sum_{i=1}^{N} \widehat{\psi}_i. \tag{6}$$

We make a few remarks. First, compared to the naive plug-in estimator Equation 3, Equation 6 is doubly robust due to the inclusion of the extra data augmentation term, making it less sensitive to the impact of the biases of ML estimators. Specifically, as we show in the proof of Theorem 3.1, the bias of Equation 6 will decay at a faster rate than that of the ML estimator. To the contrary, the bias of Equation 3 is likely to be of the same order of magnitude as that of the ML estimator, which can lead to invalid inference. Second, as commented earlier, the presence of an absolute value function in Equation 2 brings additional challenges in deriving the doubly robust estimator, leading to the violation of the pathwise differentiability. It essentially prevents us from construct the doubly robust estimator based on the efficient influence function (EIF, see e.g., Kennedy, 2022) in the classical semiparametric theory. We remark that many of the existing doubly robust estimators (e.g., DML, DRL) are obtained based on the EIF. Nonetheless, our approach is inspired by these estimators which typically involve an outcome regression-type model and a propensity score-type model. Third, the proposed test statistic is based on a modified version of Equation 6 that involve both sample splitting and cross fitting. These techniques enable us to derive the limiting distribution of the estimator under minimal conditions. We formally introduce our test statistic in the next section. A pseudocode summarizing our proposal is given in Algorithm 1.

3.2. Test Statistic

Suppose we have at least two subjects, i.e., $N \geq 2$. We begin by randomly and evenly dividing the subject indices $\{1, ..., N\}$ into two disjoint sets $\mathcal{I}_1$ and $\mathcal{I}_2$, that is, $|\mathcal{I}_1| = \lceil N/2 \rceil$ and $\mathcal{I}_2 = \{1, ..., N\} - \mathcal{I}_1$. Let $(\widehat{p}_{[t,T]}, \widehat{p}_{[0,t]})$ and $(\widehat{\omega}_{[t,T]}, \widehat{\omega}_{[0,t]})$ denote the corresponding modern ML estimators trained on the subjects in $\mathcal{I}_1$. For $t \in (0, T-1), h \in \mathcal{H}$, define

$$\widehat{S}(t,h) = \widehat{\sigma}(t,h)^{-1} \frac{1}{|\mathcal{I}_2|} \sum_{i \in \mathcal{I}_2} \widehat{\psi}_i, \tag{7}$$

where sampling variance estimator $\widehat{\sigma}^2(t,h)$ has the form

$$\frac{1}{|\mathcal{I}_2|T-1} \sum_{i \in |\mathcal{I}_2|} \Bigg\{ \sum_{j=0}^{t-1} \Big(\widehat{\phi}_{[0,t]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h) - \widehat{\mu} \Big)^2 + \sum_{j=t}^{T-1} \Big(\widehat{\phi}_{[t,T]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h) - \widehat{\mu} \Big)^2 \Bigg\},$$

where $\widehat{\mu} = \frac{1}{|\mathcal{I}_2|T} \sum_{i \in |\mathcal{I}_2|} \big\{ \sum_{j=0}^{t-1} \widehat{\phi}_{[0,t]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h) + \sum_{j=t}^{T-1} \widehat{\phi}_{[t,T]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h) \big\}$. The purpose of including $\widehat{\sigma}^2(t,h)$ in Equation 7 is to normalize $\widehat{S}(t,h)$ for each different t and h, in order to make them comparable.

Our test statistic is given by

$$\widehat{\Gamma} = \max_{\epsilon T \leq t \leq (1-\epsilon)T} \max_{b \in \{1,...,B\}} \sqrt{t(T-t)/T^2} \widehat{S}(t, h_b). \tag{8}$$

We make a few explanations here. First, the proposed test statistic is an adaption to the CUSUM statistic, which is popular in the field of change point detection (Csörgö et al., 1997). Second, $\{h_b\}_{b=1,..,B}$ are a set of test functions from $\mathcal{H}$. The number of test functions B is allowed to diverge with the number of subjects N and the horizon T. In particular, the proposed test controls the type-I error as long as $B = O\big((NT)^{c_1}\big)$ for any constant $c_1 > 0$. Nonetheless, considering a number of test functions instead of a single one greatly increases the power of the resulting test. In order to make the function class to be rich and flexible, we set $\mathcal{H}$ to be the class of neural networks (NNs). Specifically, we consider NNs with one hidden layer, a finite number of hidden nodes, and sigmoid activation function.The parameters of the NNs are initialized by the standard normal distribution. In general, the NNs with multiple layers can be adopted in our approach, however, we opt to use a single hidden layer since they produced similar performance in our numerical studies. Third, $\epsilon \in (0, 0.5)$ denotes some user-defined boundary removal parameter. The purpose of using ϵ is to ensure that there is sufficient data at the boundary of the trajectory so that the estimators can be trained well (Yu & Chen, 2017). Fourth, $\sqrt{t(T-t)/T^2}$ is the weighting factor commonly used in CUSUM statistic, which is used to adjust the scales for different t.

3.3. Bootstrap for the p-value

In this section, we propose a Gaussian multiplier bootstrap method to approximate the limiting distribution of $\widehat{\Gamma}$ and compute the resulting p-value. First, define a bootstrapped version of $\widehat{S}(t,h)$ which approximates $\widehat{S}(t,h)$ in distribution, denoted as $\widehat{S}^q(t,h)$,

$$\frac{1}{|\mathcal{I}_2|(T-t)} \sum_{i \in \mathcal{I}_2}^{N} \sum_{j=t}^{T-1} e_{i,j} \widehat{\phi}_{[t,T]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h) - \frac{1}{|\mathcal{I}_2|t} \sum_{i \in \mathcal{I}_2} \sum_{j=0}^{t} e_{i,j} \widehat{\phi}_{[0,t]}(S_{i,j}, A_{i,j}, S_{i,j+1}; h),$$

where $\{e_{i,j}\}_{i \in \mathcal{I}_2, 0 \leq j \leq T-1}$ are i.i.d. standard normal random variables. q is the bootstrap sample index that takes values from $\{1, ..., Q\}$, where Q is the total number of bootstrap samples. The bootstrapped test statistic is given by

$$\widehat{\Gamma}^q = \max_{\epsilon T \leq t \leq (1-\epsilon)T} \max_{b \in \{1,...,B\}} \sqrt{t(T-t)/T^2} \widehat{S}^q(t, h) \tag{9}$$

We use these empirical bootstrap values to approximate the distribution of $\widehat{\Gamma}$ under H_0. The resulting p-value can be calculated by $\widehat{p} = \sum_{q=1}^{Q} \mathbb{I}(\widehat{\Gamma} \geq \widehat{\Gamma}^q)/Q$. We reject the null when $\widehat{p}$ is smaller than a given significance level α.

Algorithm 1 Proposed testing procedure

1: **Input:** B, ϵ, α, Q and N observed data with horizon T.
2: **Output:** p-value $\widehat{p}$
3: Randomly generate B random testing functions $\{h_b \in H\}_{1 \leq b \leq B}$
4: Randomly divide index set $\{1, ..., N\}$ into $\mathcal{I}_1$ and $\mathcal{I}_2$, where $|\mathcal{I}_1| = \lceil N/2 \rceil$ and $\mathcal{I}_2 = \{1, ..., N\} - \mathcal{I}_1$.
5: Estimate nuisance functions $\widehat{p}_{[0,t]}$, $\widehat{p}_{[t,T]}$ and $\widehat{\omega}_{[0,t]}$, $\widehat{\omega}_{[t,T]}$ for each $\epsilon T \leq t \leq (1-\epsilon)T$ on $\mathcal{I}_1$.
6: Compute $\widehat{S}(t, h_b)$ according to Equation 7 for each $b = 1, ..., B$ and $\epsilon T \leq t \leq (1-\epsilon)T$ using samples in $\mathcal{I}_2$. Compute $\widehat{\Gamma}$ according to Equation 8.
7: Using Gaussian multiplier bootstrap to calculate $\widehat{\Gamma}^q$ for each $q = 1, ..., Q$ according to Equation 9. Calculate $\widehat{p} = \sum_{q=1}^{Q} \mathbb{I}(\widehat{\Gamma} \geq \widehat{\Gamma}^q)/Q$.

3.4. Theoretical Analysis

In this section, we establish the size property of our test (e.g., valid type-I error control) under the framework of bidirectional asymptotics, which allows either $N \to \infty$ or $T \to \infty$. We begin by introducing the following conditions.

Condition 3.2. The reference density function g is fully supported on the intersection of the support sets of $\omega_{[0,t]}$ and $\omega_{[t,T]}$, that is, for all $(a, s) \in \mathcal{A} \times \mathcal{S}$ where $g(a, s) > 0$, we have $\omega_{[0,t]}(a, s) > 0$ and $\omega_{[t,T]}(a, s) > 0$.

Condition 3.3. Under H_0, $\{S_{0,t}\}_{t\geq 0}$ is a strictly stationary Markov chain.

Condition 3.4. Under H_0, the Markov chain $\{S_{0,t}\}_{t\geq 0}$ is geometrically ergodic when $T \to \infty$.

Condition 3.5. Suppose there exists some $\kappa_1, \kappa_2, \kappa_3, \kappa_4 \in (0, 1/2)$ such that

$$\left[\mathbb{E}\left\{d_{TV}^2\left(\widehat{p}_{[t,T]}, p^*_{[t,T]}\right)\right\}\right]^{1/2} = O\left((NT)^{-\kappa_1}\right),$$
$$\left[\mathbb{E}\left\{d_{TV}^2\left(\widehat{p}_{[0,t]}, p^*_{[0,t]}\right)\right\}\right]^{1/2} = O\left((NT)^{-\kappa_2}\right),$$
$$\left[\mathbb{E}\left\{d_{TV}^2\left(\widehat{\omega}_{[t,T]}, \omega^*_{[t,T]}\right)\right\}\right]^{1/2} = O\left((NT)^{-\kappa_3}\right),$$
$$\left[\mathbb{E}\left\{d_{TV}^2\left(\widehat{\omega}_{[0,t]}, \omega^*_{[0,t]}\right)\right\}\right]^{1/2} = O\left((NT)^{-\kappa_4}\right),$$

where $\{\kappa_i\}_{i=1,\ldots,4}$ satisfy $\kappa_1 + \kappa_3 > 1/2, \kappa_2 + \kappa_4 > 1/2$. In addition, suppose $\frac{g}{\widehat{\omega}_{[t,T]}}$ and $\frac{g}{\widehat{\omega}_{[0,t]}}$ are uniformly bounded.

Condition (C3.2) requires careful selection of the reference density g. The support of g should be a subset of the intersection of the support of $\omega_{[0,t]}$ and $\omega_{[t,T]}$. This ensures that the ratios $\frac{g}{\omega_{[0,t]}}$ and $\frac{g}{\omega_{[t,T]}}$ are bounded. Condition (C3.3) is imposed to simplify the presentation. The proposed test remains valid if condition (C3.3) violates (e.g., the behavior policy is allowed to change). Similar assumptions are commonly imposed in the literature (see e.g., Kallus & Uehara, 2022). Condition (C3.4) is mild and strictly weaker than the uniform ergodicity condition required for the existing RL algorithms (Bhandari et al., 2018; Zou et al., 2019), allowing us to establish the asymptotic distribution of our test statistic when $T \to \infty$. We also remark that most existing theoretical analysis in the RL literature require the transition tuples to be independent over time. Our framework is more general in that it allows temporal dependence.

Condition (C3.5) requires that the total variation distance between the oracle distribution and the estimated distribution satisfies certain nonparametric convergence rate (e.g., slower than $(NT)^{-1/2}$). Under this condition, the bias of the proposed test statistic decays at a rate of $O((NT)^{-1/2})$, much faster than those in Condition (C3.5). This is made possible due to the double robustness property (Theorem 3.1), which ensures a zero bias if either one of two nuisance function estimators $(\widehat{p}_{[0,t]}, \widehat{p}_{[t,T]})$ or $(\widehat{\omega}_{[0,t]}, \widehat{\omega}_{[t,T]})$ is replaced with its oracle value. Together with the Neyman orthogonality property (Chernozhukov et al., 2018) of the estimating equation Equation 6, the bias term can be represented as a product of the difference between two nuisance function estimators and corresponding oracle values. Consequently, as long as $\kappa_1 + \kappa_2 > 1/2$ holds, the test statistic converges at a parametric rate, asymptotically controls the type-I error. Besides, the condition for nonparametric convergence rates can often be attainable for various modern ML methods in high-dimensional settings, which is commonly used in classical semiparametric literature (Biau, 2012; Hestness et al., 2017; Kallus & Uehara, 2022; Chernozhukov et al., 2018; Farrell et al., 2021). This rate assumption can be further relaxed by introducing high-order influence function (Mukherjee et al., 2017; Robins et al., 2017; Shi et al., 2021a). The boundedness assumption in condition (C3.5) is reasonable since g is user-specified. In practice, we can set $g = \frac{1}{t}\widehat{\omega}_{[0,t]} + \frac{1}{T-t}\widehat{\omega}_{[t,T]}$ to automatically satisfy this condition.

Before presenting the theorem of the test consistency, we define

$$\tilde{\phi}_{[t_1,t_2]}(S_{0,j}, A_{0,j}, S_{0,j+1}; h) = \text{sgn}\big(\widehat{\Delta}(A_{0,j}, S_{0,j}; h, t)\big)$$
$$\big[h(S_{0,j+1}) - \mathbb{E}^*_{[t_1,t_2]}[h(S')|A_{0,j}, S_{0,j}]\big]\frac{g(A_{0,j}, S_{0,j})}{\omega^*_{[t_1,t_2]}(A_{0,j}, S_{0,j})}.$$

Theorem 3.6 (Size). *Assume (C3.2) - (C3.5) hold. Suppose $B = O((NT)^{c_1})$ for any $c_1 > 0$ and $t = O(T)$ for any $t \in [\epsilon T, (1-\epsilon)T]$. In addition, suppose there exists some $\zeta > 0$ such that for any $i \in \mathcal{I}_2, j \in [0, T-1]$, the variance of $\tilde{\phi}_{[0,t]}(S_{0,j}, A_{0,j}, S_{0,j+1}; h)$ and $\tilde{\phi}_{[t,T]}(S_{0,j}, A_{0,j}, S_{0,j+1}; h)$ are both greater than ζ for any h, t, then as either $N \to \infty$ or $T \to \infty$, we have $\mathbb{P}(\widehat{p} \leq \alpha | H_0) = \alpha + o(1)$.*

Theorem 3.6 implies that the size of the proposed test can be controlled at the nominal level as either $N \to \infty$ or $T \to \infty$. The proof of this theorem relies on the high-dimensional martingale central limit theorem (Belloni & Oliveira, 2018), which allows the use of Gaussian multiplier bootstrap to acquire valid p-value.

4. Numerical Experiments

In this section, we conduct four numerical experiments to assess the finite-sample performance of the proposed testing procedure. A toy example (Section 4.2) illustrates the double robustness property of the proposed test statistic against model misspecification. Synthetic data (Section 4.3) demonstrates that the proposed testing procedure is superior to two existing offline stationarity tests in high-dimensional RL setting. A grid-world example (Section 4.4) demonstrates the usefulness of the proposed test in the task of learning optimal policy. Semi-synthetic data (Section 4.5) illustrates the usefulness of the proposed test in the real world setting. Code is available at https://github.com/jtwang95/Double_CUSUM_RL.

4.1. Implementation Details

To implement the proposed test, the boundary removal parameter ϵ is set to 0.1. 5000 bootstrap samples are gen-

erated to compute p-values. We set the reference density $g = \frac{1}{T-t}\widehat{\omega}_{[t,T]} + \frac{1}{t}\widehat{\omega}_{[0,t]}$. To achieve a trade-off between the computational complexity and accuracy, we do not test all the $t \in [\epsilon T, (1-\epsilon)T]$ in practice. Instead, we recommend selecting a subset as the candidate change points.

When dealing with discrete-state-space MDP, we consider the class of testing function $\mathcal{H}$ to be the cross table of states and actions, where the function value of each state-action pair is sampled from standard normal distribution. The transition functions $p_{[0,t]}$, $p_{[t,T]}$ and the marginal distributions $\omega_{[0,t]}$ and $\omega_{[t,T]}$ are estimated through frequency tables. In the context of continuous state-space MDP with binary actions, $\mathcal{H}$ is set to be the class of feed-forward neural networks that contain a single hidden layer with 32 neurons and the sigmoid function as the activation function. We use the neural network to estimate the transition functions $p_{[0,t]}$ and $p_{[t,T]}$. To be more specific, we assume each transition function is a multivariate normal distribution with mean $\mu(s,a)$ and diagonal covariance matrix with diagonal elements being $\sigma^2(s,a)$, where $\mu(s,a=0)$, $\mu(s,a=1)$, $\sigma^2(s,a=0)$ and $\sigma^2(s,a=1)$ are four separate neural networks. The loss function is set to be the Gaussian negative log likelihood. $\omega_{[0,t]}$ and $\omega_{[t,T]}$ are learned through a combination of Gaussian mixture model (GMM) and logistic regression (LR), where the GMM is used to learn the marginal distribution of state S and the LR is to learn the conditional distribution of action A given S. The number of mixture components ($n = 1, ..., 4$) and the covariance type (full, tied, diagonal, spherical) for GMM are selected with the lowest Bayesian information criterion (BIC). Monte Carlo integration with $M = 100$ Monte Carlo samples is used to approximate values of the integrals and expectations.

4.2. A Toy Example

First, we consider a toy example with one-dimensional discrete state and binary action to illustrate the doubly robust property. Specifically, we aim to demonstrate the size and the power of the proposed test when either $M_1 = (p_{[0,t]}, p_{[t,T]})$ or $M_2 = (\omega_{[0,t]}, \omega_{[t,T]})$ is corrected specified. We consider the following four scenarios: (i) both M_1 and M_2 are correct; (ii) only M_1 is misspecified; (iii) only M_2 is misspecified; (iv) M_1 and M_2 are both misspecified. To misspecify M_1 and M_2, we inject some noise into the correct models with noise level $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, where higher values correspond to higher noise. See Appendix B.1 for more details about data generating mechanism and model misspecification.

Figure 1 shows the empirical rejection probabilities of the proposed test under a) stationary and b) non-stationary environments with different misspecification settings for M_1 and M_2. It can be seen that the proposed test can control the type I error rate at the nominal level as long as either M_1 or M_2 is correctly specified. Additionally, the proposed test achieves the highest power when both M_1 and M_2 are correctly specified.

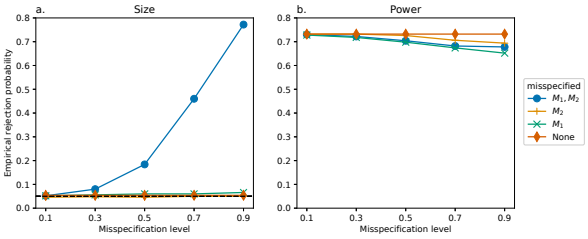

Figure 1. Empirical rejection probability of the proposed test under stationary and non-stationary environments with different misspecification settings for M_1 and M_2 (see Section 4.2 for details). Results are aggregated over 500 replications.

4.3. Synthetic Data: Superiority under Higher Dimensions

In this simulation, we compare the proposed test with two existing stationarity tests in offline reinforcement learning, which is ODCP (Padakandla et al., 2020) and CUSUM-RL (Li et al., 2022). We consider two non-stationary scenarios in this simulation: (i) the non-stationarity occurs in the state transition function; (ii) the non-stationarity occurs in the reward function. For each scenario, we consider four settings where the dimension of the state variables d_S can take values from $\{1, 10, 20, 30\}$. In all scenarios, we fix $N = 100$, $T = 50$, $\alpha = 0.05$ and true change point $t_{cpt} = 25$. In this simulation, we set $B = 100$ and use a multi-split version of the test with 10 random binary splits and set $\gamma = 0.15$ to combine the p-values (see details in Appendix A). Besides, we sequentially apply the proposed test on time interval $[T-\kappa, T]$, where κ takes values from sequence $10, 15, 20, 25, 30, 35, 40$. This strategy together with the stationarity test can be used to detect the location of the single change point. According to the data generating mechanism, the null hypothesis, where the MDP is stationary, is true when $\kappa \leq 25$ and the alternative hypothesis is true when $\kappa > 25$. See Appendix B.2 for more details about the simulation settings.

Figure 2 shows the empirical rejection probability profiles of the proposed test, ODCP and CUSUM-RL over different κ. Note that two different test statistics are used for CUSUM-RL, which are integral (int) and normalized (nor) version. It can be seen that the proposed test and two CUSUM-RL tests can properly control the type I error in all cases, while the ODCP test fails to control the type I error in high-dimensional settings. Power-wise, the proposed test and CUSUM-RL can correctly identify the true change point when $d_S = 1$, while only the proposed test can identify the true change point when $d_S \geq 10$.

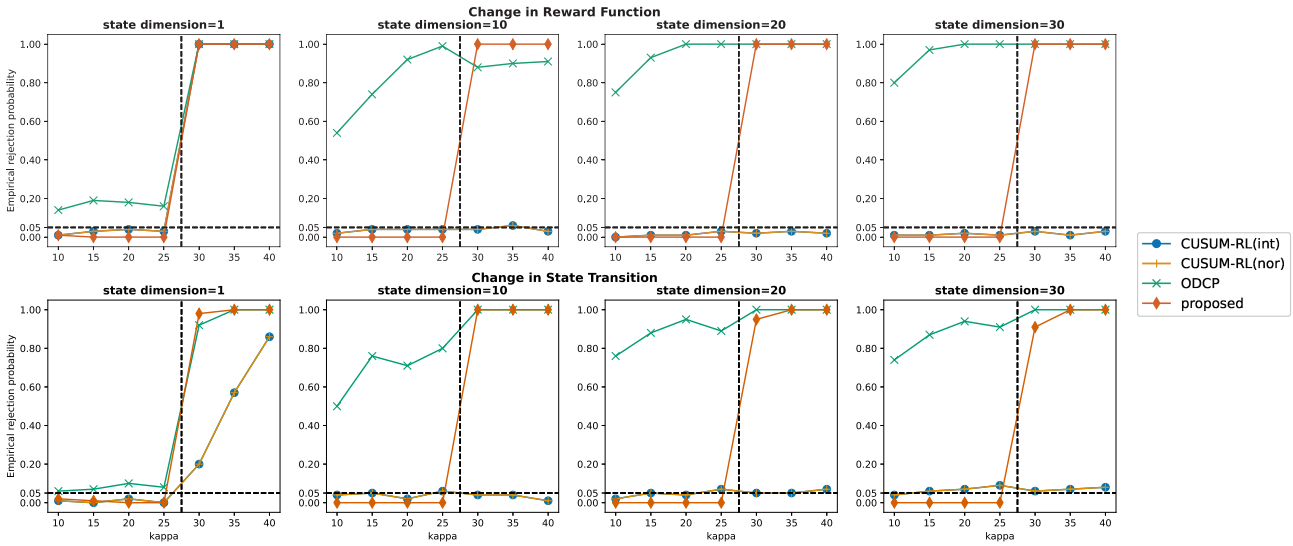

Figure 2. Empirical rejection probabilities (y-axis) over different κ (x-axis) under different simulation scenarios for state dimension $d_S = 1, 10, 20, 30$. The results are aggregated over 100 replications. See details in Section 4.3.

4.4. Grid World Example

In this section, we apply the proposed test to a grid world example to illustrate the usefulness of the test for policy learning. The grid size is set to be $4 * 4$ and the action space is set to $\{$*right, up, left, down*$\}$. We manually introduce non-stationarity to the environment and set the true change point $t_{cpt} = 25$, where $N = 100$ and $T = 50$. The environment setting is detailed in Appendix B.3. To implement the test, we set $B = 100$ and adapt a multi-split version of the test (see details in Appendix A) with 10 random splits.

Empirical rejection probabilities over different κ are shown in Appendix Figure B.1. It can be seen that the proposed test can capture the true change point 25 correctly. Furthermore, we evaluate the value of two policies: the first one is learned based on the data points after the detected change point ($[t_{cpt}, T]$); the second one is based on the whole dataset ($[0, T]$), which ignores the non-stationarity. To perform policy evaluation, we first use Q-learning to learn the "optimal" policy; we then simulate 100 trajectories with a length of horizon 50 using the learned policies to calculate the average discounted (0.9) reward as the value. According to Appendix Figure B.1, it can be seen that the learned policy based on the data after the change point achieves a higher value.

4.5. Semi-synthetic Data

In this simulation, we explore a batch online RL scenario where the policy is periodically updated during data collection and updated policy is subsequently used to collect new data. Unlike traditional online RL, the online learning agent in batch online RL updates its policy only at pre-determined time points, which has many real world applications. For instance, in the context of mobile health, the sensor data is collected from different wearable/mobile devices, which may need regular human-in-the-loop data quality checks, linkage and integration, making batch updating a practical and safer choice in healthcare settings. We consider the presence of change points in this setup, which violates the stationarity assumption. The data generation process involves collecting an initial offline dataset with a random policy, updating the policy using a chosen policy update strategy, and using the updated policy equipped with ϵ-greedy algorithm to collect new data. This process is repeated periodically, aiming to maximize the cumulative reward obtained. The simulated semi-synthetic dataset has a similar structure to the real world dataset in Section 5. In this simulation, we consider two scenarios: (i) three change points , (ii) no change point. See more details on simulation setup in Appendix B.4.

We compare the average reward obtained using four different policy update strategies: (i) *proposed*, where the policy is learned using the data after the most recent change point identified by the proposed test; (ii) *random*, which uses an unchanged random policy throughout the simulation; (iii) *slide*, the policy that is updated using the most recent data that falls into a slide window; (iv) *stationary*, where the policy is only updated once using the initial offline dataset; (v) *oracle*, where the policy is learned using data after true change points. According to Figure 3, it can be seen that the average reward received using the *proposed* strategy is similar to *oracle*, *slide* and *stationary* when the environment is stationary, indicating that the proposed test brings no harm to policy learning when dealing with stationary

RL problems. However, in the non-stationary environment, only *proposed* yield average rewards close to *oracle*, while the other three strategies exhibit significantly lower average rewards. This indicates that (i) policy learning is necessary for RL problem (*random* v.s. *oracle*); (ii) ignoring the non-stationarity can lead to poor performance of the learned policy (*stationary* v.s. *oracle*); (iii) failure to detect the change points can lead to poor performance of learned policy (*slide* v.s. *oracle*).

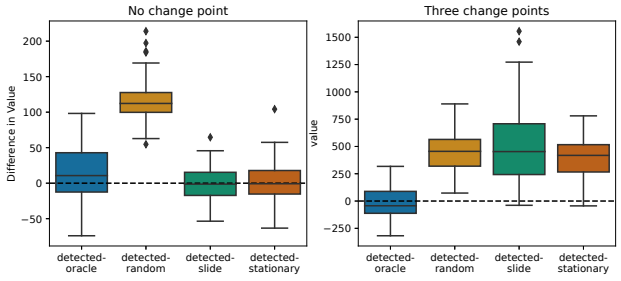

Figure 3. Boxplot of the difference in value between the *proposed* and four other policy update strategies. See details in Section 4.5.

5. Real Data Application

In this section, we apply the proposed testing procedure to a real-world mobile health dataset collected from a micro-randomized trial (MRT) aiming at improving the health outcomes of the medical interns in the United States by sending the push notifications through mobile app to induce and maintain healthy behaviors related to physical activity, sleep and mood (NeCamp et al., 2020). For each week, each intern has probability 0.75 to be randomized into the notification arm, and the intern in this arm has probability 0.5 to receive daily push notifications. Conversely, interns in the no-notification arm receive no push notifications through the week. The trial lasted for 21 weeks, during which wearable devices recorded daily measurements of step count, sleep duration, and mood score. We model the data using MDP, where S_t is the averaged step count, sleep duration and mood score at week t and R_t is the average step count at week $t+1$. $A_t = 1$ if the intern is randomized into the notification arm at week t. We apply the proposed test to assess the stationarity assumption for this dataset.

To implement the proposed method, we set $\epsilon = 0.1$ and search for change points between week 3 to 18. We set $B = 1000, M = 100$. See implementation details in Appendix B.5. The p-values of the stationarity test for different κ are shown in Figure 4. It can be seen that there exists a change point for the interns in specialty *Internal Medicine* at week 16 ($= T - (\kappa^* - 1) = 21 - (6 - 1)$; κ^* is the first κ with p-value < 0.05), while no change point is detected for interns in specialty *Family Practice*.

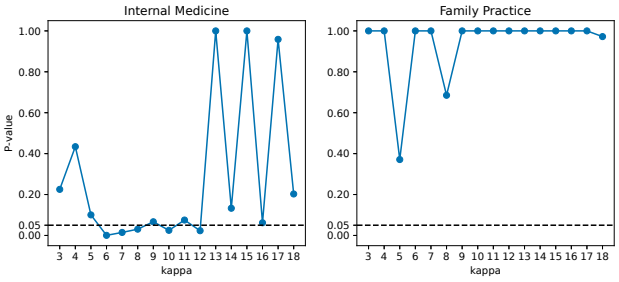

Figure 4. P-values of applying the proposed test to the IHS data over different κ for two specialities. See details in Section 5.

6. Discussion

In this paper, we present a novel doubly robust testing procedure to test the stationarity assumption in high dimensional offline reinforcement learning. Our proposed test combines the approximation capability of modern machine learning methods with semiparametric techniques in statistical inference, resulting in a test with sound statistical property under the bidirectional asymptotics. We demonstrate that the proposed test has proper control of type I error and achieves higher empirical power through both theoretical analysis and extensive numerical studies.

However, policy learning based on our approach is applicable only in the presence of some degree of homogeneity in the dataset. Specifically, we require that 1) some episodes share the locations of change points and 2) the MDPs are piecewise constant. In contrast, single-time-based policy learning approaches, such as those proposed by Jin et al. (2018) and Jin et al. (2021), can handle completely heterogeneous setting where the MDP models for each time and each episode are different. However, these methods may be inferior to detection-based approaches in the presence of homogeneity. In summary, both single-time based methods and detection-based methods have their own merits. While single-time based methods are capable of handling more general settings, detection-based methods demonstrate greater sample efficiency when homogeneity is present. Theoretical work on investigating the performance guarantee of policy learning following detection-based methods and the robustness under deviations from homogeneity warrant future research.

Acknowledgements

We would like to thank the anonymous (meta-)reviewers of ICML 2023 for helpful comments. This work was partially supported by grants from the National Institutes of Health (R01 MH101459 to ZW; R01 NR013658 to JW & ZW), an EPSRC grant EP/W014971/1 (to CS) and an investigator grant from Precision Health Initiative at the University of Michigan to ZW. We thank Dr. Srijan Sen for generous support in the IHS data access.

References

Alegre, L. N., Bazzan, A. L., and da Silva, B. C. Quantifying the impact of non-stationarity in reinforcement learning-based traffic signal control. *PeerJ Computer Science*, 7: e575, 2021.

Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.

Auer, P., Jaksch, T., and Ortner, R. Near-optimal regret bounds for reinforcement learning. *Advances in neural information processing systems*, 21, 2008.

Bai, C., Wang, L., Yang, Z., Deng, Z., Garg, A., Liu, P., and Wang, Z. Pessimistic bootstrapping for uncertainty-driven offline reinforcement learning, 2022.

Basseville, M., Nikiforov, I. V., et al. *Detection of abrupt changes: theory and application*, volume 104. prentice Hall Englewood Cliffs, 1993.

Belloni, A. and Oliveira, R. I. A high dimensional central limit theorem for martingales, with applications to context tree models. *arXiv preprint arXiv:1809.02741*, 2018.

Bercu, B. and Touati, A. Exponential inequalities for self-normalized martingales with applications. *The Annals of Applied Probability*, 18(5):1848 – 1869, 2008. doi: 10.1214/07-AAP506.

Bhandari, J., Russo, D., and Singal, R. A finite time analysis of temporal difference learning with linear function approximation. In *Conference on learning theory*, pp. 1691–1692. PMLR, 2018.

Biau, G. Analysis of a random forests model. *The Journal of Machine Learning Research*, 13(1):1063–1095, 2012.

Bradley, R. C. Basic Properties of Strong Mixing Conditions. A Survey and Some Open Questions. *Probability Surveys*, 2(none):107 – 144, 2005. doi: 10.1214/154957805100000104.

Cai, H., Shi, C., Song, R., and Lu, W. Deep jump learning for off-policy evaluation in continuous treatment settings. *Advances in Neural Information Processing Systems*, 34: 15285–15300, 2021.

Chen, B., Liu, Z., Zhu, J., Xu, M., Ding, W., Li, L., and Zhao, D. Context-aware safe reinforcement learning for non-stationary environments. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10689–10695. IEEE, 2021.

Chen, L. and Luo, H. Near-optimal goal-oriented reinforcement learning in non-stationary environments. *arXiv preprint arXiv:2205.13044*, 2022.

Chen, X. and Christensen, T. M. Optimal uniform convergence rates and asymptotic normality for series estimators under weak dependence and weak conditions. *Journal of Econometrics*, 188(2):447–465, 2015.

Chen, X., Zhu, X., Zheng, Y., Zhang, P., Zhao, L., Cheng, W., Cheng, P., Xiong, Y., Qin, T., Chen, J., et al. An adaptive deep rl method for non-stationary environments with piecewise stable context. *arXiv preprint arXiv:2212.12735*, 2022.

Chernozhukov, V., Chetverikov, D., and Kato, K. Gaussian approximations and multiplier bootstrap for maxima of sums of high-dimensional random vectors. *The Annals of Statistics*, 41(6), December 2013. ISSN 0090-5364. doi: 10.1214/13-AOS1161.

Chernozhukov, V., Chetverikov, D., and Kato, K. Detailed proof of nazarov's inequality. *arXiv preprint arXiv:1711.10696*, 2017.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 01 2018. ISSN 1368-4221. doi: 10.1111/ectj.12097.

Cheung, W. C., Simchi-Levi, D., and Zhu, R. Reinforcement learning for non-stationary markov decision processes: The blessing of (more) optimism. In *International Conference on Machine Learning*, pp. 1843–1854. PMLR, 2020.

Csörgö, M., Csörgö, M., Horváth, L., et al. *Limit theorems in change-point analysis*. John Wiley & Sons, 1997.

Da Silva, B. C., Basso, E. W., Bazzan, A. L., and Engel, P. M. Dealing with non-stationary environments using context detection. In *Proceedings of the 23rd international conference on Machine learning*, pp. 217–224, 2006.

Farrell, M. H., Liang, T., and Misra, S. Deep neural networks for estimation and inference. *Econometrica*, 89(1): 181–213, 2021.

Fei, Y., Yang, Z., Wang, Z., and Xie, Q. Dynamic regret of policy optimization in non-stationary environments. *Advances in Neural Information Processing Systems*, 33: 6743–6754, 2020.

Feng, F., Huang, B., Zhang, K., and Magliacane, S. Factored adaptation for non-stationary reinforcement learning. *arXiv preprint arXiv:2203.16582*, 2022.

Fujimoto, S., Meger, D., and Precup, D. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pp. 2052–2062. PMLR, 2019.

Gajane, P., Ortner, R., and Auer, P. A sliding-window algorithm for markov decision processes with arbitrarily changing rewards and transitions. *arXiv preprint arXiv:1805.10066*, 2018.

Hadoux, E., Beynier, A., and Weng, P. Sequential decision-making under non-stationary environments via sequential change-point detection. In *Learning over multiple contexts (LMCE)*, 2014.

Hamari, J., Koivisto, J., and Sarsa, H. Does gamification work?–a literature review of empirical studies on gamification. In *2014 47th Hawaii international conference on system sciences*, pp. 3025–3034. Ieee, 2014.

Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., Patwary, M., Ali, M., Yang, Y., and Zhou, Y. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Igl, M., Farquhar, G., Luketina, J., Boehmer, W., and Whiteson, S. Transient non-stationarity and generalisation in deep reinforcement learning. *arXiv preprint arXiv:2006.05826*, 2020.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is q-learning provably efficient? *Advances in neural information processing systems*, 31, 2018.

Jin, Y., Yang, Z., and Wang, Z. Is pessimism provably efficient for offline rl? In *International Conference on Machine Learning*, pp. 5084–5096. PMLR, 2021.

Kallus, N. and Uehara, M. Efficiently breaking the curse of horizon in off-policy evaluation with double reinforcement learning. *Operations Research*, 2022.

Kennedy, E. H. Semiparametric doubly robust targeted double machine learning: a review. *arXiv preprint arXiv:2203.06469*, 2022.

Kilinc, O. and Montana, G. Reinforcement learning for robotic manipulation using simulated locomotion demonstrations. *Machine Learning*, pp. 1–22, 2022.

Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S., and Pérez, P. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23 (6):4909–4926, 2021.

KJ, P., Singh, N., Dayama, P., Agarwal, A., and Pandit, V. Change point detection for compositional multivariate data. *Applied Intelligence*, pp. 1–26, 2021.

Klasnja, P., Smith, S., Seewald, N. J., Lee, A., Hall, K., Luers, B., Hekler, E. B., and Murphy, S. A. Efficacy of contextually tailored suggestions for physical activity: a micro-randomized optimization trial of heartsteps. *Annals of Behavioral Medicine*, 53(6):573–582, 2019.

Kober, J., Bagnell, J. A., and Peters, J. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Kosorok, M. R. and Laber, E. B. Precision medicine. *Annual review of statistics and its application*, 6:263–286, 2019.

Kumar, A., Fu, J., Soh, M., Tucker, G., and Levine, S. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

Levine, S., Kumar, A., Tucker, G., and Fu, J. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Li, M., Shi, C., Wu, Z., and Fryzlewicz, P. Testing stationarity and change point detection in reinforcement learning. *arXiv preprint arXiv:2203.01707*, 2022.

Liao, P., Greenewald, K., Klasnja, P., and Murphy, S. Personalized heartsteps: A reinforcement learning algorithm for optimizing physical activity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(1):1–22, 2020.

Liao, P., Qi, Z., Wan, R., Klasnja, P., and Murphy, S. A. Batch policy learning in average reward markov decision processes. *The Annals of Statistics*, 50(6):3364–3387, 2022.

Liu, S., See, K. C., Ngiam, K. Y., Celi, L. A., Sun, X., and Feng, M. Reinforcement learning for clinical decision support in critical care: comprehensive review. *Journal of medical Internet research*, 22(7):e18477, 2020.

Luckett, D. J., Laber, E. B., Kahkoska, A. R., Maahs, D. M., Mayer-Davis, E., and Kosorok, M. R. Estimating dynamic treatment regimes in mobile health using v-learning. *Journal of the American Statistical Association*, 115(530):692–706, 2020.

Mahmood, A. R., Korenkevych, D., Vasan, G., Ma, W., and Bergstra, J. Benchmarking reinforcement learning algorithms on real-world robots. In *Conference on robot learning*, pp. 561–591. PMLR, 2018.

Mao, W., Zhang, K., Zhu, R., Simchi-Levi, D., and Basar, T. Near-optimal model-free reinforcement learning in non-stationary episodic mdps. In *International Conference on Machine Learning*, pp. 7447–7458. PMLR, 2021.

Marling, C. and Bunescu, R. The ohiot1dm dataset for blood glucose level prediction: Update 2020. In *CEUR workshop proceedings*, volume 2675, pp. 71. NIH Public Access, 2020.

Meinshausen, N., Meier, L., and Bühlmann, P. P-values for high-dimensional regression. *Journal of the American Statistical Association*, 104(488):1671–1681, 2009.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Mukherjee, R., Newey, W. K., and Robins, J. M. Semiparametric efficient empirical higher order influence function estimators. *arXiv preprint arXiv:1705.07577*, 2017.

Murphy, S. A. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):331–355, 2003.

NeCamp, T., Sen, S., Frank, E., Walton, M. A., Ionides, E. L., Fang, Y., Tewari, A., Wu, Z., et al. Assessing real-time moderation for developing adaptive mobile health interventions for medical interns: micro-randomized trial. *Journal of medical Internet research*, 22(3):e15033, 2020.

Nie, X., Brunskill, E., and Wager, S. Learning when-to-treat policies. *Journal of the American Statistical Association*, 116(533):392–409, 2021.

Ortner, R., Gajane, P., and Auer, P. Variational regret bounds for reinforcement learning. In *Uncertainty in Artificial Intelligence*, pp. 81–90. PMLR, 2020.

Padakandla, S., KJ, P., and Bhatnagar, S. Reinforcement learning algorithm for non-stationary environments. *Applied Intelligence*, 50:3590–3606, 2020.

Plaat, A., Kosters, W., and Preuss, M. Deep model-based reinforcement learning for high-dimensional problems, a survey. *arXiv preprint arXiv:2008.05598*, 2020.

Qi, Z., Liu, D., Fu, H., and Liu, Y. Multi-armed angle-based direct learning for estimating optimal individualized treatment rules with various outcomes. *Journal of the American Statistical Association*, 115(530):678–691, 2020.

Qian, M. and Murphy, S. A. Performance guarantees for individualized treatment rules. *Annals of statistics*, 39(2): 1180, 2011.

Raghu, A., Komorowski, M., Ahmed, I., Celi, L., Szolovits, P., and Ghassemi, M. Deep reinforcement learning for sepsis treatment. *arXiv preprint arXiv:1711.09602*, 2017.

Rezaeifar, S., Dadashi, R., Vieillard, N., Hussenot, L., Bachem, O., Pietquin, O., and Geist, M. Offline reinforcement learning as anti-exploration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8106–8114, 2022.

Robins, J. M., Li, L., Mukherjee, R., Tchetgen, E. T., and van der Vaart, A. Minimax estimation of a functional on a structured high-dimensional model. *The Annals of Statistics*, 45(5):1951 – 1987, 2017. doi: 10.1214/16-AOS1515. URL https://doi.org/10.1214/16-AOS1515.

Sen, S., Kranzler, H. R., Krystal, J. H., Speller, H., Chan, G., Gelernter, J., and Guille, C. A prospective cohort study investigating factors associated with depression during medical internship. *Archives of general psychiatry*, 67(6): 557–565, 2010.

Shalev-Shwartz, S., Shammah, S., and Shashua, A. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

Shao, K., Tang, Z., Zhu, Y., Li, N., and Zhao, D. A survey of deep reinforcement learning in video games. *arXiv preprint arXiv:1912.10944*, 2019.

Shcherbina, A., Hershman, S. G., Lazzeroni, L., King, A. C., O'Sullivan, J. W., Hekler, E., Moayedi, Y., Pavlovic, A., Waggott, D., Sharma, A., et al. The effect of digital physical activity interventions on daily step count: a randomised controlled crossover substudy of the myheart counts cardiovascular health study. *The Lancet Digital Health*, 1(7):e344–e352, 2019.

Shi, C., Wan, R., Chernozhukov, V., and Song, R. Deeply-debiased off-policy interval estimation. In *International Conference on Machine Learning*, pp. 9580–9591. PMLR, 2021a.

Shi, C., Xu, T., Bergsma, W., and Li, L. Double Generative Adversarial Networks for Conditional Independence Testing. *Journal of Machine Learning Research*, 22(285): 1–32, 2021b. ISSN 1533-7928.

Shi, C., Wang, X., Luo, S., Zhu, H., Ye, J., and Song, R. Dynamic Causal Effects Evaluation in A/B Testing with a Reinforcement Learning Framework. *Journal of the American Statistical Association*, 0(0):1–13, January 2022a. ISSN 0162-1459. doi: 10.1080/01621459.2022.2027776.

Shi, C., Zhang, S., Lu, W., and Song, R. Statistical inference of the value function for reinforcement learning in infinite-horizon settings. *Journal of the Royal Statistical Society Series B*, 84(3):765–793, 2022b.

Shi, L., Li, G., Wei, Y., Chen, Y., and Chi, Y. Pessimistic q-learning for offline reinforcement learning: Towards optimal sample complexity. In *International Conference on Machine Learning*, pp. 19967–20025. PMLR, 2022c.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Tsiatis, A. A. *Semiparametric theory and missing data*. Springer, 2006.

Uehara, M. and Sun, W. Pessimistic model-based offline reinforcement learning under partial coverage. *arXiv preprint arXiv:2107.06226*, 2021.

Van Hasselt, H., Guez, A., and Silver, D. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Wang, L., Zhou, Y., Song, R., and Sherwood, B. Quantile-optimal treatment regimes. *Journal of the American Statistical Association*, 113(523):1243–1254, 2018.

Wei, C.-Y., Dann, C., and Zimmert, J. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, pp. 1043–1096. PMLR, 2022.

Xie, T., Cheng, C.-A., Jiang, N., Mineiro, P., and Agarwal, A. Bellman-consistent pessimism for offline reinforcement learning. *Advances in neural information processing systems*, 34:6683–6694, 2021.

Xu, Z., Li, Z., Guan, Q., Zhang, D., Li, Q., Nan, J., Liu, C., Bian, W., and Ye, J. Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 905–913, 2018.

Yu, M. and Chen, X. Finite sample change point inference and identification for high-dimensional mean vectors. *arXiv preprint arXiv:1711.08747*, 2017.

Zhou, W., Zhu, R., and Qu, A. Estimating optimal infinite horizon dynamic treatment regimes via pt-learning. *Journal of the American Statistical Association*, accepted, 2022.

Zhou, Y., Qi, Z., Shi, C., and Li, L. Optimizing pessimism in dynamic treatment regimes: A bayesian learning approach. In *International Conference on Artificial Intelligence and Statistics*, pp. 6704–6721. PMLR, 2023.

Zhu, R., Zhao, Y.-Q., Chen, G., Ma, S., and Zhao, H. Greedy outcome weighted tree learning of optimal personalized treatment rules. *Biometrics*, 73(2):391–400, 2017.

Zou, S., Xu, T., and Liang, Y. Finite-sample analysis for sarsa with linear function approximation. *Advances in neural information processing systems*, 32, 2019.

A. A Multi-split Version of the Test

To mitigate the randomness introduced by a single binary split of the dataset, we propose a multi-split version of our test. The multi-split version can have not only better reproducibility and asymptotic family-wise error control, but also helps improve the power when the sample size is limited (Meinshausen et al., 2009). The main idea is to repeat the proposed single split method multiple times with different split, and calculate a combined p-value from all the single p-values. To be more specific, suppose we carry out the binary split on the samples $\{1, ..., N\}$ for L times. Let $\{(\mathcal{I}_1^{(1)}, \mathcal{I}_2^{(1)}), ..., (\mathcal{I}_1^{(L)}, \mathcal{I}_2^{(L)})\}$ denote each binary split. We then apply Algorithm 1 on each split $(\mathcal{I}_1^{(l)}, \mathcal{I}_2^{(l)})$ for $l = 1, ..., L$ and acquire p-values $\{\widehat{p}^{(l)}\}_{l=1,...,L}$. Then the combined p-value can be calculated following the idea of Meinshausen et al. (2009),

$$\widehat{p}^{combined} = \min\left\{1, q_\gamma\Big(\{\widehat{p}^{(l)}/\gamma; l = 1, ..., L\}\Big)\right\}.$$

where q_γ is the empirical γ-quantile function. We can reject H_0 if $\widehat{p}^{combined}$ is less than the pre-defined significance level α.

B. More on the numerical study

B.1. Simulation Setting for Toy Example

B.1.1. General Settings

We consider two settings, which are stationary environment and non-stationary environment with one change point, in this toy example. In both settings, we fix time $T = 10$ and sample size $N = 30$. The significance level α is set to 0.05. The candidate pool of change points is set to be $\{3, 4, 5, 6, 7, 8\}$. We set the state space $\mathcal{S} = \{0, 1\}$, action space $\mathcal{A} = \{0, 1\}$ and reward space $\mathcal{R} = \{0, 1\}$. The action is randomly sampled from a Bernoulli distribution with probability 0.5. To misspecify M_1 and M_2, we inject noise to the corresponding true model with a specified noise level $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. The misspecified model can be written as $M_{mis} = \lambda M_{noise} + (1 - \lambda)M_{true}$ where higher values of λ refer to more noise. We apply the proposed test procedure to 500 randomly simulated data to calculate the empirical rejection probability in each setting. Of note, under this toy example, we are able to derive the exact form of M_1 and M_2 theoretically. We plug in the true models to the test statistic, without estimating them, so that we can directly assess the influence of misspecification on the size and power by eliminating estimation error.

B.1.2. Setting for Stationary Environment

We initiate the state variable from an independent Bernoulli distribution with support $\{0, 1\}$ and satisfies that $\mathbb{P}(S_0 = 0) = \mathbb{P}(S_0 = 1) = 0.5$. The action is randomly sampled from Bernoulli distribution with binary support and satisfies that $\mathbb{P}(A_t = 1) = \mathbb{P}(A_t = 1) = 0.5$ for all $t \in [0, T)$. We define the transition function $\mathbb{P}(S_{t+1} = S_t|S_t, A_t) = 0.8\mathbb{I}(A_t = 1) + 0.2\mathbb{I}(A_t = 0)$ and reward function $\mathbb{P}(R_t = 1|S_t, A_t) = 0.8\mathbb{I}(S_t = 1) + 0.2\mathbb{I}(S_t = 0)$.

Recall that $M_1 = (p_{[0,t]}, p_{[t,T]})$ and $M_2 = (\omega_{[0,t]}, \omega_{[t,T]})$. The noise for M_1 is shown in Table 1 and 2. The noise for M_2 is shown in Table 3.

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(a) $S_t = 0, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(b) $S_t = 0, A_t = 1$

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(c) $S_t = 1, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.25	0.25
	1	0.25	0.25

(d) $S_t = 1, A_t = 1$

Table 1. Injected noise for transition function $p_{[0,t]}$ under stationary situation depicted in Appendix B.1.2.

B.1.3. Setting for Non-stationary Situation

We consider piece-wise stationary situation. Denote the true location of the change point as $t_{cpt} \in (0, T)$. The initial state S_0 is sampled from Bernoulli distribution with $\mathbb{P}(S_0 = 1) = 0.5$. The transition function is defined as:

$$\mathbb{P}(S_{t+1} = S_t|S_t, A_t) = \begin{cases} 0.8\mathbb{I}(A_t = 1) + 0.2\mathbb{I}(A_t = 0) & \text{if } t \in [0, t_{cpt}) \\ 0.7\mathbb{I}(A_t = 1) + 0.3\mathbb{I}(A_t = 0) & \text{if } t \in [t_{cpt}, T] \end{cases},$$

		R_t	
		0	1
S_{t+1}	0	0.01	0.49
	1	0.05	0.45

(a) $S_t = 0, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.05	0.45
	1	0.05	0.45

(b) $S_t = 0, A_t = 1$

		R_t	
		0	1
S_{t+1}	0	0.05	0.45
	1	0.05	0.45

(c) $S_t = 1, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.05	0.15
	1	0.05	0.75

(d) $S_t = 1, A_t = 1$

Table 2. Injected noise for transition function $p_{[t,T]}$ under stationary situation depicted in Appendix B.1.2.

		S_t	
		0	1
A_t	0	0.2	0.3
	1	0.3	0.2

(a) $t < \lceil T/2 \rceil$

		S_t	
		0	1
A_t	0	0.4	0.2
	1	0.2	0.2

(b) $t \geq \lceil T/2 \rceil$

Table 3. Injected noise for $\omega_{[0,t]}$ and $\omega_{[t,T]}$ under stationary situation depicted in Appendix B.1.2.

and the reward function is defined as:

$$\mathbb{P}(R_t = 1|S_t, A_t) = \begin{cases} 0.8\mathbb{I}(S_t = 1) + 0.2\mathbb{I}(S_t = 0) & \text{if } t \in [0, t_{cpt}) \\ 0.7\mathbb{I}(S_t = 1) + 0.3\mathbb{I}(S_t = 0) & \text{if } t \in [t_{cpt}, T] \end{cases}$$

The noise model for M_1 is shown in Table 4 and the noise model for M_2 is shown in Table 5.

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(a) $S_t = 0, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(b) $S_t = 0, A_t = 1$

		R_t	
		0	1
S_{t+1}	0	0.45	0.05
	1	0.45	0.05

(c) $S_t = 1, A_t = 0$

		R_t	
		0	1
S_{t+1}	0	0.25	0.25
	1	0.25	0.25

(d) $S_t = 1, A_t = 1$

Table 4. Injected noise for transition functions $p_{[0,t]}$ and $p_{[t,T]}$ under non-stationary situation depicted in Appendix B.1.3.

B.2. Simulation Setting for Synthetic Data

B.2.1. General Settings

The date generating mechanism is described as follows. Denote d_S as the dimension of the states. We set the state space $\mathcal{S} = \mathbb{R}^{d_S}$, action space $\mathcal{A} = \{0, 1\}$ and reward space $\mathcal{R} = \mathbb{R}$. Before collecting data, we take burn-in step by discarding first 1000 samples. During burn-in, the actions are generated from i.i.d. Bernoulli random variables with probability $\mathbb{P}(A = 1) = 0.5$. The data starts to be collected after burn-in step. The settings for two change point types are detailed in Appendix B.2.2 and B.2.3. We also introduce the behavior policy change in the data generating mechanism, that is, the action for time $t < 0.7T$ is i.i.d. random variables sampled from Bernoulli distribution with $\mathbb{P}(A = 1) = 0.5$ and the action for time $t \geq 0.7T$ is generated from Bernoulli distribution with $\mathbb{P}(A = 1) = 0.8$.

We set the neural network that used to learn $(p_{[0,t]}, p_{[t,T]})$ to have two hidden layers with 128 nodes in each layer along with ReLU activation function. The corresponding learning rate is set to 0.001. The number of test functions B is set to 100. We also clip the density ratios $g/\omega_{[0,t]}$ and $g/\omega_{[t,T]}$ with maximum value being 100.

B.2.2. Scenario I: Change in Reward Function

Transition function

$$\mathbb{P}(S_{t+1}|S_t, A_t) = \mathcal{N}(0.5A_t\mathbf{1}_{d_S} \circ S_t, I_{d_S}).$$

Reward function before change point t^*

$$r_t(S_t, A_t) = -1.5\mathbf{1}_{d_S}^T S_t.$$

		S_t	
		0	1
A_t	0	0.2	0.3
	1	0.3	0.2

Table 5. Injected noise for $\omega_{[0,t]}$ and $\omega_{[t,T]}$ under non-stationary situation depicted in Appendix B.1.3.

Reward function after change point t^*

$$r_t(S_t, A_t) = \mathbf{1}_{d_S}^T S_t.$$

B.2.3. Scenario II: Change in State Transition

Transition function before change point t^*

$$\mathbb{P}(S_{t+1}|S_t, A_t) = \mathcal{N}(-0.5A_t\mathbf{1}_{d_S} \circ S_t, I_{d_S}).$$

Transition function after change point t^*

$$\mathbb{P}(S_{t+1}|S_t, A_t) = \mathcal{N}(0.5 * [\log_{10}(d_S) + 0.5] * A_t\mathbf{1}_{d_S} \circ S_t, I_{d_S}).$$

Reward function

$$r_t(S_t, A_t) = -0.25A_t\bar{S}_t^2 + 4\bar{S}_t,$$

where $\bar{S}$ is the average value over all dimensions.

B.3. Simulation Setting for Grid World Example

In this example, the grid size is to be $4 * 4$ and hence the number of states is 16 ($\mathcal{S} = \{0, 1, ..., 15\}$). The agent is initialized at location $(0, 0)$ ($s_0 = 0$). At each time point, the agent has four actions to take with equal probability, which are *right, up, left* and *down*. The agent will not move if it chooses an action that causes itself to potentially step across the boundary. The immediate rewards the agent receives are shown in Figure B.1(a). Next, we introduce the mechanism of non-stationarity into the simulation. We set the length of horizon of collected trajectories to be $T = 50$, and the true change point is set at time point $t_{cpt} = 25$. Before t_{cpt}, the agent has 0.8 probability to take a wrong move, which is the opposite direction as instructed. For example, if the agent choose action *right*, it will move *left* with 0.8 probability and move *right* with 0.2 probability. On the contrary, the agent has 0.1 probability to take a wrong move after t_{cpt}.

B.4. Simulation Setting for Semi-synthetic Data

The motivation for this simulation study is as follows. Consider an unknown environment, and the researcher wants to an optimal policy so that the sum of future reward is maximized. The researcher may start to collect data by making interactions with the unknown environment using a random policy (e.g., equal probability to perform action 0 or 1 if binary actions) and then learn the optimal policy from this offline dataset. This is the usual setting for RL that assumes the environment does not change, while the underlying model dynamics of the environment can change at some certain time points under non-stationarity. To mimic this problem of interest, we consider the following simulation scheme.

First, we simulate an offline dataset with N subjects and T_0 time points. The offline dataset serves as the initial experience the researcher interacts with the environment, from which some policies can be learned. Note that there may exist some change points in the offline dataset, therefore simply applying stationarity-assumed RL algorithm on the whole offline dataset can lead to useless policy. Second, we update the policy based on the offline dataset based on different strategies. For example, we can apply the proposed test to find the change points in the offline dataset and use the data points after the most recent change points to learn the optimal policy. Or we can ignore the presence of change points and learn the policy using whole offline dataset. Third, we combine the learned policy with ϵ-greedy algorithm for action selection to simulate the next ΔT data points. We repeat this procedure until some stopping criteria. Then we can calculate the average reward, which can serve as a performance metric for different policy update strategies.

0	0	0	0
0	1	1	1
0	1	2	2
0	1	2	3

Method	Value
Overall	1.53
Proposed	2.03

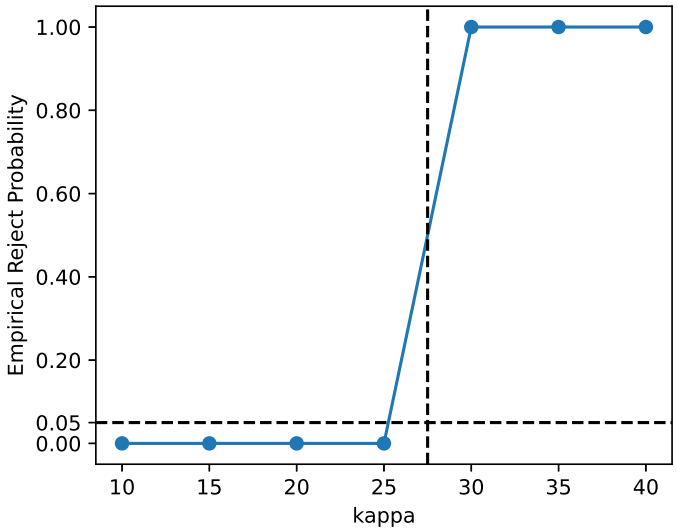

Figure B.1. a) The table shows the immediate reward the agent receives for each state. b) The table shows the values obtained by different policy learning strategies through 100 simulated trajectories. c) Empirical rejection probabilities (y-axis) over different κ (x-axis) for the grid world example. The results are aggregated over 100 simulations. See details in Section 4.4.

The setting of the simulation is described as follows. To mimic the characteristics of real-world data, we first use the data from IHS study to learn an MDP generating model. The state S_t is defined as the concatenation of the cubic root of average daily step count at week t, thus the dimension is 7. The reward R_t is defined as the cubic root of average weekly step count at week $t+1$. The action A_t is a binary variable, which indicates whether the subjects receive the intervention (i.e., push notification in IHS study) at week $t+1$. We assume the transition function $p(S_{t+1}, R_t|S_t, A_t)$ follows a multivariate normal distribution with mean $\mu(S_t, A_t)$ and diagonal covariance matrix Σ. We use a neural network with $[32, 64, 32]$ structure to approximate $\mu(S_t, A_t)$ and the diagonal elements of Σ are estimated using the sampling variance of the residuals of each component. The number of subjects N is set to 200 and the length of offline dataset T_0 is set to 20. The number of newly collected time points ΔT is set to 20. The locations of the change points are pre-selected prior to each simulation and we assume the length of the gap between two change points follows a Poisson distribution with parameter $\lambda = 30$. Three types of change point are considered: (i) change in the mean ($\mu(S_t, A_t)$) of the transition function; (ii) change in the variance (Σ) of the transition function; (iii) change in the reward function. Each type of change point is selected with fixed probability. We consider two scenarios in this numerical study: (i) there are 3 change points in the dataset; (ii) there is no change point in the dataset. The structure of the neural network used to learn $(p_{[0,t]}, p_{[t,T]})$ is set to be two hidden layers with 128 nodes in each layer along with ReLU activation function. We use double deep Q network (double DQN) (Mnih et al., 2015; Van Hasselt et al., 2016) as the policy learning algorithm. The neural net with structure $[32, 64, 128, 64, 32]$ serves as the backbone of the Q network and the discount factor is set to 0.9.

B.5. Real World Example

To implement the test, we adapt a multi-split version of the test (see details in Appendix A) with 10 random splits. The structure of the neural network used to learn the transition functions $p_{[0,t]}$ and $p_{[t,T]}$ are set to $[64, 64]$, where the number of training epochs is set to 500.

C. Notes on Non-pathwise Differentiability of (1)

One common way to construct the doubly robust statistic is to use the Gateaux derivative. Here we try to use this way to derive the DR statistic w.r.t. Equation 1 and show that Equation 1 is not pathwise differentiable.

The Gateaux derivative is defined as

$$\frac{\partial}{\partial \epsilon}\psi\{(1-\epsilon)d\mathbb{P}(z)+\epsilon\delta_{z'}\}\Big|_{\epsilon=0}=\phi(z';\mathbb{P}).$$

The equation holds due to the definition of the influence function.

The functional we used:

$$\psi(\mathbb{P})=\mathbb{E}_{g(A,S)}\Big[\Big|\frac{1}{T-t}\sum_{j=t}^{T-1}\int h(s_p,r)d\mathbb{P}_j(S_p=s_p,R=r|S=s,A=a)$$
$$-\frac{1}{t}\sum_{j=0}^{t-1}\int h(s_p,r)d\mathbb{P}_j(S_p=s_p,R=r|S=s,A=a)\Big|\Big].$$

For the submodel $\mathbb{P}_\epsilon(Z=z)=(1-\epsilon)\mathbb{P}(Z=z)+\epsilon\mathbf{1}(\delta_{z'})$, we have

$$\mathbb{P}_\epsilon(S_p=s_p,R=r|S=s,A=a)=\frac{P_\epsilon(Z=z)}{P_\epsilon(S=s,A=a)}=\frac{(1-\epsilon)\mathbb{P}(Z=z)+\epsilon\mathbf{1}(z=z')}{(1-\epsilon)\mathbb{P}(S=s,A=a)+\epsilon\mathbf{1}(s=s',a=a')}.$$

Therefore, the Gateaux derivative is

$$\frac{d}{d\epsilon}\psi\{(1-\epsilon)\mathbb{P}(z)+\epsilon\delta_{z'}\}\Big|_{\epsilon=0}=\frac{d}{d\epsilon}\sum_{a,s}\Big[\Big|\frac{1}{T-t}\sum_{j=t}^{T-1}\sum_{s_p,r}h(s_p,r)\mathbb{P}_{j,\epsilon}(Z=z|S=s,A=a)$$
$$-\frac{1}{t}\sum_{j=0}^{t-1}\sum_{s_p,r}h(s_p,r)\mathbb{P}_{j,\epsilon}(Z=z|S=s,A=a)\Big|\Big]\Big|_{\epsilon=0}$$
$$=\sum_{a,s}\frac{d\mathbf{sgn}(\Delta(t,h,\epsilon;s,a,\mathbb{P}))}{d\epsilon}\Delta(t,h;s,a,\mathbb{P},\epsilon)$$
$$+\mathbf{sgn}(\Delta(t,h;s,a,\mathbb{P},\epsilon))\frac{d\Delta(t,h,\epsilon;s,a,\mathbb{P})}{d\epsilon}\Big|_{\epsilon=0}.$$

Here $\mathbf{sgn}(\Delta(t,h,\epsilon;s,a,\mathbb{P}))$ is non-differentiable under H_0.

D. Proof for Theorem 3.1

Theorem 3.1 involves two parts: first, $\mathbb{E}\psi_i^*=0$ if the transition functions $p_{[0,t]},p_{[t,T]}$ and marginal state-action distributions $\omega_{[0,t]},\omega_{[t,T]}$ are both correctly specified; second, $\mathbb{E}\psi_i=0$ if one out of the transition functions $p_{[0,t]},p_{[t,T]}$ and marginal state-action distributions $\omega_{[0,t]},\omega_{[t,T]}$ is correctly specified. First part is the direct consequence of the second part, therefore, we focus on the second part in this proof.

Proof. When $p_{[t,T]}=p_{[0,t]}=p^*$, for any t,h we have $\Delta(a,s;h,t)\equiv 0$, thus Equation 5 holds since $\text{sgn}(0)=0$. When $\omega_{[t,T]}=\omega^*_{[t,T]},\omega_{[0,t]}=\omega^*_{[0,t]}$,

$$\mathbb{E}\Big\{\frac{1}{T-t}\sum_{j=t}^{T-1}\text{sgn}\left(\Delta(A_{0,j},S_{0,j};h,t)\right)\left[h(S_{0,j+1},R_{0,j})-\mathbb{E}_{[t,T]}[h(S',R)|A_{0,j},S_{0,j}]\right]\frac{g(A_{0,j},S_{0,j})}{\omega^*_{[t,T]}(A_{0,j},S_{0,j})}\Big\}$$
$$=\int\text{sgn}\left(\Delta(A,S;h,t)\right)\left[\mathbb{E}^*[h(S',R)|A,S]-\mathbb{E}_{[t,T]}[h(S',R)|A,S]\right]g(A,S)\lambda(A,S).$$

Similarly, we have

$$\mathbb{E}\Big\{\frac{1}{t}\sum_{j=0}^{t-1}\text{sgn}\left(\Delta(A_{0,j},S_{0,j};h,t)\right)\left[h(S_{0,j+1},R_{0,j})-\mathbb{E}_{[0,t]}[h(S',R)|A_{0,j},S_{0,j}]\right]\frac{g(A_{0,j},S_{0,j})}{\omega^*_{[0,t]}(A_{0,j},S_{0,j})}\Big\}$$
$$=\int\text{sgn}\left(\Delta(A,S;h,t)\right)\left[\mathbb{E}^*[h(S',R)|A,S]-\mathbb{E}_{[0,t]}[h(S',R)|A,S]\right]g(A,S)\lambda(A,S).$$

Combined two equations together, we show that Equation 5 holds. The doubly-robustness property thus follows. □

E. Proof for Theorem 3.6

Proof. **Step 1.** We begin by showing that any testing function $h \in \mathcal{H}$ is upper bounded by $\log(NT)$. Recall that we set $\mathcal{H}$ to be the class of neural network with one hidden layer, finitely many hidden nodes and sigmoid activation function. Suppose $h_1(\cdot)$ is random sample from $\mathcal{H}$, we can have

$$h_1(x) = \sum_{k=1}^{K} \theta_{1,k}^{(1)} \text{sigmoid}(x^T \theta_{1,k}^{(2)}).$$

Since sigmoid function is bounded, therefore $h_1(\cdot)$ is uniformly bounded by $K \max_{k=1,..,K}(|\theta_{1,k}^{(1)}|)$. Since we sample B random samples from $\mathcal{H}$, which denotes $\{h_b\}_{b=1,...,B}$, therefore, these functions are uniformly bounded by $K \max_{k=1,..,K;b=1,..,B}(|\theta_{b,k}^{(1)}|)$. Recall that the parameters in the neural network are initialized by standard normal distribution. For a standard normal random variable Y, we have $\mathbb{P}(Y \geq t) \leq \exp(-t^2/2)/\sqrt{2\pi}$ for any $t > 1$. Therefore we can conclude that with probability approaching 1, $\max_{k=1,..,K;b=1,..,B}(|\theta_{b,k}^{(1)}|)$ is upper bounded by $\log B$. Since we assume $B = O((NT)^{c_1})$, therefore we have the function $h \in \mathcal{H}$ is absolutely bounded by $\log(NT)$.

Step 2. Note that the test statistic is calculated on the one binary split of the dataset $\mathcal{I}_2$ where $|\mathcal{I}_2| = \lfloor N/2 \rfloor$. To simplify the notation, we assume the total sample size is $2N$, then $|\mathcal{I}_2| = N$. Define $\Delta_{i,j,b,t} = \Delta(A_{i,j}, S_{i,j}; h_b, t)$. Define

$$\Gamma^* = \max_{\epsilon T \leq t \leq (1-\epsilon)T} \max_{b \in \{1,...,B\}} \sqrt{t(T-t)/T^2} S^*(t, h_b),$$

where

$$S^*(t, h_b) = \sigma^*(t, h)^{-1} \Bigg\{ \frac{1}{N(T-t)} \sum_{i=1}^{N} \sum_{j=t}^{T-1} \text{sgn}\left(\widehat{\Delta}_{i,j,b,t}\right) \left[h(S_{i,j+1}) - \mathbb{E}^*_{[t,T]}[h(S')|A_{i,j}, S_{i,j}]\right] \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[t,T]}(A_{i,j}, S_{i,j})}$$
$$- \frac{1}{Nt} \sum_{i=1}^{N} \sum_{j=0}^{t} \text{sgn}\left(\widehat{\Delta}_{i,j,b,t}\right) \left[h(S_{i,j+1}) - \mathbb{E}^*_{[0,t]}[h(S')|A_{i,j}, S_{i,j}]\right] \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[0,t]}(A_{i,j}, S_{i,j})} \Bigg\}.$$

We want to prove $\widehat{\Gamma} = \Gamma^* + o_p((NT)^{-1/2} \log^{-1/2}(NT))$. Note that in this proof we omit the normalization factor $\sigma^2(t, h_b)$ for simplicity. The reason is that we can define an intermediate test statistic $\widehat{S}^*(t, h_b)$ which replaces the $\sigma^*(t, h_b)$ in $S^*(t, h_b)$ with $\widehat{\sigma}(t, h_b)$ and then use the similar steps described below to show that $\widehat{S}^*(t, h_b)$ is close to $S^*(t, h_b)$ and $\widehat{S}(t, h_b)$ is close to $S^*(t, h_b)$.

In the next step, we show that

$$\max_{\epsilon T \leq t \leq (1-\epsilon)T} \max_{b \in \{1,...,B\}} \sqrt{t(T-t)/T^2} |\widehat{S}(t, h_b) - S^*(t, h_b)| = o_p((NT)^{-1/2} \log^{-1/2}(NT)).$$

With some calculations, we can show that for any $t \in [\epsilon T, (1-\epsilon)T]$ and b,

$$\widehat{S}(t, h_b) = S^*(t, h_b) + R_1 + R_2 + R_3 + R_4 + R_5, \tag{10}$$

where the reminder terms $R_i, i = 1, ..., 5$ are given by

$$
\begin{aligned}
R_1(t, h_b) &= \int \left| \widehat{\Delta}(a, s; h_b, t) \right| g(a, s) \lambda(da, ds) \\
&+ \frac{1}{N(T-t)} \sum_{i=1}^{N} \sum_{j=t}^{T-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[\mathbb{E}^*_{[t,T]}[h_b(S')|A_{i,j}, S_{i,j}] - \widehat{\mathbb{E}}_{[t,T]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[t,T]}(A_{i,j}, S_{i,j})} \\
&- \frac{1}{Nt} \sum_{i=1}^{N} \sum_{j=0}^{t-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[\mathbb{E}^*_{[0,t]}[h_b(S')|A_{i,j}, S_{i,j}] - \widehat{\mathbb{E}}_{[0,t]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[0,t]}(A_{i,j}, S_{i,j})}, \\
R_2(t, h_b) &= \frac{1}{N(T-t)} \sum_{i=1}^{N} \sum_{j=t}^{T-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[h_b(S_{i,j+1}) - \mathbb{E}^*_{[t,T]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \\
&\times \left[\frac{g(A_{i,j}, S_{i,j})}{\widehat{\omega}_{[t,T]}(A_{i,j}, S_{i,j})} - \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[t,T]}(A_{i,j}, S_{i,j})} \right], \\
R_3(t, h_b) &= \frac{1}{N(T-t)} \sum_{i=1}^{N} \sum_{j=t}^{T-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[\mathbb{E}^*_{[t,T]}[h_b(S')|A_{i,j}, S_{i,j}] - \widehat{\mathbb{E}}_{[t,T]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \\
&\times \left[\frac{g(A_{i,j}, S_{i,j})}{\widehat{\omega}_{[t,T]}(A_{i,j}, S_{i,j})} - \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[t,T]}(A_{i,j}, S_{i,j})} \right], \\
R_4(t, h_b) &= -\frac{1}{Nt} \sum_{i=1}^{N} \sum_{j=0}^{t-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[h_b(S_{i,j+1}) - \mathbb{E}^*_{[0,t]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \left[\frac{g(A_{i,j}, S_{i,j})}{\widehat{\omega}_{[0,t]}(A_{i,j}, S_{i,j})} - \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[0,t]}(A_{i,j}, S_{i,j})} \right], \\
R_5(t, h_b) &= -\frac{1}{Nt} \sum_{i=1}^{N} \sum_{j=0}^{t-1} \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[\mathbb{E}^*_{[0,t]}[h_b(S')|A_{i,j}, S_{i,j}] - \widehat{\mathbb{E}}_{[0,t]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \\
&\times \left[\frac{g(A_{i,j}, S_{i,j})}{\widehat{\omega}_{[0,t]}(A_{i,j}, S_{i,j})} - \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[0,t]}(A_{i,j}, S_{i,j})} \right].
\end{aligned}
$$

It suffices to show

$$
\max_{\substack{T\epsilon \le t \le T(1-\epsilon) \\ b \in \{1,...,B\}}} \sqrt{t(T-t)/T^2} |R_m(t, b)| = o_p((NT)^{-1/2} \log^{-1/2}(NT)) \tag{11}
$$

for $m = 1, ..., 5$. In the following, we show that 11 holds with $m = 1, 2, 3$. Using similar arguments, one can show that 11 holds with $m = 4, 5$.

Proof of Equation 11 with $m = 1$:
Under C3.3 and C3.4, it follows from Theorem 3.7 of Bradley (2005) that $\{S_{0,j}\}_{j \ge 0}$ is exponentially β-mixing. Denote the resulting β-mixing coefficients by $\beta_0(q)$ that satisfies $\beta_0(q) = O(\rho^q)$ for some $\rho < 1$ and any $q \ge 0$. Since $\{S_{1,j}\}_{j \ge 0}, \{S_{2,j}\}_{j \ge 0}, ..., \{S_{N-1,j}\}_{j \ge 0}$ are i.i.d samples as $\{S_{0,j}\}_{j \ge 0}$, therefore, the β-mixing coefficient of

$$
\{S_{0,0}, S_{0,1}, ..., S_{0,T}, S_{1,0}, S_{1,1}, ..., S_{1,T}, ..., S_{N-1,0}, S_{N-1,1}, ..., S_{N-1,T}\}
$$

satisfies $\beta(q) = O(\rho^q)$ for any $q \ge 0$. Define

$$
\widetilde{\phi}_{[t_1,t_2]}(S_{i,j}, A_{i,j}; h_b) = \text{sgn}(\widehat{\Delta}_{i,j,b,t}) \left[\mathbb{E}^*_{[t_1,t_2]}[h_b(S')|A_{i,j}, S_{i,j}] - \widehat{\mathbb{E}}_{[t_1,t_2]}[h_b(S')|A_{i,j}, S_{i,j}] \right] \frac{g(A_{i,j}, S_{i,j})}{\omega^*_{[t_1,t_2]}(A_{i,j}, S_{i,j})}.
$$

To prove Equation 11 with $m = 1$, it suffices to show

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\left|\frac{1}{\sqrt{N(T-t)}}\sum_{i=1}^{N}\sum_{j=t}^{T-1}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\right|=o_p(\log^{-1/2}(NT)),\tag{12}$$

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\left|\frac{1}{\sqrt{Nt}}\sum_{i=1}^{N}\sum_{j=0}^{t-1}\widetilde{\phi}_{[0,t]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[0,t]}(S_{i,j},A_{i,j};h_b)\right|=o_p(\log^{-1/2}(NT)),\tag{13}$$

since $\max|a+b|\le\max|a|+\max|b|$. For brevity, we only show Equation 12 holds. Proof of Equation 13 is similar and thus omitted.

Under the boundedness assumption, we have $\left|\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\right|\le M$ for some positive value M and hence $\left|\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\right|\le 2M$. Similarly, for any $t\le j\le T$ we have

$$\max_{i,j,b}\mathbb{E}\Big[\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big]^2\le\max_{i,j,b}\mathbb{E}\Big[\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)^2\Big]\equiv\Lambda.$$

Note that Λ is a random variable that depends on $\{h_b\}_{1\le b\le B}$ and $\{A_{i,j},S_{i,j}\}_{i\in\{1,...,N\},0\le j\le T-1}$. Under the boundedness condition, we have

$$\max_{i,j,b}\mathbb{E}\Big[\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)^2\Big]\le O(\log^2 NT)\mathbb{E}\Big\{d_{TV}^2\Big(\widehat{p}_{[t,T]}(S',R|S,A),p^*_{[t,T]}((S',R|S,A)\Big)\Big\}.\tag{14}$$

Given the boundedness conditions and β-mixing property, it follows from Theorem 4.2 of Chen & Christensen (2015) that, for any integers $\tau\ge 0$ and $1<d<NT/2$, we have

$$\begin{aligned}&\mathbb{P}\Big(\Big|\sum_{i=1}^{N}\sum_{j=t}^{T-1}\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|\ge 6\tau\Big|\Lambda\Big)\le\frac{N(T-t)}{d}\beta(d)\\ +\mathbb{P}\Big(\Big|\sum_{(i,j)\in\mathcal{I}_r}&\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|\ge\tau\Big|\Lambda\Big)+4\exp\Big(-\frac{\tau^2/2}{N(T-t)d\Lambda+4d\tau/3}\Big),\end{aligned}\tag{15}$$

where $\mathcal{I}_r=\{d\lfloor N(T-t)/d\rfloor+1,...,N(T-t)\}$-th elements of

$$\{(0,t),(0,t+1),...,(0,T),(1,t),(1,t+1),...,(1,T),...,(N-1,t),(N-1,t+1),...,(N-1,T)\}$$

when $d\lfloor NT/d\rfloor<N(T-t)$ and $\mathcal{I}_r=\varnothing$ when $d\lfloor N(T-t)/d\rfloor=N(T-t)$. Suppose $\tau\ge 2Md$, note that $|\mathcal{I}_r|\le d$, it follows that

$$\mathbb{P}\Big(\Big|\sum_{(i,j)\in\mathcal{I}_r}\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|\ge\tau\Big|\Lambda\Big)=0.$$

Under the exponentially β mixing, we have $\beta(d)=O(\rho^d)$ for some positive constant ρ. Set $d=-(c^*+3)\log(NT)/\log\rho$, we obtain $N(T-t)\beta(d)/d=O(B^{-1}N^{-2}T^{-2})$ since $B=O((NT)^{c_1})$ and $T-t=O(T)$. Set $\tau=\max\{\sqrt{8N(T-t)d\Lambda\log(NTB)},32d\log(NTB)/3\}$ and hence

$$\frac{\tau^2}{4}\ge 2N(T-t)d\Lambda\log(NTB)\text{ and }\frac{\tau^2}{4}\ge 8d\tau\log(NTB)/3\text{ and }\tau\ge 2Md$$

as either $N\to\infty$ or $T\to\infty$. Therefore we have $\tau^2/(2N(T-t)d\Lambda+4d\tau/3)\ge 2\log(NTB)$ and hence

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\mathbb{P}\Big(\Big|\sum_{i=1}^{N}\sum_{j=t}^{T-1}\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|\ge 6\tau\Big|\Lambda\Big)=O(B^{-1}N^{-1}T^{-2}).$$

By Bonferroni's equality, we have

$$\mathbb{P}\Bigg(\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\Big|\sum_{i=1}^{N}\sum_{j=t}^{T-1}\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|\ge 6\tau\Big|\Lambda\Bigg)=O(N^{-1}T^{-1}).$$

Thus it follows from Equation 15 that we have

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\Big|\sum_{i=1}^{N}\sum_{j=t}^{T-1}\Big(\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)-\mathbb{E}\widetilde{\phi}_{[t,T]}(S_{i,j},A_{i,j};h_b)\Big)\Big|=O(\max\{\sqrt{NT\Lambda}\log(NTB),\log^2 NTB\}) \tag{16}$$

with probability $1-O(N^{-1}T^{-1})$. Given $T-t=O(T)$ and $B=O((NT)^{c*})$, combining with Equation 14, we have Equation 12.

Proof of (11) with $m=2$: We want to show that

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,...,B\}}}\sqrt{N(T-t)}|R_2|=o_p(\log^{-1/2}(NT)). \tag{17}$$

Define the list

$$\{(1,t),(1,t+1),...,(1,T-1),(2,t),(2,t+1),...,(2,T-1),...,(N,t),(N,t+1),...,(N,T-1)\}.$$

For any $1\le g\le N(T-t)$, denote by (n_g,T_g) the g-th element in the list. Let $\mathcal{F}^{(0)}=\{S_{1,t-1},A_{1,t-1}\}\cup\{S_{j,t'},A_{j,t'},S_{j,t'+1}:t\le t'\le T-1,j\in\mathcal{I}_1\}\cup\{\theta^{(1)}_{b,k}:b=1,...,B,k=1,...,K\}$. Then we recursively define $\mathcal{F}^{(g)}$ as

$$\mathcal{F}^{(g)}=\begin{cases}\mathcal{F}^{(g-1)}\cup\{S_{n_g,T_g},A_{n_g,T_g}\} & \text{if } n_g=n_{g-1}\\ \mathcal{F}^{(g-1)}\cup\{S_{n_g-1,T},S_{n_g,t-1},A_{n_g,t-1}\} & \text{otherwise}\end{cases}$$

Let

$$\begin{aligned}\chi_{g,b,t}=&\text{sgn}(\widehat{\Delta}_{n_g,T_g,b,t})\Big[h_b(S_{n_g,T_g+1})-\mathbb{E}^*_{[t,T]}[h_b(S')|A_{n_g,T_g},S_{n_g,T_g}]\Big]\\ &\times\left[\frac{g(A_{n_g,T_g},S_{n_g,T_g})}{\widehat{\omega}_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})}-\frac{g(A_{n_g,T_g},S_{n_g,T_g})}{\omega^*_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})}\right].\end{aligned}$$

Under MA, R_2 can be written as $\frac{1}{N(T-t)}\sum_{g=1}^{N(T-t)}\chi_{g,b,t}$ and forms a sum of martingale difference sequence with respect to the filtration $\{\sigma(\mathcal{F}^{(g)}):g\ge 0\}$.

Under the boundedness conditions and Markov assumption, we have

$$\begin{aligned}\mathbb{E}\big(\chi^2_{g+1,b,t}|\sigma(\mathcal{F}^{(g)})\big)\le&\mathbb{E}\Big\{\big[h_b(S_{n_g,T_g+1})-\mathbb{E}^*_{[t,T]}[h_b(S')|A_{n_g,T_g},S_{n_g,T_g}]\big]^2\Big\}\\ &*\Big[\frac{g(A_{n_g,T_g},S_{n_g,T_g})}{\widehat{\omega}_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})}-\frac{g(A_{n_g,T_g},S_{n_g,T_g})}{\omega^*_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})}\Big]^2\\ \le&4Q\log^2(NT)\Big[\widehat{\omega}_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})-\omega^*_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})\Big]^2\end{aligned}$$

where $\max_{A,S}\frac{g(A,S)}{\widehat{\omega}_{[t,T]}(A,S)\omega^*_{[t,T]}(A,S)}\le Q$ under boundedness conditions. It follows from Theorem 2.1 of Bercu & Touati (2008) that for any y,τ

$$\mathbb{P}\Bigg(\Big|\sum_{g=1}^{N(T-t)}\chi_{g,b,t}\Big|\ge\tau,\ \sum_{g=1}^{N(T-t)}4Q\log^2(NT)\Big[\widehat{\omega}_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})-\omega^*_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})\Big]^2\le y\Bigg)\le 2\exp\Big(-\frac{\tau^2}{2y}\Big).$$

By Bonferroni's inequality, for any y, τ we have

$$\mathbb{P}\Bigg(\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\left|\sum_{g=1}^{N(T-t)}\chi_{g,b,t}\right|\ge\tau, \max_{T\epsilon\le t\le T(1-\epsilon)}\sum_{g=1}^{N(T-t)}\left[\widehat{\omega}_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})-\omega^*_{[t,T]}(A_{n_g,T_g},S_{n_g,T_g})\right]^2 \le \frac{y}{4Q\log^2(NT)}\Bigg) \le 2BT\exp\left(-\frac{\tau^2}{2y}\right).$$

Set $y=4Q\log^2(NT)(NT)^{-2\kappa_3+1}\log^4(NTB)$, by Equation 21, we obtain

$$\mathbb{P}\left(\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\left|\sum_{g=1}^{N(T-t)}\chi_{g,b,t}\right|\ge\tau\right)\le 2BT\exp\left(-\frac{\tau^2}{8Q\log^2(NT)(NT)^{-2\kappa_3+1}\log^4(NTB)}\right)+o(1). \tag{18}$$

Set $\tau=4\sqrt{Q(NT)^{-2\kappa_3+1}}\log(NT)\log^{5/2}(NTB)$, the RHS of Equation 18 is $O(N^{-1}T^{-1})$. Under Theorem 3.5, we obtain Equation 17.

Proof of (11) with $m=3$:
Define

$$\psi_{i,j,b}=\mathbb{E}^*_{[t,T]}[h_b(S')|A_{i,j},S_{i,j}]-\widehat{\mathbb{E}}_{[t,T]}[h_b(S')|A_{i,j},S_{i,j}].$$

Under similar arguments to Equation 16, we have

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\left|\sum_{i=1}^{N}\sum_{j=t}^{T-1}(\psi^2_{i,j,b}-\mathbb{E}\psi^2_{i,j,b})\right|=O(\max\{\sqrt{NT\Xi}\log(NTB),\log^2 NTB\}) \tag{19}$$

where

$$\max_{i,j,b}\mathbb{E}[\psi^2_{i,j,b}-\mathbb{E}\psi^2_{i,j,b}]^2\le\max_{i,j,b}\mathbb{E}\psi^4_{i,j,b}\equiv\Xi\le 16\log^4(NT)\mathbb{E}\left\{d^4_{TV}\left(\widehat{p}_{[t,T]}(S',R|S,A),p^*_{[t,T]}((S',R|S,A)\right)\right\}.$$

By triangular inequality, Theorem 3.5 and boundedness condition, we have

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\sum_{i=1}^{N}\sum_{j=t}^{T-1}\left\{\mathbb{E}^*_{[t,T]}[h_b(S',R)|A_{i,j},S_{i,j}]-\widehat{\mathbb{E}}_{[t,T]}[h_b(S',R)|A_{i,j},S_{i,j}]\right\}^2=O((NT)^{-2\kappa_1+1}\log^2(NT)). \tag{20}$$

Similar to Equation 20, we have

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\sum_{i=1}^{N}\sum_{j=t}^{T-1}\left\{\omega^*_{[t,T]}(A_{i,j},S_{i,j})-\widehat{\omega}_{[t,T]}(A_{i,j},S_{i,j})\right\}^2=O((NT)^{-2\kappa_3+1}\log^2(NT)).$$

Together with Equation 20 and Equation 21, by Cauchy-Schwarz inequality,

$$\max_{\substack{T\epsilon\le t\le T(1-\epsilon)\\ b\in\{1,\dots,B\}}}\sqrt{N(T-t)}|R_3(t,b)|=O((NT)^{-\kappa_1-\kappa_3+1/2}\log^2(NT))=O(\log^{-1/2}(NT)). \tag{21}$$

Step 3
In this step, we show the limiting distribution of Γ^*. We begin by defining vectors $\lambda^*_{i,j,t}\in\mathbb{R}^B$, where the b-th element is

$$\frac{\mathbf{I}(j\ge t)}{\sqrt{N(T-t)}}\text{sgn}\left(\widehat{\Delta}_{i,j,b,t}\right)\left[h_b(S_{i,j+1})-\mathbb{E}^*_{[t,T]}[h_b(S')|A_{i,j},S_{i,j}]\right]\frac{g(A_{i,j},S_{i,j})}{\omega^*_{[t,T]}(A_{i,j},S_{i,j})} -\frac{\mathbf{I}(j<t)}{\sqrt{Nt}}\text{sgn}\left(\widehat{\Delta}_{i,j,b,t}\right)\left[h_b(S_{i,j+1})-\mathbb{E}^*_{[0,t]}[h_b(S')|A_{i,j},S_{i,j}]\right]\frac{g(A_{i,j},S_{i,j})}{\omega^*_{[0,t]}(A_{i,j},S_{i,j})}\Bigg\}. \tag{22}$$

Define $\tau = \{\lceil \epsilon T \rceil, \lceil \epsilon T \rceil + 1, ..., \lfloor (1-\epsilon)T \rfloor\}$, which is all the values t can take. Define $\lambda^*_{i,j} = (\lambda^*_{i,j,\lceil \epsilon T \rceil}, \lambda^*_{i,j,\lceil \epsilon T \rceil+1}, ..., \lambda^*_{i,j,\lfloor (1-\epsilon)T \rfloor})$ as a vector with $B|\tau|$ dimensions. Define the list

$$\{(1,0),(1,1),...,(l_1,T-1),(2,0),(2,1),...,(l_2,T-1),...,(N,0),(N,1),...,(N,T-1)\}.$$

For any $1 \le g \le NT$, denote by (n_g, T_g) the g-th element in the list. Let $\mathcal{F}^{(0)} = \{S_{1,0}, A_{1,0}\} \cup \{S_{j,t'}, A_{j,t'}, S_{j,t'+1} : t \le t' \le T-1, j \in \mathcal{I}_1\} \cup \{\theta^{(1)}_{b,k} : b = 1, ..., B, k = 1, ..., K\}$. Then we recursively define $\mathcal{F}^{(g)}$ as

$$\mathcal{F}^{(g)} = \begin{cases} \mathcal{F}^{(g-1)} \cup \{S_{n_g,T_g}, A_{n_g,T_g}\} & \text{if } n_g = n_{g-1} \\ \mathcal{F}^{(g-1)} \cup \{S_{n_g-1,T}, S_{n_g,0}, A_{n_g,0}\} & \text{otherwise} \end{cases}$$

Under MA, the high-dimensional vector $\sum_{g=1}^{NT} \sqrt{(T-t_g)t_g/T^2}\lambda^*_{n_g,t_g}$ forms a sum of martingale difference sequence with respect to the filtration $\{\sigma(\mathcal{F}^{(g)}) : g \ge 0\}$. Note that $\Gamma^* = \| \sum_{g=1}^{NT} \sqrt{(T-t_g)t_g/T^2}\lambda^*_{n_g,t_g}\|_\infty$. For each g, define $\Sigma_g = \mathbb{E}(((T-t_g)t_g/T)\lambda^*_{n_g,t_g}\lambda^{*T}_{n_g,t_g}|\sigma(\mathcal{F}^{(g-1)}))$, $V^* = \sum_{g=1}^{NT}\Sigma_g$ and $V_0 = \mathbb{E}V^*$. Similar to the proof of Equation 19, we can show that $\|V^* - V_0\|_{\infty,\infty}$ is absolutely bounded by $O((NT)^{-1/2}\log^3(NT))$ with probability $1 - O(N^{-1}T^{-1})$. By Theorem 3.1 of Belloni & Oliveira (2018), we have for any Borel set $\mathcal{R}$, any $\delta > 0$ and some constant $C > 0$

$$\begin{aligned}\mathbb{P}(\Gamma^* \in \mathcal{R}) &\le \mathbb{P}(\|N(0,V_0)\|_\infty \in \mathcal{R}^{C\delta}) \\ &+ C\Big(\frac{1}{NT} + \frac{\log^3(NT)\log(B|\tau|)}{\delta^2\sqrt{NT}} + \frac{\log^3(B|\tau|)}{\delta^3\sqrt{NT}} + \frac{\log^2(B|\tau|)}{\delta^3}\sum_{g=1}^{N(T-1)}\mathbb{E}\|\eta_g\|^3_\infty\Big) \end{aligned} \tag{23}$$

where $\eta_g = \Sigma_g^{1/2}N_g, 1 \le g \le NT$ and $\{N_g\}_{g=1,...,N(T-1)}$ are i.i.d. standard $B|\tau|$-dimensional Gaussian random vectors defined in the same probability as $\lambda^*_{n_g,t_g}$. By the boundedness conditions, we have each element in Σ_g is bounded by $4log^2(NT)(NT)^{-1}$ and $\mathbb{E}\|N_g\|^3_\infty = O(log^{3/2}(B|\tau|))$. Therefore, we have $\sum_{g=1}^{NT}\mathbb{E}\|\eta_g\|^3_\infty = O((NT)^{-1/2}\log^{9/2}(NT))$. Combined all the arguments together, we have

$$\mathbb{P}(\Gamma^* \in \mathcal{R}) \le \mathbb{P}(\|N(0,V_0)\|_\infty \in \mathcal{R}^{C\delta}) + O(1)\Big(\frac{1}{NT} + \frac{log^4(NT)}{\delta^2\sqrt{NT}} + \frac{\log^3(NT)}{\delta^3\sqrt{NT}} + \frac{\log^{13/2}(NT)}{\delta^3\sqrt{NT}}\Big). \tag{24}$$

Let $\mathcal{R} = (z, \infty)$ and $\delta = \epsilon\log^{-1/2}(NT)/C$, then

$$\mathbb{P}(\Gamma^* \le z) \ge \mathbb{P}(\|N(0,V_0)\|_\infty \le z - \epsilon\log^{-1/2}(NT)) - o(1). \tag{25}$$

Let $\mathcal{R} = (-\infty, z]$ and $\delta = \epsilon\log^{-1/2}(NT)/C$, then

$$\mathbb{P}(\Gamma^* \le z) \ge \mathbb{P}(\|N(0,V_0)\|_\infty \le z + \epsilon\log^{-1/2}(NT)) + o(1). \tag{26}$$

Together with $\widehat{\Gamma} = \Gamma^* + o_p((NT)^{-1/2}\log^{-1/2}(NT))$ from step 2, we have for any constant $\epsilon > 0$,

$$\begin{aligned}\mathbb{P}(\widehat{\Gamma} \le z) &\ge \mathbb{P}(\|N(0,V_0)\|_\infty \le z - 2\epsilon\log^{-1/2}(NT)) - o(1), \\ \mathbb{P}(\widehat{\Gamma} \le z) &\ge \mathbb{P}(\|N(0,V_0)\|_\infty \le z + 2\epsilon\log^{-1/2}(NT)) + o(1).\end{aligned} \tag{27}$$

Step 4

In this step, we show that $\|\widehat{V} - V_0\|_{\infty,\infty}$ is uniformly bounded by $O((NT)^{-c_2})$ with some positive constant $c_2 > 0$, where $\widehat{V}$ is the estimated version of V_0 by replacing $p^*_{[t_1,t_2]}, \omega^*_{[t_1,t_2]}$ with estimated ones. By Cauchy-Schwarz inequality, $\|\widehat{V} - V_0\|_{\infty,\infty}$ $\le \|\widehat{V} - V^*\|_{\infty,\infty} + \|V^* - V_0\|_{\infty,\infty}$. Since we have shown that $\|V^* - V_0\|_{\infty,\infty}$ is bounded by $O((NT)^{-1/2}\log^3(NT))$ with probability $1 - O(N^{-1}T^{-1})$, we will show that $\|\widehat{V} - V^*\|_{\infty,\infty} = O_p((NT)^{-c_3})$ for some constant $c_3 > 0$.

$$\|\widehat{V} - V^*\|_{\infty,\infty} = \max_{\substack{t_1,t_2\in\tau \\ b_1,b_2\in\{1,...,B\}}} |\sum_{i=1}^{N}\sum_{j=0}^{T-1}\widehat{\lambda}_{i,j,t_1,b_1}\widehat{\lambda}_{i,j,t_2,b_2} - \lambda^*_{i,j,t_1,b_1}\lambda^*_{i,j,t_2,b_2}|. \tag{28}$$

Since $\widehat{a}\widehat{b} - a^*b^* = \frac{1}{2}\left[\widehat{a}(\widehat{b}-b^*) + a^*(\widehat{b}-b^*) + \widehat{b}(\widehat{a}-a^*) + b^*(\widehat{a}-a^*)\right]$, following the similar steps in step 1, we can show that $\|\widehat{V} - V^*\|_{\infty,\infty}$ is bounded by $O((NT)^{-c_3})$ with probability $1 - O(N^{-1}T^{-1})$.

Step 5

In this step, we finish the proof by showing that $|\mathbb{P}(\widehat{\Gamma} \leq z) - \mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z|\widehat{V})| = o(1)$.

Since $\|\widehat{V} - V^*\|_{\infty,\infty}$, using the similar steps in step 3, we can show that for any $\epsilon > 0$,

$$\begin{aligned}
\mathbb{P}(\widehat{\Gamma} \leq z) &\geq \mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z - 2\epsilon \log^{-1/2}(NT)|\widehat{V}) - o(1), \\
\mathbb{P}(\widehat{\Gamma} \leq z) &\geq \mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z + 2\epsilon \log^{-1/2}(NT)|\widehat{V}) + o(1).
\end{aligned} \tag{29}$$

By the condition from Theorem 3.6 that the diagonal element of V_0 is uniformly bounded below by some $\zeta > 0$ and $\|\widehat{V} - V_0\|_{\infty,\infty} = O_p((NT)^{-c_2})$, by the Theorem 1 of Chernozhukov et al. (2017), we can show that

$$\begin{aligned}
\mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z + 2\epsilon \log^{-1/2}(NT)|\widehat{V}) - \mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z - 2\epsilon \log^{-1/2}(NT)|\widehat{V}) \\
\leq O(1)\epsilon \log^{1/2}(B|\tau|) \log^{-1/2}(NT).
\end{aligned} \tag{30}$$

Together with Equation 29, we have

$$|\mathbb{P}(\widehat{\Gamma} \leq z) \geq \mathbb{P}(\|N(0, \widehat{V})\|_\infty \leq z|\widehat{V})| = o(1),$$

since $\epsilon > 0$ can be arbitrary small. Replacing z with $\widehat{c}_\alpha$, which is α percentile generated by Gaussian multiplier bootstrap (Chernozhukov et al., 2013), we finish the proof. □