

# Information Relaxation and A Duality-Driven Algorithm for Stochastic Dynamic Programs

Nan Chen<sup>\*</sup>      Xiang Ma<sup>†</sup>      Yanchu Liu<sup>‡</sup>      Wei Yu<sup>§</sup>

This Version: July 24, 2020  
First Version: July 8, 2019

## Abstract

We use the technique of information relaxation to develop a duality-driven iterative approach to obtaining and improving confidence interval estimates for the true value of finite-horizon stochastic dynamic programming problems. We show that the sequence of dual value estimates yielded from the proposed approach in principle monotonically converges to the true value function in a finite number of dual iterations. Aiming to overcome the curse of dimensionality in various applications, we also introduce a regression-based Monte Carlo algorithm for implementation. The new approach can be used not only to assess the quality of heuristic policies, but also to improve them if we find that their duality gap is large. We obtain the convergence rate of our Monte Carlo method in terms of the amounts of both basis functions and the sampled states. Finally, we demonstrate the effectiveness of our method in an optimal order execution problem with market friction and in an inventory management problem in the presence of lost sale and lead time. Both examples are well known in the literature to be difficult to solve for optimality. The experiments show that our method can significantly improve the heuristics suggested in the literature and obtain new policies with a satisfactory performance guarantee.

KEYWORDS: stochastic dynamic programming; information relaxation; duality; regression based Monte Carlo method; optimal execution; inventory management.

## 1 Introduction

Stochastic dynamic programming (SDP) provides a powerful framework for modeling and solving decision-making problems under a random environment in which uncertainty is resolved and actions are taken sequentially over time. Recently it also has become increasingly important to help us understand the general principle behind reinforcement learning,

---

<sup>\*</sup>Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N. T., Hong Kong. Email: nchen@se.cuhk.edu.hk

<sup>†</sup>Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N. T., Hong Kong. Email: xma@se.cuhk.edu.hk

<sup>‡</sup>Department of Finance, Lingnan (University) College, Sun Yat-sen University, Guangzhou, China. Email: liuych26@mail.sysu.edu.cn

<sup>§</sup>World Quant (Singapore), 1 Wallich Street, #20-01 Guoco Tower, Singapore. Email: wei.yu@worldquant.com

a rapidly developing area of artificial intelligence. The Bellman backward recursion fully characterizes the structure of the optimal policies of an SDP problem. However, hampered by the curse of dimensionality, it is practically infeasible to implement this principle of optimality to derive the solutions for many high dimensional applications. Hence, people often have to settle for a suboptimal control policy that strikes a reasonable balance between convenient implementation and adequate performance. This practice naturally gives rise to the following two research questions:

1. How can we assess the quality of a given control policy?
2. If we know the performance of a policy is not satisfactory, do we have a systematic way to improve it?

Motivated by these two questions, especially the second one, we develop in this paper a duality-driven iterative approach to obtaining and improving confidence interval estimates for the true value of an SDP problem with finite time horizon. This new approach stems from information relaxation and the corresponding dual formulation in the SDP literature. Take a cost minimization problem as an example. Within the dual framework laid out in Brown, Smith, and Sun (2010), we relax the admissible constraint that requires policies to be dependent only upon the information up to the moment when a decision is made, and meanwhile impose a penalty in the problem’s objective function that punishes any violations of the admissible constraint. This two-step construction results in a lower bound on the optimal expected cost.

The above duality bounds enable us to assess the performance of a candidate policy. Fixing the policy we are interested in assessing, we can use standard simulation techniques to estimate the expected costs under this policy (refer to, for example, Powell (2011) for other related statistical learning approaches for policy evaluation). Note that every policy is suboptimal and thus produces a value higher than the optimal cost. If the difference, referred to as the duality gap hereafter, between the expected value of this policy and the aforementioned lower bound from the dual formulation is tight, we can assert that the policy must be close to the optimality. A variety of applications of this duality based policy assessment can be found, just to name a few, in Lai, Margot and Secomandi (2010) and Lai et al. (2011) for natural gas storage valuation, Brown and Smith (2011), Haugh and Wang (2014), and Haugh, Iyengar and Wang (2016) for dynamic portfolio investment, Brown, Smith, and Sun (2010) and Brown and Smith (2014) for inventory management, Goodson, Ohlmann and Thomas (2013) for multi-vehicle routing, Brown and Smith (2014) for revenue management, Kim and Lim (2016) for robust multi-armed bandits, Devalkar, Anupindi and Sinha (2011) for an integrated optimization problem of procurement, processing, and trade of commodities, Balseiro, Brown and Chen (2018) for stochastic scheduling problems, Balseiro and Brown (2019) for stochastic knapsack problems, stochastic scheduling on parallel machines, and sequential search problems, and most recently, Brown and Smith (2020) for dynamic selection problems.

Complementing the applications of the SDP duality in policy assessments, the primary focus of our work is how to improve a candidate policy if we find that its duality gap is not small. The paper makes two contributions to the literature on SDP duality. First, we propose a new duality-driven dynamic programming (DDP) algorithm that is capable of iteratively

improving the estimates of the dual value to an SDP problem. In each iteration, the algorithm utilizes the dual values from the last iteration as inputs to construct the penalty and then outputs new dual values for the next round. We manage to show that the sequence of lower bound estimates that result from the proposed algorithm monotonically converges from below to the true value function of an SDP problem with a cost minimization objective. More importantly, for problems with a finite time horizon, we also prove that such convergence will be accomplished in a finite number of dual iterations and the optimal control can thereby be obtained on the basis of the dual value function that is output at the termination of the DDP algorithm. With these important theoretical underpinnings, the new algorithm systematizes the improvement of a policy with a large duality gap, which addresses the second issue imposed at the beginning of the paper that remains largely unanswered in the SDP duality literature. We demonstrate this convergence result by applying the DDP algorithm to the linear-quadratic control (LQC) problem, one of the most fundamental problems in control theory. Corroborating the above theoretical discovery, the calculation reveals that, from a suboptimal policy, our DDP algorithm can yield the optimal linear policy within just two dual iterations.

The second contribution of this paper is that we present a high-dimensional numerical implementation approach for DDP and develop its related performance guarantee. To overcome the curse of dimensionality in the high-dimensional setup, we combine the regression architecture with Monte Carlo simulation to extrapolate the dual estimates observed on the sampled states to the entire state space for approximating dual functions in each iteration of the DDP algorithm. The dual bound yielded from this algorithm can help us build up effective confidence interval estimates on the value of the SDP problem, from which we can determine the optimality of the improved policy. Though the approach shares some common features with the existing simulation and approximation methods in the study of approximate dynamic programming (see, e.g., Bertsekas and Tsitsiklis (1996), Longstaff and Schwartz (2001), Tsitsiklis and Van Roy (1999, 2001), Powell (2011)), the special structure of the dual formulation distinguishes it from the others in several key aspects:

- Compared with the Monte Carlo duality in American option pricing (see, e.g., Rogers (2002), Haugh and Kogan (2004), Andersen and Broadie (2004), Chen and Glasserman (2007), and Desai, de Farias, and Moallemi (2012b)), one additional layer of complexity in dealing with a general dynamic program is that the policies taken by the decision maker will affect the evolution of the underlying system. This leads us to face the challenging tradeoff between exploration and exploitation when we try to numerically implement the DDP algorithm; see the counterexample in Appendix D.2. To avoid the exploration pitfall, we introduce a device called a state sampler into our Monte Carlo approach and analyze its role in determining the convergence of the method.
- To determine the dual value in each iteration, the DDP algorithm requires solving an optimization problem before taking expectation. Along one sample path of randomness, such an optimization problem is deterministic. This salient characteristic is in stark contrast to the classical value iteration algorithm widely used in dynamic programming where one has to solve stochastic programs to optimize an expected value. As shown in the discussion on the LQC problem (Sec. 3.2) and the numerical examples

(Sec. 5), the vast research base of deterministic optimization enables us to have a high degree of flexibility in choosing effective numerical procedures for our DDP algorithm.

- Another advantage of solving optimization inside expectation is that it allows us to deploy parallel computing to accelerate the execution of the DDP algorithm. In particular, we can simulate different groups of sample paths in parallel processors and solve the corresponding optimization programs simultaneously; then we can take the average across all the outcomes collected from the parallel processors to compute the dual values. The parallelization grants scalability to the DDP algorithm.

To develop a performance guarantee for the above regression-based simulation approach, we characterize its rate of convergence to the true value in terms of the amounts of both basis functions for the purpose of function approximation and the sampled states on which the dual values are estimated. Our analysis reveals an intriguing trade-off between model complexity and simulation efforts. More specifically, the number of sampled states should be proportionally sufficient relative to the number of basis functions; otherwise, the effect of model overfitting may cause the outcome from the DDP algorithm to diverge, rather than converge, even if both amounts tend to infinity. The paper quantifies a relative growth order between the numbers of the sampled states and basis functions as a sufficient condition to warrant the convergence.

We demonstrate the effectiveness of our DDP algorithm with two numerical examples. One is about portfolio execution (a variant of Bertsimas and Lo (1998)) and the other is about inventory management (Zipkin (2008a,b)). Both examples are widely known in the literature to be intractable due to the constraints imposed on the policies and the complex high-dimensional dynamics. Using the above DDP algorithm, we significantly improve a variety of conventional heuristics suggested in the literature, such as lookahead and linear programming approximation, to yield new policies with satisfactory performance. It is worthwhile mentioning that, aiming at the convex structure in these examples, we apply difference-of-convex (DC) programming to solve the inner optimization problem in their dual formulation. The tightness of the resulted confidence intervals strongly indicates this programming technique works very effectively for convex control problems.

As noted earlier, the paper extends and complements the literature on information relaxation and SDP dualities initiated by Brown, Smith, and Sun (2010). Along this research line, Brown and Smith (2014) consider dynamic programs that have a convex structure and use the first-order linear approximations of value functions to construct gradient penalties that can provide tight bounds. Brown and Haugh (2017) and Ye and Zhou (2015) generalize the information relaxation approach for calculating performance bounds for infinite horizon Markov decision processes and continuous-time controls, respectively. Desai, de Farias, and Moallemi (2013) compare the duality in the perfect information relaxation (called martingale duality in their paper) with the approximate linear programming approach in the literature (e.g., Schweitzer and Seidmann (1985), de Farias and Van Roy (2003, 2004)). They find that the former one can produce tighter lower bounds on the optimal cost-to-go function of a Markov decision problem. More recently, Haugh and Ruiz-Lacedelli (2018) derive the information relaxation bounds to Markov decision processes with partial observations.

To the best of our knowledge, the idea of information relaxation based duality can be dated back to Rockafellar and Wets (1976), who show the possibility of associating with the

non-anticipative requirement on the solution of a multi-stage stochastic program a Lagrange multiplier that satisfies a martingale property. Davis (1989, 1991) and Davis and Zervos (1995) also find that introducing appropriate Lagrange multiplier terms in the objective function of an LQC problem and solving the corresponding pathwise optimization problem will lead to the optimal controls for the original problem. Later, Rogers (2007) represents the value function of a discrete-time controlled Markov process in a dual Lagrangian form with the help of measure-change arguments and the perfect information relaxations.

A lot of interesting theoretical results, such as weak and strong dualities under various setups, have been established by the aforementioned papers. People especially find that the dual value should be identical to the true value of the original SDP problem for an optimally chosen penalty — the strong duality relation. However, solving for this optimal penalty is not easy. Thus the existing literature typically heuristically selects “good” martingale penalty functions and numerically examine its quality. Contributing to this literature, the DDP algorithm presents a systematic approach to iteratively construct the optimal duality.

As a special case of the general SDP problem, Rogers (2002), Haugh and Kogan (2004), Andersen and Broadie (2004), and Desai, de Farias, and Moallemi (2012b) investigate the dual representation of American option pricing and more generally the optimal stopping problem. In particular, Chen and Glasserman (2007) discuss how to improve the dual bounds on the option prices iteratively. However, what differentiates the case of American option pricing or more broadly optimal stopping from a general SDP problem is that the state transition probabilities in the former case generally do not depend on the exercising actions taken by the option holder. In this sense, our paper extends the study of Chen and Glasserman (2007) to a general setup of dynamic programming.

The remainder of the paper is organized as follows. In Section 2, we review the basic duality results developed by Brown, Smith, and Sun (2010). We develop the theory underpinning the DDP algorithm in Section 3 and illustrate how it works using the LQC problem as an example. Section 4 is devoted to the regression-based Monte Carlo simulation implementation and the related convergence analysis. Section 5 presents two numerical experiments. All the proofs and some supplementary discussions are deferred to the AppendixAppendix.

## 2 The Dual Formulation of an SDP Problem

To fix the idea, we consider a generic finite-horizon discrete-time SDP problem in a probability space  $(\Omega, \mathcal{F}, \mathbb{P})$ . Suppose that a planner makes sequential control decisions on a system over a  $T$ -period time horizon indexed by  $t = 0, 1, \dots, T$ . At the beginning of each time period  $t$ , given the system state  $x_t \in \mathbb{R}^n$ , she takes an action  $a_t \in A_t \subseteq \mathbb{R}^m$ , where  $A_t$  is the set of all feasible actions at that moment. A random vector  $\xi_t : \Omega \rightarrow \mathbb{R}^d$  will materialize during the period. To make the problem Markovian, we assume that all  $\xi_t$ 's are independent. The purpose of this assumption is only for notational simplicity. Most of the subsequent results still hold when we generalize the discussion to non-Markovian cases in which the probability distribution of  $\xi_t$  may depend on the whole trajectories of  $\{\xi_0, \dots, \xi_{t-1}\}$  and  $\{x_0, \dots, x_t\}$ . The planner then incurs a cost amounting to  $r_t$  that may be dependent on  $x_t$ ,  $a_t$ , and  $\xi_t$ . The system evolves to a new state according to the following recursive dynamic

$$x_{t+1} = f_t(x_t, a_t, \xi_t) \tag{1}$$

and the next round of decision making starts. Here  $f_t$ ,  $t = 0, 1, \dots, T - 1$ , is a function from  $\mathbb{R}^n \times A_t \times \mathbb{R}^d$  to  $\mathbb{R}^n$ , mapping the current state, the selected action, and the realized randomness to another state. The planner attempts to minimize the expected aggregate costs

$$\mathbb{E} \left[ \sum_{t=0}^{T-1} r_t(x_t, a_t, \xi_t) + r_T(x_T) \middle| x_0 \right] \quad (2)$$

in this process by taking proper actions, where  $r_T(x_T)$  stands for the terminal cost received at the end of the planning horizon.

We call  $\alpha = (\alpha_0, \dots, \alpha_{T-1})$  a *policy* if each argument  $\alpha_t$  of it is a function from  $\Omega$  to  $A_t$ ,  $t = 0, \dots, T - 1$ . In other words, a policy prescribes the rule of action selection for the planner for each possible outcome  $\omega$  in  $\Omega$  in each period. To reflect the information constraint that the planner faces, assume that she cannot peek into the future of the system dynamics. Hence, the decision that she makes in period  $t$  relies only on what is known about the past trajectory of the system at the beginning of the period. More formally, letting  $\mathcal{F}_t = \sigma(x_0, \dots, x_t)$  be the  $\sigma$ -algebra generated by the information about the system states up to time  $t$ , we require the planner's policy to be *admissible* in the sense that  $\alpha_t$  is  $\mathcal{F}_t$ -measurable for all  $0 \leq t \leq T - 1$ . Denote  $\mathbb{F} = (\mathcal{F}_0, \mathcal{F}_1, \dots, \mathcal{F}_{T-1})$  with  $\mathcal{F}_0 = \{\emptyset, \Omega\}$ . The objective of the decision maker can then be formulated as optimizing

$$V_0(x) = \inf_{\alpha \in \mathcal{A}_{\mathbb{F}}} \mathbb{E} \left[ \sum_{t=0}^{T-1} r_t(x_t, \alpha_t, \xi_t) + r_T(x_T) \middle| x_0 = x \right], \quad (3)$$

where  $\mathcal{A}_{\mathbb{F}}$  denotes the collection of all admissible policies with respect to the information filtration  $\mathbb{F}$ .

It is well known that we may invoke the *principle of dynamic programming* (or the Bellman equation) to solve the above SDP problem (3). Let  $V_t(x)$  be the cost-to-go function of the system from time  $t$  onward; that is,

$$V_t(x) = \inf_{\alpha \in \mathcal{A}_{\mathbb{F}}|t} \mathbb{E} \left[ \sum_{s=t}^{T-1} r_s(x_s, \alpha_s, \xi_s) + r_T(x_T) \middle| x_t = x \right], \quad (4)$$

where

$$\mathcal{A}_{\mathbb{F}}|t = \left\{ \alpha = (\alpha_t, \dots, \alpha_{T-1}) : \alpha_s \text{ is } \mathcal{F}_s\text{-measurable for all } t \leq s \leq T - 1 \right\}.$$

The Bellman equation dictates that we can determine the value of  $V_t$  in a backward fashion:

$$V_T(x) = r_T(x); \quad (5)$$

$$V_t(x) = \inf_{a_t \in A_t} \mathbb{E} [r_t(x, a_t, \xi_t) + V_{t+1}(f_t(x, a_t, \xi_t))] \quad (6)$$

for all  $t = 0, \dots, T - 1$  and  $x \in \mathbb{R}^n$ . The expectation in (6) is taken with respect to the probability distribution of  $\xi_t$ . Furthermore, if  $a_t^* = \alpha_t^*(x)$  minimizes the right hand side of (6) for each  $x$  and  $t$ , the policy  $\alpha^* = (\alpha_0^*, \dots, \alpha_{T-1}^*)$  is optimal.

However, the curse of dimensionality prevents us from directly utilizing the Bellman equations (5-6) to solve the SDP problem because the computational complexity that this procedure incurs grows exponentially as the dimensionality of the state, randomness, and action spaces increase; see, e.g., Sections 1.2 and 4.1 in Powell (2011) for detailed discussions on this issue. In light of this difficulty, people often have to settle for a computationally tractable approximate (thus, suboptimal) policy of adequate performance. This gives rise to a natural question about how to assess such approximate policies without knowing where the optimality is. As noted in the introduction, the dual formulation proposed in Brown, Smith, and Sun (2010) presents a systematic approach by which we can measure the quality of a suboptimal policy, or in other words, how close it is to the optimal one.

The key ingredients of their duality are the concept of *information relaxation* and a related *penalty*. For the purpose of this paper, we only consider the case of perfect relaxation and refer readers to their paper for a rigorous development of the dual theory under a general framework. Intuitively, if we relax the requirement of information admissibility on policies by allowing the decision maker to take actions after she observes the entire realization of randomness  $(\xi_1, \dots, \xi_T)$ , we should be able to obtain a lower bound to the true cost value  $V_0$ . More precisely, by Jensen's inequality, we have

$$\mathbb{E} \left[ \inf_{a \in A} \left( \sum_{s=0}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) \right) \middle| x_0 = x \right] \leq V_0(x) \quad (7)$$

for all  $x$ . Note that the minimizer of the optimization inside the expectation on the left hand side of (7) is not admissible in the original problem because it may depend on the whole trajectory of  $(\xi_1, \dots, \xi_T)$ .

Brown, Smith, and Sun (2010) further points out that we can achieve equality in (7) if properly penalizing the objective function inside the expectation. Corresponding to the above perfect relaxation, one possible penalty can be constructed as follows. Let  $W = (W_1(\cdot), \dots, W_T(\cdot))$  be any sequence of functions such that each argument  $W_t : \mathbb{R}^n \rightarrow \mathbb{R}$  maps the system state to real numbers. Given an action sequence  $a = (a_0, \dots, a_{T-1}) \in A := A_0 \times \dots \times A_{T-1}$  and a sequence of randomness  $\xi = (\xi_0, \dots, \xi_{T-1})$ , we can use Eq. (1) to recursively generate a trajectory of system states  $(x_1, \dots, x_T)$ . Along it, define a penalty function such as

$$z(a, \xi) = \sum_{t=0}^{T-1} \{ \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(f_t(x_t, a_t, \xi_t))] - (r_t(x_t, a_t, \xi_t) + W_{t+1}(f_t(x_t, a_t, \xi_t))) \}, \quad (8)$$

where the expectation inside the sum is taken with respect to the distribution of  $\xi_t$ . Then, Brown, Smith, and Sun (2010) show that

$$V_0(x) = \sup_W \mathbb{E} \left[ \inf_{a \in A} \left( \sum_{s=0}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z(a, \xi) \right) \middle| x_0 = x \right]. \quad (9)$$

The strong duality relationship (9) paves a useful way to assessing the quality of a specific admissible policy  $\alpha$ . First, we may evaluate the policy by calculating

$$\bar{V}_t(x) = \mathbb{E} \left[ \sum_{s=t}^{T-1} r_s(x_s, \alpha_s, \xi_s) + r_T(x_T) \middle| x_t = x \right] \text{ for all } 0 \leq t \leq T.$$

Surely  $\bar{V}_t(x) \geq V_t(x)$  for any  $x$  because of the sub-optimality of  $\alpha$ . Then, we replace the generic  $W$  in (8) by  $\bar{V}_t$  to construct a penalty  $z$  and compute the associated dual value

$$\underline{V}_0(x) = \mathbb{E} \left[ \inf_{a \in A} \left( \sum_{s=0}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z(a, \xi) \right) \middle| x_0 = x \right].$$

From (9), we have  $V_0(x) \geq \underline{V}_0(x)$ , which implies

$$0 \leq \bar{V}_0(x) - V_0(x) \leq \bar{V}_0(x) - \underline{V}_0(x).$$

When the dual gap  $\bar{V}_0 - \underline{V}_0$  is sufficiently tight, we can conclude that the performance of policy  $\alpha$  must be very close to the optimality. One can refer to those works mentioned in the introduction for various applications of the above duality-based policy assessment.

### 3 DDP: A Duality-Driven Dynamic Programming Method

Beyond the aforementioned policy assessment, the primary interest of the current paper is on the second research question posed in the introduction: can we develop a systematic approach to improving the policy in hand if we find that its dual gap is not tight enough? In this section, we build up an iterative method on the basis of the SDP information duality to achieve the goal of policy improvement.

#### 3.1 Subsolutions and Dual Value Iteration

Central to our investigation are the notion of *subsolution* and, more importantly, its close relationship with the information duality.

**Definition 3.1 (subsolution)** *A functional sequence  $S = (S_0, S_1, \dots, S_T)$  with  $S_t : \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $0 \leq t \leq T$ , is called a subsolution to the problem (3) if it satisfies*

$$S_t(x) \leq \inf_{a_t \in A_t} \mathbb{E} [r_t(x, a_t, \xi_t) + S_{t+1}(f_t(x, a_t, \xi_t))]$$

for any  $t = 0, 1, \dots, T - 1$  and  $x \in \mathbb{R}^n$  with the convention that  $S_T(x) = r_T(x)$ .

The concept of subsolutions to a generic SDP problem has been long known in the literature; one may see, for instance, Theorem 6.2.2 in Putman (1994) or Theorem 3.4.1 in Powell (2011). It just generalizes the Bellman equation (cf. (6)) by replacing the equality with an inequality. One well-known fact is that any subsolution provides a lower bound on the true value of the primal problem (3) (e.g., Theorem 6.2.2 in Putman (1994)). Using the subsolution requirement on each state as the constraints, de Farias and Van Roy (2003) developed a linear programming based approach to approximate solutions to the SDPs. Let  $\mathcal{S}$  denote the collection of all the subsolutions to the problem (3).

As one of the key underpinnings of our DDP algorithm, Proposition 3.2 points out that the dual operation actually offers us a way to construct subsolutions. Introducing some operator notations here will help us present the main results in a compact way. Take any



functional sequence  $W = (W_0(\cdot), \dots, W_T(\cdot))$  and consider the tail subproblem (4) for each  $t$ ,  $0 \leq t \leq T - 1$ . Note that it is still an SDP problem. Hence, we can apply the corresponding dual formulation to it, namely, construct the associated penalty

$$z_t(a, \xi) = \sum_{s=t}^{T-1} \{ \mathbb{E}[r_s(x_s, a_s, \xi_s) + W_{s+1}(f_s(x_s, a_s, \xi_s))] - (r_s(x_s, a_s, \xi_s) + W_{s+1}(f_s(x_s, a_s, \xi_s))) \}, \quad (10)$$

by using the tail sequence of  $W$ ,  $(W_{t+1}(\cdot), \dots, W_T(\cdot))$ , and obtain the dual function

$$W'_t(x) := \mathbb{E} \left[ \inf_{a \in A|t} \left( \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t(a, \xi) \right) \middle| x_t = x \right] \quad (11)$$

for each  $t$ , where  $A|t = A_t \times \dots \times A_{T-1}$ . In this way, as implied by the duality theory discussed in the last section, we reach a sequence of lower bounds  $W' = (W'_0(\cdot), \dots, W'_T(\cdot))$  to the true cost value of every tail problem. From now on, let  $\mathcal{D}$  denote the dual operator defined through (10-11) that can be viewed as acting on any functional sequence  $W$  to produce another function sequence  $\mathcal{D}W = ((\mathcal{D}W)_0, \dots, (\mathcal{D}W)_T)$ , where  $(\mathcal{D}W)_t(x) = W'_t(x)$  for  $0 \leq t \leq T - 1$  and  $(\mathcal{D}W)_T(x) = r_T(x)$ .

Examining the relationship among  $(\mathcal{D}W)_t$  across all  $t$ 's, we have

**Proposition 3.2** *Let  $W = (W_0, W_1, \dots, W_T)$  be any functional sequence. Then,  $\mathcal{D}W \in \mathcal{S}$ , i.e., for all  $t$ ,*

$$\mathcal{D}W_t(x) \leq \inf_{a_t \in A_t} \mathbb{E}[r_t(x, a_t, \xi_t) + \mathcal{D}W_{t+1}(f_t(x, a_t, \xi_t))].$$

This proposition reveals that the information relaxation based duality and the subsolutions are closely related. In particular, the former presents a systematic way of constructing the latter. This finding is new to the existing literature to the best of our knowledge. Moreover, Proposition 3.2 indicates that, if we repeatedly apply the the operator  $\mathcal{D}$  on  $W$ , i.e., letting  $\mathcal{D}^n W = \mathcal{D}(\mathcal{D}^{n-1}W)$  for all  $n \geq 1$ , we can obtain a sequence of subsolutions  $\{\mathcal{D}^n W, n \geq 1\}$ .

Now we are ready to present Theorem 3.3, one of the main results of the paper. In it, we show that the above dual value sequence increasingly converges to the true cost-to-go function of the primal problem (3).

**Theorem 3.3** *(i) The subsolution sequence  $\{\mathcal{D}^n W, n \geq 1\}$  is increasing in  $n$  in the sense that  $(\mathcal{D}^{n+1}W)_t(x) \geq (\mathcal{D}^n W)_t(x)$  for all  $n \geq 1$ ,  $0 \leq t \leq T$ , and  $x \in \mathbb{R}^n$ ;  
(ii) if, for some  $n$ ,  $(\mathcal{D}^{n+1}W)_t(x) = (\mathcal{D}^n W)_t(x)$  for all  $t$  and  $x$ , then  $\mathcal{D}^n W \equiv V$ ;  
(iii)  $\mathcal{D}^{T+1}W = V$ .*

Recall that any subsolution is dominated by the true cost-to-go function. Hence, one implication of Part (i) of Theorem 3.3 is that  $\mathcal{D}^n W \leq \mathcal{D}^{n+1}W \leq V$ . In other words, the subsolution sequence  $\{\mathcal{D}^n W\}$  iteratively improves its quality of approximation as lower bounds on  $V$ . Two key facts underpin the proof of Part (i). First, we need to show that, for any given subsolution, applying the dual operation on it will lead to a tighter lower bound on the true value function of the primal problem. It is worth noting that similar results have been established in the setups of optimal stopping problems (Chen and Glasserman (2007)) and infinite-horizon Markov decision processes (Desai, de Farias, and Moallemi (2013) and

Brown and Haugh (2017)). To prove Part (i), we manage to extend the fact to a finite-horizon framework. The second fact, Proposition 3.2, also plays an important role in the proof. It guarantees that  $\mathcal{D}^n W$ , as the output of the last dual iteration, is still a subsolution. So, implied by the first fact, we can further apply  $\mathcal{D}$  on it in the next iteration to yield more improvement. In other words, Proposition 3.2 accomplishes the inductive step for us to carry out induction on the sequence of  $\{\mathcal{D}^n W\}$  to show (i).

A more powerful conclusion stems from Parts (ii) and (iii) of the theorem. That is, the improvements in the sequence  $\{\mathcal{D}^n W, n \geq 1\}$  will terminate in a finite number of iterations and when it terminates, the optimal value of the primal problem has been achieved. From this, we propose the following DDP algorithm to solve the problem (3) in an iterative manner:

**Table I: A Duality Driven Dynamic Programming (DDP) Algorithm**

• **Step 0.** Initialization:

- **Step 0a.** Select an initial approximate value function sequence  $W^0 = (W_0^0, \dots, W_T^0)$ . One way to do it, for instance, is to use a feasible policy  $\alpha$  to compute its corresponding value

$$W_t^0(x) := \mathbb{E} \left[ \sum_{s=t}^{T-1} r_s(x_s, \alpha_s, \xi_s) + r_T(x_T) \middle| x_t = x \right]$$

for all  $x \in \mathbb{R}^n$  and  $0 \leq t \leq T - 1$ . Let  $\underline{V}^0 = W^0$ .

- **Step 0b.** Set  $n = 1$ .

• **Step 1.** Construct subsolutions using the dual operator  $\mathcal{D}$ :

- **Step 1a.** For  $\underline{V}^{n-1}$ , define a penalty function sequence  $z^n = (z_0^n, \dots, z_T^n)$  such that  $z_T^n(a, \xi) = 0$  and for any given  $0 \leq t \leq T - 1$ ,

$$z_t^n(a, \xi) = \sum_{s=t}^{T-1} \{ \mathbb{E}[r_s(x_s, a_s, \xi_s) + \underline{V}_{s+1}^{n-1}(f_s(x_s, a_s, \xi_s))] - (r_s(x_s, a_s, \xi_s) + \underline{V}_{s+1}^{n-1}(f_s(x_s, a_s, \xi_s))) \} \quad (12)$$

with  $a = (a_0, \dots, a_{T-1}) \in A$  and  $\xi = (\xi_0, \dots, \xi_{T-1})$ .

- **Step 1b.** For all state  $x$  and time  $t$ , determine the value of the following lower bound

$$\underline{V}_t^n(x) = \mathbb{E} \left[ \inf_{a \in A|t} \left( \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t^n(a, \xi) \right) \middle| x_t = x \right]. \quad (13)$$

- **Step 2.** If  $\underline{V}^n(x) \neq \underline{V}^{n-1}(x)$  for some  $x$ , let  $n = n + 1$  and go to Step 1.

Though the DDP algorithm focuses on updating the dual value, it can be used to improve control policies as well. For a suboptimal policy  $\alpha$ , we may run Step0a to evaluate it and initiate the algorithm with its policy value. Suppose that the algorithm terminates at the  $n$ th iteration. Replace the value function  $V_{t+1}$  in the one-step Bellman equation (6) with

$\underline{V}_{t+1}^n$  and solve

$$\alpha_t^n(x) = \arg \min_{a_t \in A_t} \mathbb{E} [r_t(x, a_t, \xi_t) + \underline{V}_{t+1}^n(f_t(x, a_t, \xi_t))] \quad (14)$$

for a new policy  $\alpha_t^n(\cdot)$  at time  $t$ ,  $0 \leq t \leq T-1$ . According to Part (ii) of Theorem 3.3, we should have achieved the optimality, i.e.,  $\underline{V}^n = V$ . The optimality of  $\alpha^n = (\alpha_0^n, \dots, \alpha_{T-1}^n)$  ensues.

### 3.2 An Illustration: Linear-Quadratic Control

Below we will use the classical LQC problem to demonstrate the effectiveness of policy improvement of the algorithm. In this case, the DDP algorithm can yield the optimal policy after just *two* iterations of the dual operation, no matter how long the time horizon of the problem is. By (13), the key steps in each dual iteration involve solving the inner optimization problem and determining the outer expectation value. One caveat is that, unlike the LQC example in which closed-form expressions for both are available, it is in general difficult to explicitly carry out these two types of computation, especially in high-dimensional problems. To address this issue, we shall explore in Section 4 how to resort to some numerical techniques, such as Monte Carlo simulation and the related approximation architectures, to implement the DDP algorithm effectively. One error bound is also developed therein (cf. Theorem 4.5) to deliver the performance guarantee. Theorem 3.3, despite its theoretical nature, still serves as an important cornerstone for us to obtain such numerical performance guarantees.

The LQC problem has received a lot of attention in control theory because of its tractability. It is widely applied in automatic control of a motion or a process to formulate how to regulate a system to stay close to the origin. The closed-form solution to the problem is well known in the literature. The intention of this subsection is definitely not to repeat these known results. Instead, we want to corroborate the result of the last subsection by showing its policy-improving effect. Following the standard setup of a LQC problem, consider a system whose dynamic equation is given by

$$x_{t+1} = D_t x_t + B_t a_t + \xi_t, \quad t = 0, \dots, T-1. \quad (15)$$

When it runs, it will incur a cost of

$$\sum_{t=0}^{T-1} (x_t^{tr} Q_t x_t + a_t^{tr} R_t a_t) + x_T^{tr} Q_T x_T. \quad (16)$$

In these expressions,  $D_t \in \mathbb{R}^{n \times n}$ ,  $B_t \in \mathbb{R}^{n \times m}$ ,  $Q_t \in \mathbb{R}^{n \times n}$ , and  $R_t \in \mathbb{R}^{m \times m}$ , are all given. The matrices  $Q_t$  are positive semidefinite symmetric and the matrices  $R_t$  are positive definite symmetric. There is no constraint on the controls  $a_t$ , i.e., we may take any vector in  $\mathbb{R}^m$  as its value. Each  $\xi_t$  has zero mean and a finite second moment. Assume that the decision maker has perfect information of the state  $x$  over the course of system evolution.

From the above description, it is not difficult to see that this problem is just a special case of (1-2) by taking a linear form for the evolution function  $f_t$  and a quadratic form for

the cost  $r_t$ . Its optimal policy is explicitly known in the literature (see, e.g., Sec. 4.1 in Bertsekas (1995)) to be of the following linear form:  $\alpha_t^*(x) = -L_t x$ , for  $t = 0, \dots, T - 1$ . Accordingly, the optimal cost function equals

$$V_t^*(x) = x^{tr} K_t x + \sum_{s=t}^{T-1} \mathbb{E} [\xi_s^{tr} K_{s+1} \xi_s]. \quad (17)$$

Here, both matrices  $L_t \in \mathbb{R}^{m \times n}$  and  $K_t \in \mathbb{R}^{n \times n}$  are explicitly computable. Detailed discussions are deferred to Electronic Companion B.

Applying the DDP algorithm to the LQC problem, we have

**Proposition 3.4** *Fix a matrix  $P_t \in \mathbb{R}^{m \times n}$  and a vector  $E_t \in \mathbb{R}^{m \times n}$  for each  $t$ . Consider a policy of the linear form*

$$\alpha_t(x) = P_t x + E_t, \quad 0 \leq t \leq T - 1. \quad (18)$$

*If we start the DDP algorithm with this policy, then it will terminate after two iterations at  $\underline{V}^2 \equiv V^*$ .*

Corroborating the results in Theorem 3.3, Proposition 3.4 shows that our DDP algorithm warrants the convergence to the true cost function of the LQC problem in two iterations. There are several studies in the literature related to the application of the information relaxation technique in LQC. Davis and Zervos (1995) postulate a linear form for the optimal penalty and thereby present a new proof of the LQC optimal control theorem based on the dual formulation. Haugh and Lim (2012) develop two types of approaches to constructing optimal penalties for an LQC problem. However, both of their constructions require some prior knowledge on the optimal value function of LQC. Compared with these studies, our DDP algorithm provides a more mechanical way to find the optimal penalty with little prior knowledge required.

The proof of Proposition 3.4 is contained in Appendix B. This example highlights one advantage of working with the duality-driven method in the computational aspect. That is, to compute the dual value, we just need to solve a deterministic optimization problem inside the expectation for which there is a vast research base that we can draw on for help. In particular, the proof of Proposition 3.4 shows that the minimization problem leading to the duality for the LQC problem turns out to be a quadratic program, which is well known to be tractable in the optimization literature (see, e.g., Nocedal and Wright (1999), Chapter 16).

## 4 Monte Carlo Implementation of the DDP Algorithm

As noted at the beginning of Section 3.2, the intrinsic difficulty of dealing with a general SDP problem lies in the fact that the inner optimization and the outer conditional expectation in (13) often cannot be analytically solved. Below we propose the use of regression to estimate the duality  $\underline{V}^n$  from simulated states for the purpose of implementing the DDP algorithm via Monte Carlo simulation. A related convergence analysis is developed in Section 4.2.

## 4.1 Regression-based Algorithm

In the first step of the algorithm, we need to generate a group of states on which the value of

$$\inf_{a \in A|t} \left( \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t^n(a, \xi) \right) \quad (19)$$

will be estimated so that we can use regression to build up the approximation to the conditional expectation in (13). Many of the regression-based methods in the literature on American option pricing (see, e.g., Carrière (1996), Longstaff and Schwartz (2001), Tsitsiklis and Van Roy (1999, 2001)) directly invoke the dynamic of the underlying asset to simulate states for continuation value estimation. Note that the exercising decision for an American option has no impact on the underlying price dynamics. One additional layer of complexity encountered here in a general SDP problem is that its state evolution hinges on the policy that we are using. As illustrated by the example in Appendix D.2, using a suboptimal policy of the original problem (1) to generate the states that our DDP algorithm will visit later can possibly lead to being stuck in suboptimality, because with this policy the algorithm may have no chance to access such states that contain useful information for us to improve the estimation.

To avoid this exploration pitfall, we suggest that the sequence of probability density functions  $\{G_1, \dots, G_T\}$  that are utilized for the purpose of state selection should be independent of the current policy of the SDP problem. In particular, if the support sets of all the  $G$ 's contain the entire state space of the problem, these sampling distributions enable us to reach any states in the space with nonzero chance. Imposing this ergodic requirement on  $G$  as one of the sufficient conditions, we investigate in Theorem 4.5 the asymptotic properties of the regression-based implementation of the DDP algorithm. Denote the number of simulated representative states by  $L$  hereafter. We independently draw  $L$  states,  $(x_t^{(1)}, \dots, x_t^{(L)})$ , from the distribution  $G_t$  at each time period  $t$ ,  $t = 1, \dots, T$ , where the superscript  $(l)$ ,  $1 \leq l \leq L$ , indicates the  $l$ th sample at time  $t$ . The values of the dual functions will be estimated on these points.

We now turn to present the core step of the implementation, i.e., how to use Monte Carlo regression to obtain an approximation to  $\underline{V}^n$  from the previous estimate of  $\underline{V}^{n-1}$  (cf. Step 1 in Table I). Let  $\{\psi_1, \dots, \psi_M\}$  denote a pre-specified set of basis functions, where each argument  $\psi_m$  is a function mapping from  $\mathbb{R}^n$  to  $\mathbb{R}$ . Assume that the previous iteration has yielded that  $\underline{V}^{n-1}$  can be approximated by

$$\underline{V}_t^{n-1}(x) \approx \widehat{\mathfrak{Y}}_t^{n-1}(x) := \sum_{m=1}^M \widehat{\beta}_{t,m}^{n-1} \psi_m(x) \quad (20)$$

for some constants  $\widehat{\beta}_{t,m}^{n-1}$ ,  $1 \leq m \leq M$  and  $1 \leq t \leq T$ . Following Step 1a in DDP, to construct a new penalty for the next round, we substitute the right hand side of (20) into (12). We then have the following approximation to  $z_t^n(a, \xi)$ :

$$\begin{aligned} & \mathfrak{z}_t^n(a, \xi) \\ &= \sum_{s=t}^{T-1} \left\{ \mathbb{E} \left[ r_s(x_s, a_s, \xi_s) + \sum_{m=1}^M \widehat{\beta}_{s+1,m}^{n-1} \psi_m(f_s(x_s, a_s, \xi_s)) \right] - \left( r_s(x_s, a_s, \xi_s) + \sum_{m=1}^M \widehat{\beta}_{s+1,m}^{n-1} \psi_m(f_s(x_s, a_s, \xi_s)) \right) \right\} \end{aligned}$$

for any  $a$  and  $\xi$ , where the expectation in the first term of  $\mathfrak{z}_t^n$  is taken with respect to  $\xi_s$ .

In evaluating  $\mathfrak{z}_t^n$ , we need to compute  $\mathbb{E}[\psi_m(f_s(x_s, a_s, \xi_s))]$ . For many applications, especially when  $\psi_m$  is a polynomial,  $f_s$  is simple, and the distribution of  $\xi_s$  is analytically known, we can explicitly compute this expectation. For the cases in which its closed-form expression is not available, we may rely on Monte Carlo simulation to generate samples from the distribution of  $\xi_s$  and then use sample averages to approximately evaluate it. To expedite the computation in this step, we also attempt an alternative simulation method, which is the low-discrepancy method from the quasi-Monte Carlo (QMC) literature, in the numerical experiments of the next section. Different from plain Monte Carlo, this QMC approach deterministically chooses representative points for  $\xi$ . We find that QMC can deliver excellent approximation performance with a relatively smaller number of simulation trials, consistent with the well known fact that the QMC converges faster than the ordinary Monte Carlo. One may refer to Chapter 5 of Glasserman (2004) for a comprehensive coverage of this subject.

Once the value of  $\mathfrak{z}_t^n(a, \xi)$  is determined, we proceed to build up the regression estimators for the conditional expectation (13) in Step 1b of the DDP algorithm. To this end, we posit that (13) can be represented as a linear combination of the basis functions, i.e.,

$$\mathbb{E}[J_{t,n}(\xi|t, x_t)] := \mathbb{E}\left[\inf_{a \in A|t} \left(\sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t^n(a, \xi)\right) \middle| x_t = x\right] = \sum_{m=1}^M \beta_{t,m}^n \psi_m(x), \quad (21)$$

at  $x_t = x$  for any given  $t$ ,  $t = 0, \dots, T-1$ . Here,  $J_{t,n}(\xi|t, x_t)$  is a shorthand notation for the minimization problem inside the expectation, whose value apparently depends on the tail vector of random perturbation  $\xi|t = (\xi_t, \dots, \xi_{T-1})$  and the system state  $x_t$  at time  $t$ . The standard least square arguments imply that the coefficient vector  $\beta_t^n = (\beta_{t,1}^n, \dots, \beta_{t,M}^n)^{tr}$  in (21) should be given by

$$\beta_t^n = (B_{\psi\psi}^t)^{-1} B_{J\psi}^{t,n} := (\mathbb{E}^G[\Psi_M(X_t)\Psi_M(X_t)^{tr}])^{-1} \mathbb{E}^{G \otimes \xi}[\Psi_M(X_t)J_{t,n}(\xi|t, X_t)]. \quad (22)$$

In (22),  $B_{\psi\psi}^t$  is the indicated  $M \times M$  matrix  $\mathbb{E}^G[\Psi_M(X_t)\Psi_M(X_t)^{tr}]$  (assumed nonsingular) with  $\Psi_M(x) = (\psi_1(x), \dots, \psi_M(x))^{tr}$ . The superscript  $G$  stresses that the expectation is defined on  $G_t$ , the distribution of  $X_t$ . Meanwhile,  $B_{J\psi}^{t,n}$  is the indicated vector of dimension  $M$  computed from  $\mathbb{E}^{G \otimes \xi}[\Psi(X_t)\mathfrak{J}_{t,n}(\xi|t, X_t)]$  with  $X_t \sim G_t$  and  $\xi|t$  independently drawn from its own distribution.

Both  $B_{\psi\psi}^t$  and  $B_{J\psi}^{t,n}$  can be estimated on the basis of observations of pairs  $(\xi|t, X_t)$ . More explicitly, starting from each point  $x_t^{(l)}$ , we independently simulate one path of  $\xi^{(l)}|t = (\xi_t^{(l),t}, \xi_{t+1}^{(l),t}, \dots, \xi_{T-1}^{(l),t})$  from the distribution of  $\xi$ . Suppose for a moment that the value of  $J_{t,n}(\xi|t, X_t)$  can be (approximately) computed at each pair  $(\xi^{(l)}|t, x_t^{(l)})$  and denote that quantity by  $\mathfrak{J}_{t,n}^{(l)}$ . Let  $\hat{B}_{\psi\psi}^t$  be an  $M \times M$  matrix with the  $(i, j)$ -entry

$$\frac{1}{L} \sum_{l=1}^L \psi_i(x_t^{(l)}) \psi_j(x_t^{(l)}) \quad (23)$$

and  $\hat{B}_{\mathfrak{J}\psi}^{t,n}$  be an  $M$ -vector with the  $k$ th entry

$$\frac{1}{L} \sum_{l=1}^L \mathfrak{J}_{t,n}^{(l)} \psi_k(x_t^{(l)}). \quad (24)$$

Then, an estimate to  $\beta_t^n$  can be formed by  $\hat{\beta}_t^n = (\hat{B}_{\psi\psi}^t)^{-1} \hat{B}_{\mathfrak{J}\psi}^{t,n}$ . From it, we complete one iteration in our DDP algorithm by building up a new approximate to the dual value function

$$\underline{V}_t^n(x) \approx \hat{\mathfrak{Y}}_t^n(x) := \sum_{m=1}^M \hat{\beta}_{t,m}^n \psi_m(x),$$

where  $\hat{\beta}_{t,m}^n$  is the  $m$ th entry of  $\hat{\beta}_t^n$ .

To determine the value of  $\mathfrak{J}_{t,n}^{(l)}$ , we replace  $z_t^n(a, \xi)$  by its approximation  $\mathfrak{z}_t^n(a, \xi)$  in  $J_{t,n}(\xi|t, X_t)$  and solve the following optimization problem:

$$\mathfrak{J}_{t,n}^{(l)} := \inf_{a \in A|t} \left( \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s^{(l),t}) + r_T(x_T) + \mathfrak{z}_t^n(a, \xi^{(l)}|t) \right) \quad (25)$$

subject to the constraints  $x_t = x$  and

$$x_s = f_{s-1}(x_{s-1}, a_{s-1}, \xi_{s-1}^{(l),t}) \quad (26)$$

for all  $s = t + 1, \dots, T$ . The outcome of the above optimization problem, denoted by  $\mathfrak{J}_{t,n}^{(l)}$ , will be used as one observation of  $J_{t,n}(\xi|t, X_t)$  at  $(\xi^{(l)}|t, x_t^{(l)})$  to estimate  $B_{J\psi}^{t,n}$ .

It is worth mentioning that, given  $\xi^{(l)}|t = (\xi_t^{(l),t}, \xi_{t+1}^{(l),t}, \dots, \xi_{T-1}^{(l),t})$ , the problem (25-26) is indeed a deterministic optimization program. Compared with many of the SDP algorithms in which stochastic optimization is involved, the computation for the solution to (25-26) is less demanding. Similar to the case of LQC problems, the vast research literature on deterministic optimization provides us various flexible and potent methodologies that we can draw on to solve it. In particular, we develop in the next section an efficient numerical scheme based on the DC programming to solve this inner optimization problem for a broad class of control problems. Simplifying the underlying probabilistic structure of an SDP problem to yield some computational advantages is a commonly used strategy in approximate dynamic programming. For instance, the approach of certainty equivalent control replaces the stochastic disturbances with deterministic quantities so as to reduce the SDP problem to a deterministic one; see, e.g. Chapter 2.3 of Bertsekas (2019). Such simplification arises naturally in the duality formulation.

We encapsulate the implementation procedure discussed above in Table III in Appendix D.1. As noted in the introduction, another computational advantage of the algorithm is that we can deploy parallel computing to expedite it. Note that, for different representative state  $x^{(l)}$ ,  $1 \leq l \leq L$ , simulation of the associated  $\xi^{(l)}|t$  and the subsequent inner optimization in Step 1b of Table III are independent. It is easy to parallelize the execution of these procedures at different  $x^{(l)}$  using multiprocessor machines. Finally, with the help of the approximate lower bound  $\hat{\mathfrak{Y}}^n$  obtained from our regression-based algorithm, we can also build up a confidence interval estimate, which many approximate dynamic programming methods are short of, for the true cost-to-go value of the original problem. One may refer to the discussion around Table IV in D.1 and the numerical examples in the next section for details in this regard.

## 4.2 Convergence Analysis

Theorem 3.3 establishes that in principle the DDP method should lead a convergence to the true value of the SDP problem in finite rounds of iterations. In contrast, its regression-based implementation, as discussed in Section 4.1, is apparently subject to the biases coming from three sources. First, the functional approximations built upon the basis functions may be biased relative to the true dual function  $\underline{V}^n$ . Second, the states  $\{x^{(l)}, 1 \leq l \leq L\}$  simulated at the beginning of algorithm execution may not be sufficiently representative. Third, the solver of the optimization problem (25-26) may only be able to find its local optimal solution. However, in comparison with the first two errors, the error that arises in solving the deterministic optimization problem is typically not significant if a proper optimizer is used, as suggested by the numerical examples in Section 5. Hence, we focus only on the characterization of how the performance of the DDP algorithm will be affected by those factors in Theorem 4.5.

Without loss of generality, let us assume that the state space  $\mathcal{X}$  of the original problem is compact in the subsequent convergence analysis. Many numerical examples, including the ones in Section 5.2, satisfy this assumption. In addition, for those cases with unbounded state spaces, we can obtain approximations with sufficient accuracy by truncating the spaces into compact ones; see, for instance, Altman (1999), Kushner and Dupuis (2001), Dufour and Prieto-Rumeau (2012), and Saldi, Linder and Yuksel (2018) for more discussions in that direction.

Consider an infinite series of basis functions  $\{\psi_m(x), m \geq 1\}$ . Suppose that we take the first  $M$  functions from this set to form a functional vector  $\Psi_M(x) = (\psi_1(x), \dots, \psi_M(x))^{tr}$  to perform the DDP algorithm. We intend to characterize how its outcome will converge to the true value as we increase both the number of representative states  $L$  and the number of the basis functions  $M$ . We need several other technical assumptions to proceed. First,

**Assumption 4.1** *There exists a measure  $F$ , whose support is  $\mathcal{X}$ , such that the basis function sequence  $\{\psi_m(x), m \geq 1\}$  is orthonormal under this measure  $F$ ; that is to say,*

$$\int_{\mathbb{R}^n} \psi_i(x)\psi_j(x)dF(x) = \begin{cases} 0 & i \neq j, \\ 1 & i = j. \end{cases}$$

Note that this assumption is not restrictive at all because we may perform the celebrated Gram-Schmidt orthogonalization to construct an orthogonal basis from any given set of linearly independent functions.

The second assumption is about the distributions  $\{G_t, 1 \leq t \leq T\}$  that are used for sampling representative states.

**Assumption 4.2** *Each of the sampling distributions  $G_t(x)$  is absolutely continuous with respect to the measure  $F$  in Assumption 4.1. Furthermore, the Radon-Nykodym derivative between these two measures  $dG_t/dF(x)$  is bounded away from zero and infinity on  $\mathcal{X}$ . In other words, there exist strict positive constants  $\epsilon$  and  $D$  such that  $\epsilon < dG_t/dF(x) < D$  for all  $x \in \mathcal{X}$ .*

Essentially the purpose of Assumption 4.2 is to help us avoid the aforementioned exploration pitfall (cf. Appendix D.2). The positiveness of  $dG/dF$  over the entire state space  $\mathcal{X}$  ensures



that the state samplers introduced in the algorithm have non-zero probability to access any part of  $\mathcal{X}$ .

Finally, we assume

**Assumption 4.3** *There exists a constant  $C$  such that, for any positive integer  $M$ , a functional vector consisting of the first  $M$  basis functions in the set,  $\Psi_M(x) = (\psi_1(x), \dots, \psi_M(x))^{tr}$ , satisfies*

$$\sup_{x \in \mathcal{X}} \left( \sum_{m=1}^M \psi_m^2(x) \right)^{1/2} \leq CM \quad \text{and} \quad \sup_{1 \leq m \leq M} \mathbb{E}^G[\psi_m^2(X)] \leq C,$$

and

**Assumption 4.4** *The optimal cost-to-go function  $\{V_t(x)\}_{0 \leq t \leq T}$  is bounded on the compact set  $\mathcal{X}$ .*

Indeed, one can show that Assumption 4.3 holds for many popular series used in the literature on the approximation theory, including Fourier series, spline series, and local polynomial partition series; see Belloni et al. (2015) for a detailed discussion. The boundedness of the value function  $V$  in Assumption 4.4 is natural if we can establish its continuity. Hernández-Lerma and Lasserre (1997) suggest some technical conditions under which a general SDP problem has continuous value functions.

Now we turn to present the main asymptotic result for our regression-based algorithm. Let

$$\Delta_M := \max_{0 \leq t \leq T} \inf_{\gamma_t = (\gamma_t^1, \dots, \gamma_t^M) \in \mathbb{R}^M} \|V_t - \Psi_M^{tr} \gamma_t\|_\infty,$$

where  $\|\cdot\|_\infty$  is the  $L_\infty$  norm such that  $\|f\|_\infty = \sup_{x \in \mathcal{X}} |f(x)|$ . The quantity  $\Delta_M$  measures the least error magnitude that we can achieve if we approximate the true value function of the SDP problem by linearly combining the  $M$  basis functions. Recall that, absent both the simulation and the approximation errors, the dual value sequence from the DDP method should converge to the true value in at most  $T + 1$  iterations as shown in Theorem 3.3. Correspondingly, we develop an upper bound on the bias of  $\widehat{\mathfrak{V}}^{T+1}$ , the approximate dual value after  $T + 1$  rounds of iterations of the regression-based algorithm, in the next theorem.

**Theorem 4.5** *Suppose that Assumptions 4.1 to 4.4 hold. Then, there exists a constant  $C$ , independent of  $L$  and  $M$ , such that*

$$\mathbb{E} \left[ \left| \widehat{\mathfrak{V}}_0^{T+1}(x) - V_0(x) \right| \right] \leq \left( 1 + 2l_M + C \left( \frac{M^6}{L} \right) \right)^T \left[ (1 + l_M) \Delta_M + C \left( \frac{M^6}{L} \right)^{1/4} \right], \quad (27)$$

where  $l_M$  is the corresponding Lebesgue constant of the basis functions  $\{\psi_m(x), m \geq 1\}$  (cf. Definition D.1 in Appendix D.3)

Theorem 4.5 clearly shows how the algorithm accuracy is determined by the choice of basis functions and the amount of simulation effort. Note that both  $\Delta_M$  and  $l_M$  are the characteristics of the basis functions that we choose. In Remarks 4.6 and 4.7 below, we present

the corresponding orders of  $\Delta_M$  and  $l_M$  with respect to  $M$  under a variety of commonly used basis functions. For instance,  $l_M$  will be bounded by a constant and  $\Delta_M$  decays in a power order of  $M$  for some choices of basis functions. Once the set of basis functions is chosen and  $M$  is fixed, we need to pick up a sufficiently large  $L$  to control the right-hand side of (27). Theorem 4.5 spells out explicitly that  $L$  should grow faster than  $O(M^6)$  in order to keep such error in check. It is well known in regression analysis that a model can be overfitted if the amount of observed data is insufficient relative to the number of regressors. When  $L$ , the number of simulated states on which we estimate the dual values, is not adequate in our DDP algorithm, this overfitting effect will cause a divergence for the DDP algorithm, as illustrated by the numerical examples in Section 5. From the above discussion, we can see that Theorem 4.5 can help us understand the asymptotic behavior of our DDP estimator when both  $L$  and  $M$  tends to infinity for a given basis function set and thereby provide us valuable guidances on the choice of basis functions and such parameters as  $L$  and  $M$ . The numerical examples in Section 5 also show that the relative ratio between  $L$  and  $M$  for the DDP algorithm to converge could be lower under some specific cases. We leave the investigation on tighter error bounds to the future work.

**Remark 4.6** *The approximation theory has produced some bounds on the Lebesgue constant  $l_M$  for a variety of basis function sets. Suppose that the density function of  $G$  on  $\mathcal{X}$  is bounded away from zero and infinity. We can show that  $l_M$  should be bounded by a constant  $C$  for spline series, wavelet series and local polynomial partition series, and  $l_M \leq C \log(M)$  for Chebyshev polynomial series and Fourier series. See, e.g., Zygmund (2002), Huang (2003), Belloni et al. (2015), and Chen and Christensen (2015).*

**Remark 4.7** *As for  $\Delta_M$ , some studies show that, if the true value function is  $s$  times continuously differentiable, the approximation error of the spline or polynomial regressors is bounded by*

$$\Delta_M \leq M^{-\kappa},$$

*where  $\kappa = s/d$  and  $d$  stands for the dimensionality of the function. For the proofs of this property, one may refer to Section 7.6 of DeVore and Lorentz (1993), Section 5.3.2 of Timan (1963), and Theorem 12.8 of Schumaker (1981).*

## 5 Numerical Experiments

In this section we shall apply the regression-based Monte Carlo DDP algorithm to solve two problems related to order execution and inventory management. Both are well known to be intractable in the literature and only approximate methods are available so far. Our algorithm demonstrates great potential in effectively assessing and improving these heuristic policies towards optimality.

### 5.1 Optimal Order Execution in the Presence of Market Frictions

The first numerical example we consider in the paper is an optimal order execution problem in which a trader plans to transact a large block of equity over a fixed time framework with

minimum impact costs. It can be viewed as a variant of the models proposed in Bertsimas and Lo (1998), Almgren and Chriss (2000), and Haugh and Wang (2014). Assume that there are  $n$  different assets traded in the market, and the trader aims to acquire  $\bar{\mathbf{R}} = [\bar{R}_1, \dots, \bar{R}_n]^{tr}$  shares in each of the assets in  $T$  periods. The objective of the trader is to determine a trading schedule, i.e., how many shares to purchase in each period, denoted by  $\{\mathbf{S}_1, \dots, \mathbf{S}_T\}$ ,  $\mathbf{S}_t \geq 0$ ,  $t = 1, 2, \dots, T$ , to minimize the associated transaction cost. Let  $\mathbf{R}_t \in \mathbb{R}^n$  denote the number of shares in each asset short of the target  $\bar{\mathbf{R}}$  at time  $t$ . Then, a feasible trading schedule should satisfy

$$\sum_{t=1}^T \mathbf{S}_t = \bar{\mathbf{R}}, \quad \mathbf{S}_t \geq 0, \quad \mathbf{S}_t \in \mathbb{R}^n, \quad (28)$$

$$\mathbf{R}_{t+1} = \mathbf{R}_t - \mathbf{S}_t, \quad \mathbf{R}_1 = \bar{\mathbf{R}}, \quad \text{for all } t = 1, \dots, T. \quad (29)$$

To complete the statement of the problem, we must specify the price dynamics. In particular, we use  $\tilde{\mathbf{P}}_t \in \mathbb{R}^n$  and  $\mathbf{P}_t \in \mathbb{R}^n$  to represent the fundamental values and actual transaction prices of all assets at time  $t$ , respectively, and assume that  $\tilde{\mathbf{P}}_t$  and  $\mathbf{P}_t$  follow the evolution laws such that

$$\tilde{\mathbf{P}}_t = \tilde{\mathbf{P}}_{t-1} + \mathbf{A}\mathbf{S}_t + \mathbf{B}\mathbf{X}_t + \boldsymbol{\epsilon}_t, \quad (30)$$

$$\mathbf{P}_t = \tilde{\mathbf{P}}_t + h(\mathbf{S}_t), \quad (31)$$

for all  $t$ , where  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is a positive definite matrix and  $\mathbf{B} \in \mathbb{R}^{n \times m}$ . Here  $\{\boldsymbol{\epsilon}_t, t = 1, \dots, T\}$  is a sequence of white noises with mean zero and covariance matrices  $\Sigma_{\boldsymbol{\epsilon}}$ . As shown in (30-31), our model incorporates both permanent and temporary price impacts of transaction activities. In (30), the constant matrix  $\mathbf{A}$  is used to capture the intensity of the permanent impact: trading the amount of  $\mathbf{S}_t$  changes the assets' fundamental values by  $\mathbf{A}\mathbf{S}_t$  and this change will last persistently in the future via the iterative relation of  $\tilde{\mathbf{P}}$ . Note that this permanent price impact takes a linear form, which is a commonly adopted modeling assumption in the literature; see Bertsimas and Lo (1998), Almgren and Chriss (2000), Huberman and Stanzl (2005), and Haugh and Wang (2014), for example. Huberman and Stanzl (2004) and Gatheral (2010) argue that including a nonlinear permanent price impact will introduce the possibility of arbitrage.

On the other hand, we introduce the function  $h(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  in (31) to reflect the trading-caused impacts that will not last into the next period. The literature documents that this kind of temporary impact in the real-life market should be concave in trading quantities (cf. Bouchaud, Farmer, and Lillo (2009)). However, an assumption of concavity often makes the control problem intractable. To demonstrate that our algorithm still works well when analytical solutions are unavailable, we assume  $h(\mathbf{S}_t) = \mathbf{D}\sqrt{\mathbf{S}_t}$  in the experiment, where  $\mathbf{D}$  is a constant coefficient matrix.

In addition, our model allows the trader to incorporate some predictive "signals" to extract information about the stock's future movements for improving the performance of her trade execution. The auxiliary process  $\mathbf{X}_t \in \mathbb{R}^m$  in (30) serves this purpose. There are several possibilities proposed in the literature for the choice of such signals. For instance, Bertsimas and Lo (1998) suggest that  $\mathbf{X}$  could be the return of a broader market index such as S&P500, a factor commonly used in traditional asset pricing models such as CAPM, or

the outputs of an “alpha” model from the trader’s private stock-specific analysis that is not yet impounded into market prices. In the following experiment, we abstract out the true meaning of  $\mathbf{X}$  and assume it to follow a stationary AR(1):

$$\mathbf{X}_t = \mathbf{C}\mathbf{X}_{t-1} + \boldsymbol{\eta}_t, \quad (32)$$

where  $\mathbf{C} \in \mathbb{R}^{m \times m}$  is a matrix with all of its eigenvalues less than unity in modulus, which determines the “decay” speed of the information, and the random noises  $\boldsymbol{\eta}_t \sim N(0, \Sigma_\eta)$  are Gaussian white, independent of  $\boldsymbol{\epsilon}_t$ . It is worthwhile to point out that the particular form of (32) is not essential for our algorithm to work. We have tried some other specifications in the experiments for  $\mathbf{X}_t$  and found that does not affect the effectiveness of the method. Gârleanu and Pedersen (2013, 2016) use the same dynamic to model the return-predicting factor in the investigation of portfolio policy when trading is costly and security returns are predictable by signals.

As aforementioned, the trader’s problem is to minimize

$$\min_{\{\mathbf{S}_t, 1 \leq t \leq T\}} \mathbb{E} \left[ \sum_{t=1}^T \mathbf{P}_t^{tr} \mathbf{S}_t \right], \quad (33)$$

where  $\mathbf{P}_t^{tr} \mathbf{S}_t$  is how much the trader actually pays in period  $t$ . In E, we show that this objective is indeed equivalent to

$$\min_{\{\mathbf{S}_t, 1 \leq t \leq T\}} \mathbb{E} \left[ \sum_{t=1}^T \mathbf{S}_t^{tr} h(\mathbf{S}_t) + \sum_{t=0}^{T-1} (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1} \right]. \quad (34)$$

The new representation (34) clearly points out two sources that are contributing to the ultimate transaction costs of the trader. The first term corresponds to the temporary impact cost that the trader needs to pay in the process of purchasing  $\tilde{\mathbf{R}}$  shares of assets due to the presence of  $h(\mathbf{S})$ . The second term consists of the changes in the fundamental value of the assets because of the permanent price impact that her trading activities will generate. It is easy to see that the above SDP problem has a convex objective function. Hence, the optimal policy of the problem uniquely exists. It should be a function of state variables  $\mathbf{X}$  and  $\mathbf{R}$ .

Next we use the DDP algorithm to solve the minimization problem (34) with the constraints (28-32). The nonnegative constraint  $\mathbf{S}_t \geq 0$  turns out to be the most difficult one to deal with. As suggested by Bertsimas and Lo (1998), imposing it will introduce a partition structure to the optimal policy, and more seriously, the number of partitioned regions increases combinatorially with the time horizon  $T$ . This renders solving the problem through the Bellman equation computationally infeasible; see also Bemporad et al. (2002) for more discussion on this issue in the context of a general constrained linear quadratic system. Aiming at some applications in market microstructure, Chen, Kou, and Wang (2018) develop a partitioning algorithm for linear-quadratic Markov decision processes with linear inequality constraints. Their method recursively constructs polyhedral regions in which the optimal value function and policy have analytical quadratic and linear forms, respectively. Note that the complexity of their method is still exponential in  $T$  (cf. Notes 5 and 6 of their paper). Moreover, it cannot be applied here because the existence of the concave temporary impact  $h(\cdot)$  makes our model no longer a linear-quadratic problem.

A variety of heuristic approaches can help us derive approximate solutions to this problem. The following numerical experiments show that our DDP algorithm can be used not only for evaluating the performance but, more importantly, to effectively improve them. Below is a summary of the heuristics that we consider in this paper.

- From a tractable simplification of the problem. It is straightforward to see that, if we ignore the no-sales constraint  $\mathbf{S}_t \geq 0$  and the temporary price impact  $h(\mathbf{S}_t)$ , the problem (34) with the constraints (30-32) indeed degenerates to a standard LQC. The computation in E shows that the optimal policy of this simplified problem is analytically known:

$$\tilde{\mathbf{S}}_t(\mathbf{X}_t, \mathbf{R}_t) = \left( \mathbf{I} - \frac{1}{2} \mathbf{Q}_{t+1}^{-1} \mathbf{A}^{tr} \right) \mathbf{R}_t + \left( \frac{1}{2} \mathbf{Q}_{t+1}^{-1} \mathbf{K}_{t+1} \mathbf{C} \right) \mathbf{X}_t, \quad (35)$$

where  $\mathbf{Q}_t \in \mathbb{R}^{n \times n}$  and  $\mathbf{K}_t \in \mathbb{R}^{n \times m}$  are two matrices that can be determined by the matrix equations in (98-101). This linear policy emphasizes the importance of trading on signals in the process of meeting the execution target, as the current information level  $\mathbf{X}_t$  affects the amount of trading volume  $\tilde{\mathbf{S}}_t$ . However, such  $\tilde{\mathbf{S}}_t$  is not feasible to the original problem because it may take negative values when  $\mathbf{X}_t$  is negative. To restore a feasible policy, we may project  $\tilde{\mathbf{S}}_t$  into the region  $[0, \bar{\mathbf{R}}]$  by letting

$$\mathbf{S}_t^{LQ}(\mathbf{X}_t, \mathbf{R}_t) = \min \left( \max \left( \tilde{\mathbf{S}}_t(\mathbf{X}_t, \mathbf{R}_t), 0 \right), \mathbf{R}_t \right). \quad (36)$$

- From linear program approximation. Introduced by Schweitzer and Seidmann (1985) and further developed by de Farias and Van Roy (2003, 2004) and Desai, de Farias, and Moallemi (2012a, 2013), the linear programming based approach provides us an attractive way to construct approximate solutions to the dynamic programs. Consider a collection of basis functions  $\{\psi_1, \dots, \psi_K\}$  and use the following regression to approximate the optimal value functions at every time  $t = 1, \dots, T$ :

$$V_t(\mathbf{X}_t, \mathbf{R}_t) \approx \sum_{k=1}^K \theta_{k,t} \psi_k(\mathbf{X}_t, \mathbf{R}_t),$$

where  $\theta_t = (\theta_{1,t}, \dots, \theta_{K,t})$  is the regression coefficient to be determined. As noted in the discussion around Definition 3.1, the true value function of an SDP must be the largest subsolution. Select some representative states  $\{(\mathbf{X}^i, \mathbf{R}^i) : i = 1, \dots, I\}$ . Let  $c_{t,i}$  be a positive constant for all  $t = 1, \dots, T$  and  $i = 1, \dots, I$ . We may recast this fact as a linear program for the problem (34):

$$\max_{\theta_t: t=1, \dots, T} \sum_{i=1}^I c_{t,i} \sum_{k=1}^K \theta_{k,t} \psi_k(\mathbf{X}^i, \mathbf{R}^i)$$

subject to

$$\sum_{k=1}^K \theta_{k,t} \psi_k(\mathbf{X}^i, \mathbf{R}^i) \leq \min_{\mathbf{S}_i \geq 0} \mathbb{E} \left[ \mathbf{S}_i^tr h(\mathbf{S}_i) + (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1} + \sum_{k=1}^K \theta_{k,t+1} \psi_k(\mathbf{X}_{t+1}, \mathbf{R}_{t+1}) \mid (\mathbf{X}_t, \mathbf{R}_t) = (\mathbf{X}^i, \mathbf{R}^i) \right]. \quad (37)$$

Note that the constraint (37) is just a rephrasing of Definition 3.1 and it is a linear inequality with respect to the regression coefficient  $\theta_t$ .

- Lookahead. One-step and multistep lookahead constitute another class of commonly used approaches to produce approximate solutions to dynamic programs. We replace  $V_{t+1}$  in the Bellman equation (6) with any of its approximation  $\tilde{V}_{t+1}$ . Then, the minimization

$$\mathbf{S}_t^{LO}(\mathbf{X}_t, \mathbf{R}_t) = \arg \min_{\mathbf{S}_t \geq 0} \mathbb{E} \left[ \mathbf{S}_t^{tr} h(\mathbf{S}_t) + (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1} + \tilde{V}_{t+1}(\mathbf{X}_{t+1}, \mathbf{R}_{t+1}) \middle| (\mathbf{X}_t, \mathbf{R}_t) \right] \quad (38)$$

defines the one-step lookahead policy at state  $(\mathbf{X}_t, \mathbf{R}_t)$ . For the purpose of illustration, we make use of the value function of the simplified problem discussed in the first bullet as  $\tilde{V}$ . To derive the multistep lookahead, we can minimize the cost of the first  $L > 1$  steps with the future cost approximated by a function  $\tilde{V}_{t+L}$ .

- Backward dynamic programming. To overcome the curse of dimensionality encountered in utilizing the equations (5-6), one may use the basis functions to obtain low-dimensional regression representations of the value functions and repeatedly substitute them into the one-step Bellman's equation (6) to produce approximate solutions to the problem in a backward fashion. The regression coefficients can be estimated by using the least square method on some representative states that are fixed beforehand.

Starting from any of these heuristics, our DDP algorithm demonstrates a strong ability to construct improved approximations for all of them. Table 1 displays the related convergence results. Here we consider a case with three assets and a signal vector of two variables, i.e.  $\mathbf{R} = [R_1, R_2, R_3]^{tr}$  and  $\mathbf{X} = [X_1, X_2]^{tr}$ . To deal with this 5-dim problem, we use the following set of basis functions in the experiment:

$$\left\{ 1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}, (X_i X_j)_{1 \leq i, j \leq 2}, (R_k R_l)_{1 \leq k, l \leq 3}, (R_k X_j)_{1 \leq k \leq 3, 1 \leq j \leq 2}, \right. \\ \left. (R_k \sqrt{R_k})_{1 \leq k \leq 3}, (X_i^3)_{1 \leq i \leq m}, (R_i^3)_{1 \leq i \leq n}, (X_i^4)_{1 \leq i \leq m}, (R_i^4)_{1 \leq i \leq n} \right\}. \quad (39)$$

The abbreviation  $(X_i)_{i=1,2}$ , for example, represents that both functions  $X_1$  and  $X_2$  are included. The other notations should be understood in the same way. We also include a constant, represented by 1 in the set (39), in the regressors. In the interest of space, the values of all the model parameters are reported in Appendix E.

As noted in Section 4, we need a state selector  $G$  to generate a number of representative pairs of  $(\mathbf{X}, \mathbf{R})$  in the state space at each period  $t$  so that we can run regressions to extrapolate the dual values observed on these pairs. Note that the signal process  $\mathbf{X}_t$  has an autonomous dynamic (32), independent of the control policies taken by the trader. We thereby use its marginal distribution in the experiment to simulate samples for  $\mathbf{X}$ . Meanwhile, since the sample trajectory of  $\mathbf{R}_t$  resides in  $[0, 10^5]^3$  under any trading scheme, we take the uniform distribution in this cube to sample  $\mathbf{R}$ . To speed up the overall calculation when evaluating the dual values, we parallelize the simulation of the state pairs  $(\mathbf{X}, \mathbf{R})$  and random noises  $(\boldsymbol{\epsilon}_t, \boldsymbol{\eta}_t, t = 1, \dots, 20)$  to multicore CPUs (32 cores in our experiments) and solve the corresponding optimization programs simultaneously.

Table 1 consists of four subparts. Each of them reports the respective convergence results for the four approximate heuristics. We first assess the performance of each approximate policy by evaluating its corresponding average transaction costs along  $K = 1 \times 10^4$  simulated

Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
LQ	0			325.49	(0.49)	18.88%
	1	264.05	(2.71)	-	-	-
	2	267.12	(0.42)	-	-	-
	3	269.26	(0.25)	272.82	(0.53)	1.28%
	4	269.33	(0.25)			
Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
Linear	0			433.28	(8.21)	40.38%
	1	258.32	(0.63)	-	-	-
	2	266.38	(0.62)	-	-	-
	3	268.76	(0.32)	-	-	-
	4	269.63	(0.26)	272.44	(0.51)	1.06%
	5	269.55	(0.25)			
Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
Lookahead	0			324.80	(0.49)	18.64%
	1	264.23	(2.59)	-	-	-
	2	266.99	(0.44)	-	-	-
	3	269.10	(0.25)	272.38	(0.50)	1.12%
	4	269.33	(0.25)			
Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
Backward	0			285.55	(0.50)	8.21%
	1	262.08	(2.80)	-	-	-
	2	269.53	(0.24)	272.70	(0.50)	1.28%
	3	269.56	(0.23)			

Table 1: The convergence results of the DDP algorithm in the example of order execution. The four approximation methods are used to construct the initial policies as the inputs to the DDP algorithm. We denote them by LQ, Linear, Lookahead, and Backward, respectively, in the table. We simulate  $K = 1 \times 10^4$  sample paths of random noises  $(\epsilon_t, \eta_t, t = 1, \dots, 20)$  to estimate their corresponding values, which are reported in the cell of Primal Values in Iteration 0 of every subparts of the table. The standard error of this policy estimation is shown in the column “(SE)”. The entry in Row “Iteration 1” and Column “Dual Values” displays the dual value associated with each approximate policy. We sample  $L = 1.5 \times 10^4$  pairs of  $(\mathbf{X}, \mathbf{R})$  in each time step from the distribution  $G$  mentioned in the body text to compute the dual values in each iteration. The same distribution  $G$  is also used to generate representative states for the methods of linear programming approximation and backward dynamic programming. In LP, we simulate 300 state pairs and thereby solve a linear program with 300 constraints. In Backward DP, we simulate  $1.5 \times 10^4$  states for carrying out the least square estimation. The numbers in the parentheses in the column next to “Dual Values” are the standard errors of the dual estimations. The percentage gaps in the last column of the table are computed according to the ratio of  $(\text{Primal} - \text{Dual})/\text{Primal}$ . The default parameters used in the experiments are  $\lambda = 10$  and  $\delta = 1$ . The values of other parameters are reported in E. All the computation experiments are conducted on a PC equipped with an Intel Xeon 32-core 2.93 GHz CPU and 12.0 GB of RAM. The computation environment is Windows 7 and MATLAB R2017a and parallel pool. The average computational time is 1416.1s per iteration.

paths of random noises ( $\epsilon_t, \eta_t, t = 1, \dots, 20$ ). The outcome is reported in the first row of each subpart. Meanwhile, we compute the dual value associated with each heuristic policy in the second row of the column “Dual Values”. All the approximations have significant duality gaps, which show that the performance of all the policies are not satisfactory.

Consistent with the theoretical convergence results in the last sections, the dual values increase as we run more iterations of the DDP algorithm, no matter which approximate heuristic we start with. These dual values thereby provide a sequence of increasingly tighter lower bounds to the true value of the problem. The algorithm terminates after several rounds of iterations when it produces no essential changes on the dual value. More precisely, the termination criterion is that the dual value in the penultimate iteration falls within the 95% confidence interval of the dual value in the terminal iteration. We then apply the direct policy evaluation scheme (cf. Table IV in D.1) to estimate the value of the policy obtained through our DDP algorithm. As shown by the last row of each subpart, the dual gap of the improved policy shrinks down to around 1%, strongly suggesting that the new policy is very close to the optimality. In addition, we find that, irrespective of the initial approximation that we start with, all the final outcomes that the DDP algorithm converges to are identical. Denote hereafter the policy we obtain by  $\mathbf{S}^{DDP}$ .

In this experiment, we use the DC programming to solve the inner optimization problem in the dual formulation. The consideration underlying this choice is that the penalty  $z$ , one part of the objective function of the inner optimization problem (cf. (19) and (25)), takes a very special form of functional difference. After decomposing the objective function to the difference of two convex functions in  $\mathbf{S}_t$ , we rely on sequential convex relaxation to transform the optimization job down to solving a sequence of convex programs. The literature has established the property of global convergence for this approach; that is, starting from any given initial point, the sequence generated by it converges to a solution to DC programs that satisfies the Karush-Kuhn-Tucker condition; see Yuille and Rangarajan (2003), Le Thi and Pham Dinh (2005), Sriperumbudur and Lanckriet (2009), Lu (2016), Le Thi and Pham Dinh (2018), and Boyd and Vandenberghe (2004). A brief introduction on the DC programming is also provided in Appendix C.

While in theory it is possible that the above sequential convex programming may only lead to local optimal solutions for the inner optimization problem, we need to stress that the numerical evidence shows that does not affect the optimality of  $\mathbf{S}^{DDP}$  reported in Table 1. To see this, we develop a sanity check in the following remark.

**Remark 5.1** *At the termination of the DDP algorithm, we expand the output dual value function  $\underline{\mathfrak{Y}}_t$  to its first order, i.e., for  $t = 0.1, \dots, T - 1$ ,*

$$\underline{\mathfrak{Y}}_t(\mathbf{X}, \mathbf{R}) \approx \underline{\mathfrak{Y}}_t(\mathbf{X}^0, \mathbf{R}^0) + \nabla_{\mathbf{x}} \underline{\mathfrak{Y}}_t(\mathbf{X}^0, \mathbf{R}^0)(\mathbf{X} - \mathbf{X}^0) + \nabla_{\mathbf{R}} \underline{\mathfrak{Y}}_t(\mathbf{X}^0, \mathbf{R}^0)(\mathbf{R} - \mathbf{R}^0) \quad (40)$$

where  $(\mathbf{X}^0, \mathbf{R}^0)$  is a state pair that we fix in advance, and  $\nabla_{\mathbf{x}}$  and  $\nabla_{\mathbf{R}}$  are the gradients with respect to  $\mathbf{X}$  and  $\mathbf{R}$ , respectively. Note that  $\underline{\mathfrak{Y}}_t$  is indeed a linear combination of the basis functions in (39). Hence, the function on the right hand side of (40) is explicitly known and it is linear in the variable  $\mathbf{R}$ . In Table 1 (cf. the dual value in the last row of each subpart), we use  $\underline{\mathfrak{Y}}_t$  to construct a duality to assess the quality of  $\mathbf{S}^{DDP}$ . Alternatively we may substitute the linear function on the right-hand side of (40) into (8) to construct another penalty. Note that the inner optimization problem in the resulting dual formulation



will become a convex program, which is globally solvable. So we do not need to worry about the issue of local solutions for this new duality. It turns out that the dual value we obtain in this way is 269.47 with a standard deviation 0.25, which is very close to the ultimate dual values reported in Table 1. This strongly suggests that the sequential convex programming procedure can effectively lead us to find a policy with adequate performance.

Recall that Theorem 4.5 reveals a crucial trade-off between the model complexity and the sampling adequacy facing us in the implementation of the DDP algorithm; that is, given the number of basis functions  $M$ , we need a sufficiently large number of samples  $L$  to ensure the convergence of the DDP algorithm. Both Table 2 and Figure 1 corroborate this conclusion. In Table 2, we can easily see that, for a fixed basis function set, there exists a minimum  $L$  for the DDP algorithm to converge. Moreover, as the number of basis functions used in the approximation increases, this critical  $L$  tends to become larger. Figure 1 empirically examines how fast this minimum  $L$  grows with  $M$  using the log-log plot. The slope suggests that the number of representative states  $L$  should be at least as large as  $O(M^{3/2})$  to ensure the convergence of the DDP algorithm. Note that this rate is much smaller than the theoretical rate established in Theorem 4.5. We leave the research on tightening the bound to future work.

	$L$									
$M$	500	1000	2000	3000	4000	5000	6000	7000	8000	9000
6	×	✓	✓	✓	✓	✓	✓	✓	✓	✓
11	×	×	×	✓	✓	✓	✓	✓	✓	✓
15	×	×	×	×	×	✓	✓	✓	✓	✓
21	×	×	×	×	×	×	×	✓	✓	✓
24	×	×	×	×	×	×	×	×	✓	✓

Table 2: The convergence performance of the DDP algorithm under different choices of  $L$  and  $M$ .

If it converges, we input  $\checkmark$  in the corresponding entry; otherwise, we use  $\times$ . The basis function sets we choose for each row are  $\{1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}\}$ ,  $\{1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}, (X_i^2)_{1 \leq i \leq 2}, (R_k^2)_{1 \leq k \leq 3}\}$ ,  $\{1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}, (X_i X_j)_{1 \leq i, j \leq 2}, (R_k R_l)_{1 \leq k, l \leq 3}\}$ ,  $\{1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}, (X_i X_j)_{1 \leq i, j \leq 2}, (R_k R_l)_{1 \leq k, l \leq 3}, (R_k X_j)_{1 \leq k \leq 3, 1 \leq j \leq 2}\}$  and  $\{1, (X_i)_{i=1,2}, (R_k)_{k=1,2,3}, (X_i X_j)_{1 \leq i, j \leq 2}, (R_k R_l)_{1 \leq k, l \leq 3}, (R_k X_j)_{1 \leq k \leq 3, 1 \leq j \leq 2}, (R_k \sqrt{R_k})_{1 \leq k \leq 3}\}$ , respectively.

Examining  $\mathbf{S}^{DDP}$  will shed more insights into this improved policy. In Figure 2, we compare it with  $\mathbf{S}^{LQ}$ , which is the policy derived from the simplified auxiliary problem, by simulating them on the same sample paths of the signal process  $\mathbf{X}_t$ . Let us assume that the trader receives a large signal at  $t = 1$ , i.e., the initial value  $\mathbf{X}_1$  is large. The top panel displays the evolution of the two-dimensional signal  $\mathbf{X}_t = (X_t^1, X_t^2)$  over time under different

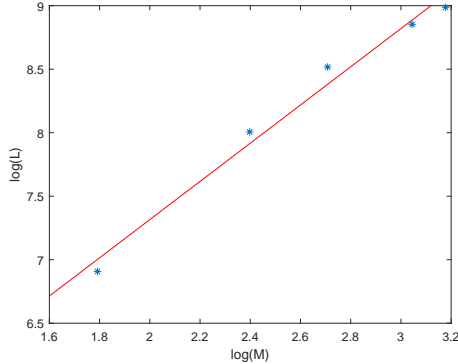


Figure 1: The regression result of the critical  $\log(L)$  against  $\log(M)$ . As shown in Table 2,  $L$  has to be as large as  $[1000, 3000, 5000, 7000, 8000]$ , respectively, for  $M = [6, 11, 15, 21, 24]$  to ensure the convergence of the DDP algorithm. The red straight line is the linear extrapolation between  $\log(L)$  and  $\log(M)$ . We have approximately  $\log(L) = 1.5 \log(M) + 4.3$ .

autocorrelation coefficient  $\delta$ . The other three rows illustrate how these two policies respond to the changes in  $\mathbf{X}_t$  in terms of the respective purchase amounts of the three assets. We can see that, in response to the “good” initial signal, the suboptimal strategy  $\mathbf{S}^{LQ}$  (the red curves in Figure 2) immediately increases its purchase. This behavior is economically sensible. Under our choice of  $\mathbf{C}$  in the dynamic of (32), a high current value of  $\mathbf{X}$  implies that the prices of the assets are likely to move up in the future. To avoid the high transaction cost that the trader might pay consequently, she would like to buy more at the current price immediately.

However, this policy is suboptimal in the presence of such market frictions as the price impact and the no-sale constraint. The blue curves in the figure illustrate how the optimal policy  $\mathbf{S}^{DDP}$  should behave. Interestingly, it executes transactions much more slowly in response to the same signal  $\mathbf{X}$  compared with  $\mathbf{S}^{LQ}$ . Moreover, the autocorrelation of the signal process accentuates the difference in the trading speeds between the two policies. As we increase the value of  $\delta$  from the left column to the right in Figure 2,  $\mathbf{S}^{LQ}$  and  $\mathbf{S}^{DDP}$  become distinct. We also examine the effect of the temporary impact by changing  $\lambda$  in the experiments. The green curves are corresponding to the case in which  $\lambda = 100$ . In comparison with the red curves ( $\lambda = 0$ ), the trader further smooths her transactions in order to avoid the excessive costs associated with the temporary impact of trading (cf. the first term in (34)).

## 5.2 Inventory Management with Lost Sales and Lead Time

In this section, we consider a single-item inventory management problem with stochastic demands, a constant lead time and lost sales. Assume that a manager has a finite planning horizon of  $T$  periods. In period  $t$ ,  $t = 1, 2, \dots, T$ , a random demand amounting to  $d_t$  will arise. All the demands across different periods are supposed to be independent and have the identical distribution. The manager needs to use the current inventory to meet the demand in each period and meanwhile determines an amount of  $a_t$  to order. Denote  $L$  to

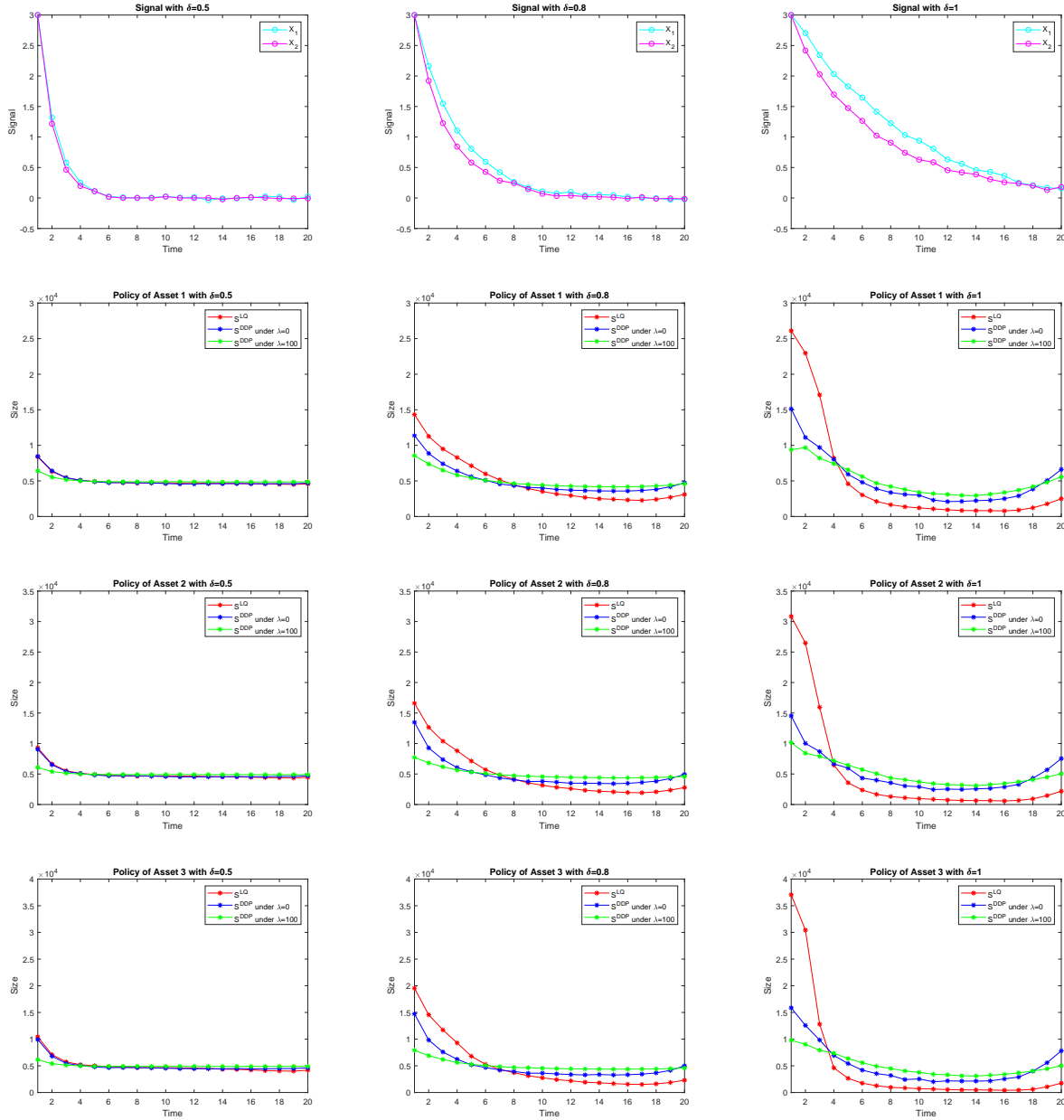


Figure 2: A simulation comparison between two policies  $\mathbf{S}^{DDP}$  and  $\mathbf{S}^{LQ}$ . We simulate 10,000 sample paths for random noises  $(\epsilon_t, \eta_t, t = 1, \dots, 20)$  to drive the model. Let  $\mathbf{X}_1 = [3, 3]^{tr}$ . The top panel plots the average values of  $\mathbf{X}$  in different time periods over all these sample paths. As we increase  $\delta$ , the decay in signal  $\mathbf{X}$  slows, indicating a stronger autocorrelation in the information process. The remaining rows display the average quantities of assets that the trader needs to buy under the policies  $\mathbf{S}^{DDP}$  and  $\mathbf{S}^{LQ}$  across these 10,000 paths of  $\mathbf{X}$  during each period.

be the order lead time; that is, the order placed in period  $t$  will arrive in period  $t + L$ . Hence, the manager's decision making should be based on a state vector of  $L$  components  $\mathbf{x}_t = (x_{0,t}, x_{1,t}, \dots, x_{L-1,t})$ , where  $x_{0,t}$  is the amount of the current inventory in period  $t$  and  $x_{l,t}$  is the order arriving in the subsequent periods  $t + l$  for  $l = 1, \dots, L - 1$ . If the current inventory is not sufficient, we assume that the unfulfilled demands will be immediately lost. After receiving  $x_{1,t}$  at the beginning of the next period, the inventory level transits to  $(x_{0,t} - d_t)^+ + x_{1,t}$  and the manager starts a new decision-making loop. From the above discussion, we can easily see that the state vector for period  $t + 1$  should be given by

$$\mathbf{x}_{t+1} = ((x_{0,t} - d_t)^+ + x_{1,t}, x_{2,t}, \dots, x_{L-1,t}, a_t). \quad (41)$$

Note that this dynamic is not linear.

The manager faces three types of costs: procurement cost associated with orders, inventory holding cost, and lost-sale penalty. For notational simplicity, we ignore the first type of cost in our model by letting the unit cost of procurement be 0. As argued in Janakiraman and Muckstadt (2004), this assumption will not hurt the generality of the setup. Let  $h$  and  $p$  denote the marginal cost of holding inventory and the penalty of lost sales, respectively. Then the manager attempts to minimize the discounted total cost over  $T + L$  periods, namely,

$$\min_{\substack{a_t \in \mathbb{Z}_+, \\ 1 \leq t \leq T+L}} \mathbb{E} \left[ \sum_{t=1}^{T+L} \gamma^t (h(x_{0,t} - d_t)^+ + p(d_t - x_{0,t})^+) \right], \quad (42)$$

where

$$q(\mathbf{x}_t, d_t) := h(x_{0,t} - d_t)^+ + p(d_t - x_{0,t})^+$$

is the sum of the inventory cost and the lost-sale penalty in period  $t$ ,  $\gamma \in (0, 1)$  is the discount factor used by the manager, and  $\mathbb{Z}_+$  stands for the set of all nonnegative integers.

The lost-sales model was first formulated in Karlin and Scarf (1958) and further explored in Morton (1969, 1971). It is well known that the model is intractable, especially for a large lead time  $L$ . Zipkin (2008a,b) presents insightful structural analysis on this standard problem and, based on that, tests several plausible heuristics. He finds that the following myopic policy yields analytical value functions and performs reasonably well. Rather than considering the entire time horizon, the myopic policy chooses the order quantity  $a_t$  in period  $t$  to minimize the cost from period  $t$  to period  $t + L$ . That is, letting

$$a_t^{\text{my}} = \arg \min_{a_t \in \mathbb{Z}_+} \mathbb{E} \left[ \sum_{s=t}^{t+L} \gamma^{s-t} q(\mathbf{x}_s, d_s) \right]. \quad (43)$$

Note that the order  $a_t$  arrives in period  $t + L$  and has nothing to do with the inventory prior to that period. Thus we can easily show that the optimization in (43) is equivalent to

$$a_t^{\text{my}} = \arg \min_{a_t \in \mathbb{Z}_+} \mathbb{E}[\gamma^L q(\mathbf{x}_{t+L}, d_{t+L})]. \quad (44)$$

This policy apparently neglects the evolution of the inventory system after period  $t + L$ .

Relatedly, Chen, Dawande, and Janakiraman (2014) develop a new numerical approach to approximate the optimal value function of this example using a selected number of points

in a bounded rectangular domain. Their method hinges on the  $L^\sharp$ -convex property of the value function. Bu, Gong and Yao (2017) analyze the asymptotic optimality of a given heuristic in an infinite-horizon lost-sales inventory model with positive lead time. Brown and Smith (2014) apply the information relaxation based dual method to assess the above myopic policy.

The following numerical experiments test the performance of the DDP algorithm by using it to assess and improve several heuristic policies. We assume that the stochastic demand  $d_t$  follows a geometric distribution with mean  $m$ . As pointed out by Zipkin (2008a), this distribution is more likely to produce extreme demand scenarios. Two possible lead times,  $L = 4$  and  $L = 10$ , are considered. As in the previous optimal execution problem, we need to choose a proper state selector  $G$  to sample the representative states  $\mathbf{x}_t$  in each period  $t$ . Let  $\theta = h/(p + h)$  and define

$$s_l = \min \left\{ s : \mathbb{P} \left( \sum_{m=l}^L d_m > s \right) \leq \theta \right\}$$

for  $l = 0, \dots, L - 1$ . Both Morton (1969) and Zipkin (2008a,b) show that, starting with initial state  $\mathbf{x}_1 = 0$ , the inventory process under the optimal policy will never leave the region

$$\mathcal{X}_t = \left\{ \mathbf{x}_t \geq 0 : \sum_{m=l}^{L-1} x_{m,t} \leq s_l, l = 0, \dots, L - 1 \right\}. \quad (45)$$

In light of these results, we take  $G$  to be the discrete uniform distribution over the compact set  $\mathcal{X}$ .

In the interest of space, we defer the explicit expressions of all the basis functions used in this section to Appendix E. To evaluate the penalty function, we need to calculate the expectations of these basis functions. This step may be computationally expensive when  $L$  is large. As mentioned in Section 4, we suggest using low-discrepancy sequences from the QMC literature to develop effective approximations. A detailed explanation of this approximation can also be found in E.

Along each sample path of demand  $\mathbf{d}_t = (d_t, d_{t+1}, \dots, d_{T+L})^{tr}$ , the DDP algorithm solves the following deterministic inner optimization problem

$$J(\mathbf{x}_t, \mathbf{d}_t) := \inf_{a \in \mathbb{Z}_+^{T-t+1}} \sum_{s=0}^{T+L-t} \left\{ \gamma^s q(\mathbf{x}_{t+s}, d_{t+s}, a_{t+s}) + z_t(a, \mathbf{d}_t) \right\} \quad (46)$$

at each time step  $t$  for the dual value determination. It can be reduced to an integer DC program. Maehara, Marumo, and Murota (2018) employ a special form of continuous relaxation (known as “lin-vex extension” in their paper) to find an exact solution to DC optimization programs with integer constraints. However, to save the computational effort, we take an alternative approach here by simply relaxing the integer constraint  $a \in \mathbb{Z}_+^{T-t+1}$  to  $a \geq 0$  when solving (46). The relaxation enables us to apply the sequential-convex-programming method in C to obtain a lower bound for  $J(\mathbf{x}_t, \mathbf{d}_t)$ . The numerical experiments show that the convergence of the DDP algorithm is not affected by this continuous relaxation.

Table 3 displays the performance of our DDP algorithm in improving some heuristic policies. In addition to the myopic policy given in (47), we also consider several alternative approximate policies as follows:

- Lookahead. The above myopic policy ignores the long run impact of the current order. To remedy this, we may introduce a  $\tilde{V}$  to approximately capture the future impact of the present order. A lookahead policy stems from solving

$$a_t^{\text{LA}} = \arg \min_{a_t \in \mathbb{Z}_+} \mathbb{E} \left[ \sum_{s=t}^{t+L} \gamma^{s-t} q(\mathbf{x}_s, d_s) + \gamma^{L+1} \tilde{V}(\mathbf{x}_{t+L+1}, d_{t+L+1}) \right]. \quad (47)$$

In the experiment, we try the total cost function at time  $t + L + 1$ ,  $q(\mathbf{x}_{t+L+1}, d_{t+L+1})$ , as the approximation  $\tilde{V}$ .

- Linear programming approximation. We omit the details here because the idea is similar to the LP approximation in the previous example.

It is worth mentioning that this problem is solvable through the associated Bellman equation when the lead time  $L = 4$ . Using (45), the total number of the states that we need to visit in each period in this case is 60,129. By brute-force searching for the best order quantities in all these states, we find that the true optimal value of the problem when  $L = 4$  should be 541.82 under the parameter values we set up for the experiment. The structure of Table 3 remains similar to that of Table 1. We can see that the DDP algorithm manages to significantly reduce down the duality gaps of all these heuristics.

Figures 3 and 4 help us gain more insight about where the improvement of the policy that the DDP finally converges to comes from. We simulate the inventory system under both the myopic policy and the policy obtained from the DDP method. The two types of costs, the inventory holding cost

$$\mathbb{E} \left[ \sum_{t=1}^{T+L} \gamma^t h(x_{0,t} - d_t)^+ \right]$$

and the lost sales penalty cost

$$\mathbb{E} \left[ \sum_{t=1}^{T+L} \gamma^t p(d_t - x_{0,t})^+ \right],$$

are calculated and compared in the figure. The myopic policy focuses on the short-term performance and neglects the long-run impact of orders on the inventory level. Therefore it incurs a smaller lost-sale penalty than the optimal one. However, this comes at the expense of the inventory holding cost. In contrast, the improved policy that resulted from the DDP algorithm strikes a better balance between these two costs. The inventory cost under it is smaller than what the myopic policy causes, which leads to a better overall cost performance. Figure 5 further compares the average inventory level for a system controlled by both policies and subject to the same demand shocks over the time horizon. It clearly demonstrates that the system tends to build up more inventory, thus incurring more holding costs, if the manager uses the myopic policy.

L=4						
Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
Myopic	0			563.72	(0.42)	4.36%
	1	539.16	(0.38)	-	-	-
	2	539.86	(0.09)	542.00	(0.43)	0.39%
	3	539.88	(0.08)			
Lookahead	0			560.13	(0.41)	3.63%
	1	539.78	(0.36)	-	-	-
	2	539.88	(0.08)	542.13	(0.43)	0.41%
	3	539.89	(0.08)			
Linear	0			566.31	(0.50)	5.00%
	1	537.86	(0.40)	-	-	-
	2	539.80	(0.08)	542.08	(0.40)	0.41%
	3	539.84	(0.08)			
L=10						
Approximation	Iteration	Dual Values	(SE)	Primal Values	(SE)	Gap
Myopic	0			829.63	(0.28)	7.36%
	1	768.58	(0.36)	-	-	-
	2	770.93	(0.08)	-	-	-
	3	771.80	(0.08)	779.36	(0.29)	0.96%
	4	771.89	(0.07)			
Lookahead	0			827.14	(0.30)	7.04%
	1	768.91	(0.36)	-	-	-
	2	771.10	(0.08)	-	-	-
	3	771.83	(0.08)	779.54	(0.29)	0.99%
	4	771.84	(0.08)			
Linear	0			844.28	(0.30)	9.50%
	1	764.10	(0.35)	-	-	-
	2	770.59	(0.09)	-	-	-
	3	771.61	(0.08)	-	-	-
	4	771.89	(0.08)	779.80	(0.30)	1.02%
	5	771.83	(0.08)			

Table 3: Convergence results of the DDP algorithm under different initial policies in the example of inventory management. The default parameters used in this experiment are  $\{m = 4, h = 1, p = 9, T = 30\}$ . Two lead times are implemented, i.e.,  $L = 4$  and  $L = 10$ . The three types of heuristic policies used as the inputs are Myopic, Lookahead, and Linear. We sample 500 states for  $L = 4$  and 1000 states for  $L = 10$  in each time step from the distribution  $G$  to compute the dual values. In LP, we simulate 100 states to set up the corresponding constraints. All computation experiments are conducted on a PC equipped with an Intel Xeon 32-core 2.93 GHz CPU and 12.0 GB of RAM. The computation environment is Windows 7 and MATLAB R2017a and parallel pool. The average computational time is 323.6s per iteration for  $L = 4$  and 2365.1s for  $L = 10$ .

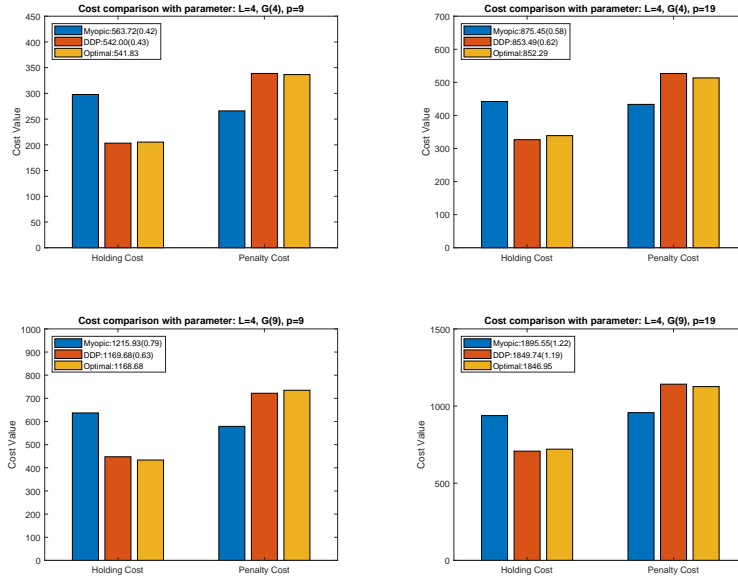


Figure 3: Cost comparison in  $L = 4$ . We compare the holding and penalty costs under three policies: the myopic policy  $a^{my}$ , the policy improved from the DDP algorithm  $a^{DDP}$ , and the optimal one  $a^{op}$ . We simulate 10,000 sample paths of random demands and evaluate both  $a^{my}$  and  $a^{DDP}$  based on the same set of sample paths. We use brute-force searching to solve the Bellman equation to obtain the optimal value for  $a^{op}$ .  $G(4)/G(9)$  stands for the geometric random demand with mean  $4/9$ . In the legend of each subfigure, we report the total cost of each policy and the corresponding standard error in the brackets. Note that  $a^{DDP}$ , the resulted policy from our DDP algorithm, behaves exactly the same as the optimal one.



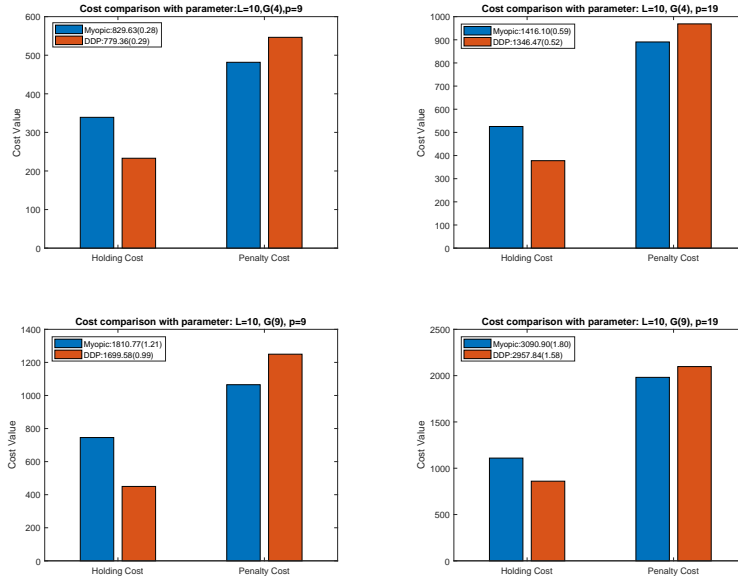


Figure 4: Cost comparison in  $L = 10$ . Note that we do not report the costs associated with the optimal policy because it is impossible to apply the Bellman equation to solve for the optimal solution due to the high dimensionality.

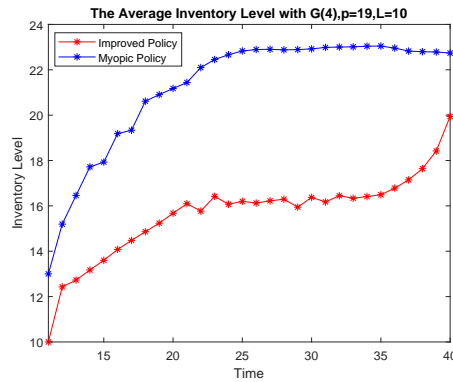


Figure 5: Average inventory level for a system controlled by myopic policy  $a^{my}$  and improve policy  $a^{DDP}$ . The parameter settings are:  $L = 10$ ,  $p = 19$ ,  $T = 30$  and the geometric demand distribution with mean 4. In this figure we sample 10,000 random demands  $\{d_t, 1 \leq t \leq T + L\}$ . Under the same sample path, we run the myopic and improved policies. The two curves in the figure display the average inventory level at each time step across all the sampled demands.

## 6 Conclusions

In this paper we present a duality-driven iterative approach (DDP) for solving a general SDP problem. The duality gap yielded by the method can be used to assess the performance of a given policy. More importantly, repeatedly applying the dual operation on the basis of the technique of information relaxation will lead to policy improvement and convergence to the optimality. To implement the DDP, we also develop a regression Monte Carlo method. In conjunction with such techniques as DC programming and parallel computing, our method demonstrates numerical effectiveness and accuracy in dealing with multidimensional complex SDP problems.

**Acknowledgments:** This research is supported by the Research Grant Council Hong Kong through the scheme of General Research Fund (Grant No. 14237616 and 14207918), and the National Natural Science Foundation of China (Grant No. 71991474, 71721001, and U1811462).

## References

- Almgren R, Chriss N (2000) Optimal Execution of Portfolio Transactions. *J. Risk* **3**: 5–39.
- Andersen L, Broadie M (2004) Primal-dual Simulation Algorithm for Pricing Multidimensional American Options. *Management Sci.* **50**: 1222–1234.
- Balseiro SR, Brown DB (2019) Approximations to Stochastic Dynamic Programs via Information Relaxation Duality. *Oper. Res.* **67(2)**: 577-597.
- Balseiro SR, Brown DB, Chen C (2018) Static Routing in Stochastic Scheduling: Performance Guarantees and Asymptotic Optimality. *Oper. Res.* **66(6)**: 1641-1660.
- Bemporad A, Morari M, Dua V, Pistikopoulos EN (2002) The explicit linear quadratic regulator for constrained systems. *Automatica* **38**: 3–20.
- Bertsekas DP (1995) *Dynamic Programming and Optimal Control*, Vol. 1. Athena Scientific, Belmont, Massachusetts, USA.
- Bertsekas DP (1997) *Dynamic Programming and Optimal Control*, Vol. 2. Athena Scientific, Belmont, Massachusetts, USA.
- Bertsekas DP (2019) *Reinforcement Learning and Optimal Control*, Athena Scientific, Belmont, Massachusetts, USA.
- Bertsekas DP, Tsitsiklis JN (1996) *Neuro-Dynamic Programming*, Athena Scientific, Belmont, Massachusetts, USA.
- Bertsimas D, Lo AW (1998) Optimal Control of Execution Costs. *J. Financ. Mark.* **1**: 1–50.
- Bouchaud JP, Farmer D, Lillo F (2009) How Markets Slowly Digest Changes in Supply and Demand. In *Handbook of Financial Markets: Dynamics and Evolution*. North-Holland (Elsevier), Amsterdam.

- Boyd, S. and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press, Cambridge, UK.
- Brown DB, Haugh MB (2017) Information Relaxation Bounds for Infinite Horizon Markov Decision Processes. *Oper. Res.* **65**: 1355–1379.
- Brown DB, Smith JE (2011) Dynamic Portfolio Optimization with Transaction Costs: Heuristics and Dual Bounds. *Management Sci.* **57**: 1752–1770.
- Brown DB, Smith JE (2014) Information Relaxations, Duality and Convex Stochastic Dynamic Programs. *Oper. Res.* **62**: 1394–1415.
- Brown DB, Smith JE (2020) Index Policies and Performance Bounds for Dynamic Selection Problems. *Management Sci.* Forthcoming.
- Brown DB, Smith JE, Sun P (2010) Information Relaxations and Duality in Stochastic Dynamic Programs. *Oper. Res.* **58**: 785–801.
- Bu J, Gong X, Yao D (2017) Constant-order Policies for Lost-sales Inventory Models with Random Supply Functions: Asymptotics and Heuristic. Available at SSRN: <https://ssrn.com/abstract=3063730>.
- Carrière J (1996) Valuation of Early-Exercise Price of Options Using Simulations and Non-parametric Regression. *Insur. Math. Econ.* **19**: 19–30.
- Chen W, Dawande M, Janakiraman G (2014) Fixed-dimensional stochastic dynamic programs: An approximation scheme and an inventory application. *Oper. Res.* **62**: 81–103.
- Chen N, Glasserman P (2007) Additive and Multiplicative Duals for American Option Pricing. *Financ. Stoch.* **11**: 153–179.
- Chen N, Kou S, Wang C (2018) A Partitioning Algorithm for MDPs and Market Microstructure. *Management Sci.* **64**: 784–803.
- Davis MHA (1989) Anticipative LQG control. *IMA J. Math. Control I.* **6**: 259–265.
- Davis MHA (1991) Anticipative LQG control II. In *Applied Stochastic Analysis*, Eds. by M. H. A. Davis and R. J. Elliott. Gordon and Breach Science Publishers, New York, NY.
- Davis MHA, Zervos M (1995) A New Proof of the Discrete-Time LQG Optimal Control Theorems. *IEEE Trans. Automat. Contr.* **40**: 1450–1453.
- de Farias DP, Van Roy B (2003) The linear programming approach to approximate dynamic programming. *Oper. Res.* **51**: 850–865.
- de Farias DP, Van Roy B (2004) On constraint sampling in the linear programming approach to approximate dynamic programming. *Math. Oper. Res.* **293**: 462–478.
- Desai VV, Farias VF, Moallemi CC (2012a) Approximate dynamic Programming via a Smoothed Approximate Linear Program. *Oper. Res.* **60**: 655 – 674.

- Desai VV, Farias VF, Moallemi CC (2012b) Pathwise optimization for optimal stopping problems. *Management Sci.* **58**: 2292 – 2308.
- Desai VV, Farias VF, Moallemi CC (2013) Bounds for Markov Decision Processes. In *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*, Eds. by F. L. Lewis, D. Liu, eds. IEEE Press.
- Devalkar S, Anupindi R, Sinha A (2011) Integrated optimization of procurement, processing, and trade of commodities. *Oper. Res.* **59**: 1369–1381.
- Gârleanu N, Pedersen LH (2013) Dynamic Trading with Predictable Returns and Transaction Costs. *J. Finance* **68**: 2309–2340.
- Gârleanu N, Pedersen LH (2016) Dynamic Portfolio Choice with Frictions. *J. Econ. Theory* **165**: 487–516.
- Gatheral J (2010) No-Dynamic-Arbitrage and Market Impact. *Quant. Finance* **10**: 749–759.
- Goodson JC, Ohlmann JW, Thomas BW (2013) Rollout policies for dynamic solutions to the multivehicle routing problem with stochastic demand and duration limits. *Oper. Res.* **61**: 138–54.
- Hartman P (1959) On Functions Representable as a Difference of Convex Functions. *Pac. J. Math.* **9**: 707–713.
- Haugh MB, Iyengar G, Wang C (2016) Tax-aware dynamic asset allocation. *Oper. Res.* **64**: 849–866.
- Haugh MB, Kogan L (2004) Pricing American Options: a Duality Approach. *Oper. Res.* **52**: 258–270.
- Haugh MB, Lim AEB (2012) Linear-Quadratic Control and Information Relaxation. *Oper. Res. Lett.* **40**: 521–528.
- Haugh MB, Ruiz-Lacedelli O (2018) Information relaxation bounds for partially observed Markov decision processes. *Working Paper*.
- Haugh MB, Wang C (2014) Dynamic Portfolio Execution and Information Relaxation. *SIAM J. Financ. Math.* **5**: 316–359.
- Hernández-Lerma O, Lasserre JB (1997) *Discrete-Time Markov Control Processes: Basic Optimality Criteria*, Springer-Verlag, New York, USA.
- Horst R, Thoai NV (1999) DC Programming: Overview. *J. Optimiz. Theory App.* **103**: 1–13.
- Huberman G, Stanzl W (2004) Price Manipulation and Quasi-Arbitrage. *Econometrica* **72**, 1247–1275.
- Huberman G, Stanzl W (2005) Optimal Liquidity Trading. *Rev. Finance* **9**: 165–200.

- Janakiraman G, Muckstadt J (2004) Inventory Control in Directed Networks: A Note on Linear Costs. *Oper. Res.* **52**: 491–495.
- Karlin S, Scarf H (1958) Inventory models of the Arrow-Harris-Marschak type with time lag. In *Studies in the Mathematical Theory of Inventory and Production*, Eds. by K. Arrow, S. Karlin, and H. Scarf. Stanford University Press, Stanford.
- Kim MJ, Lim AEB (2016) Robust multiarmed bandit problems. *Management Sci.* **62**: 264–285.
- Lai G, Margot F, Secomandi N (2010) An approximate dynamic programming approach to benchmark practice-based heuristics for natural gas storage valuation. *Oper. Res.* **58**: 564–582.
- Lai G, Wang MX, Kekre S, Scheller-Wolf A, Secomandi N (2011) Valuation of storage at a liquefied natural gas terminal. *Oper. Res.* **59**: 602–616.
- Le Thi HA, Pham Dinh T (2005) The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. *Ann. Oper. Res.* **133**: 23–48.
- Le Thi HA, Pham Dinh T (2018) DC programming and DCA: thirty years of developments. *Math. Program. B* **169**: 5–68.
- Longstaff FA, Schwartz ES (2001) Valuing American Options by Simulation: a Simple Least-Squares Approach. *Rev. Financial Stud.* **14**: 113–147.
- Lu Z (2016) Sequential Convex Programming Methods for a Class of Structured Nonlinear Programming. Working Paper of Simon Fraser University, Canada.
- Maehara T, Marumo J, Murota J (2018) Continuous Relaxation for Discrete Dc Programming. *Math. Program.* **169**: 199–219.
- Morton T (1969) Bounds on the Solution of the Lagged Optimal Inventory Equation with No Demand Backlogging and Proportional Costs. *SIAM Rev.* **11**: 572–576.
- Morton T (1971) The Near-Myopic Nature of the Lagged-Proportional-Cost Inventory Problems with Lost Sales. *Oper. Res.* **19**: 1708–1716.
- Nocedal J, Wright SJ (1999) *Numerical Optimization*. Springer-Verlag, New York.
- Powell WB (2011) *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Second Edition. John Wiley & Sons, Hoboken, New Jersey.
- Putman ML (1994) *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Hoboken, New Jersey.
- Rockafellar RT, Wets RJB (1976) Nonanticipativity and  $L^1$  martingales in stochastic optimization problems, *Math. Program. Stud.* **6**: 170–187.

- Rogers LCG (2002) Monte Carlo valuation of American options. *Math. Finance* **12**: 271–286.
- Rogers LCG (2007) Pathwise Stochastic Optimal Control. *SIAM J. Control Optim.* **46**: 1116–1132.
- Schweitzer P, Seidmann A (1985) Generalized polynomial approximations in Markovian decision processes. *J. Math. Anal. Appl.* **110**: 568–582.
- Sriperumbudur BK, Lanckriet GR (2009) On the convergence of the concave-convex procedure. *Adv. Neural Inf. Process Syst.* **22** (NIPS 2009): 1759–1767.
- Tsitsiklis J, Van Roy B (1999) Optimal Stopping of Markov Processes: Hilbert Space Theory, Approximation Algorithms, and an Application to Pricing High-Dimensional Financial Derivatives. *IEEE Trans. Automat. Contr.* **44**: 1840–1851.
- Tsitsiklis J, Van Roy B (2001) Regression Methods for Pricing Complex American-Style Options. *IEEE Trans. Neural Netw.* **12**: 694–703.
- Wang Y, Boyd S (2009) Performance bounds for linear stochastic control. *Syst. Control Lett.* **58**: 178–182.
- Whittle P (1982) *Optimization Over Time*. John Wiley & Sons, Inc., New York, USA.
- Ye F, Zhou E (2015) Information relaxation and dual formulation of controlled Markov diffusions. *IEEE Trans. Automat. Contr.* **60**:2676–2691.
- Yuille AL, Rangarajan A (2003) The Concave-Convex Procedure. *Neural Comput.* **15**: 915–936.
- Zipkin P (2008a) On the Structure of Lost-Sales Inventory Models. *Oper. Res.* **56**: 937–944.
- Zipkin P (2008b) Old and New Methods for Lost-Sales Inventory Systems. *Oper. Res.* **56**: 1256–1263.

# Appendix

## A Proofs of Main Results in Section 3.1

Before proving Proposition 3.2, we establish a Bellman equation-like characterization of the value of the inner optimization problem in the dual formulation. Let

$$\mathfrak{J}_t(\xi|t, x_t) = \inf_{a \in A|t} \left[ \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t(a, \xi) \right]$$

with  $z_t(a, \xi)$  being defined as in (8). We have

**Lemma A.1** For  $0 \leq t \leq T - 1$ ,

$$\mathfrak{J}_t(\xi|t, x_t) = \inf_{a_t \in A_t} \left\{ \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})|x_t = x] - W_{t+1}(x_{t+1}) + \mathfrak{J}_{t+1}(\xi|t+1, x_{t+1}) \right\}.$$

*Proof of Lemma A.1.* Note that  $\mathfrak{J}_t(\xi|t, x_t)$  admits the following representation

$$\begin{aligned} \mathfrak{J}_t(\xi|t, x_t) &= \inf_{a_t \in A_t} \left\{ r_t(x_t, a_t, \xi_t) + \mathbb{E}[r_s(x_s, a_s, \xi_s) + W_{s+1}(f_s(x_s, a_s, \xi_s))] - (r_s(x_s, a_s, \xi_s) \right. \\ &\quad \left. + W_{s+1}(f_s(x_s, a_s, \xi_s))) + \inf_{a \in A|(t+1)} \left[ \sum_{s=t+1}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_{t+1}(a, \xi) \right] \right\}. \end{aligned}$$

The conclusion trivially follows.  $\square$

*Proof of Proposition 3.2.* By the definition of the dual operator  $\mathcal{D}$ , we have

$$\mathcal{D}W_t(x) = \mathbb{E} \left[ \inf_{a \in A|t} \left[ \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + z_t(a, \xi) \right] \middle| x_t = x \right] = \mathbb{E} \left[ \mathfrak{J}_t(\xi|t, x_t) \middle| x_t = x \right]. \quad (48)$$

According to Lemma A.1, for any action  $a_t \in A_t$ ,

$$\mathfrak{J}_t(\xi|t, x_t) \leq \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})|x_t] - W_{t+1}(x_{t+1}) + \mathfrak{J}_{t+1}(\xi|t+1, x_{t+1}).$$

Therefore,

$$\mathbb{E} \left[ \mathfrak{J}_t(\xi|t, x_t) \middle| x_t = x \right] \leq \mathbb{E} \left[ \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})|x_t] - W_{t+1}(x_{t+1}) + \mathfrak{J}_{t+1}(\xi|t+1, x_{t+1}) \middle| x_t = x \right]. \quad (49)$$

Note that, by the iterated law of conditional expectation, we have

$$\mathbb{E} \left[ \mathbb{E}[W_{t+1}(x_{t+1})|x_t] - W_{t+1}(x_{t+1}) \middle| x_t = x \right] = 0.$$

Moreover,

$$\mathbb{E}[\mathfrak{J}_{t+1}(\xi|t+1, x_{t+1})|x_t] = \mathbb{E}[\mathbb{E}[\mathfrak{J}_{t+1}(\xi|t+1, x_{t+1})|x_t = x]] = \mathbb{E}[\mathcal{D}W_{t+1}(x_{t+1})|x_t = x].$$

Both equalities lead to that the right hand of (49) is equal to

$$\mathbb{E}[r_t(x_t, a_t, \xi_t) + \mathcal{D}W_{t+1}(x_{t+1})|x_t = x]$$

In conjunction with (48), we have

$$\mathcal{D}W_t(x) \leq \inf_{a_t \in A_t} \mathbb{E}[r_t(x, a_t, \xi_t) + \mathcal{D}W_{t+1}(f_t(x, a_t, \xi_t))]. \square$$

*Proof of Theorem 3.3.* (i) To show this, we need the following two claims:

- For any sequence  $W = (W_0, W_1, \dots, W_T)$ ,  $\mathcal{D}^n W$  is a subsolution for any  $n \geq 1$ .
- For a subsolution sequence  $W = (W_0, W_1, \dots, W_T)$ ,  $W_t : \mathbb{R}^n \rightarrow \mathbb{R}$ , we have

$$W_t(x) \leq (\mathcal{D}W)_t(x) \text{ for } 0 \leq t \leq T - 1$$

$$\text{and } W_T(x) = (\mathcal{D}W)_T(x) = r_T(x).$$

According to Proposition 3.2,  $\mathcal{D}^n W$  is a subsolution for any  $n \geq 1$ . Now we turn to the second claim. It suffices to prove that, for any sequence of subsolution  $W$ , we have  $\mathfrak{J}_t(\xi|t, x_t) \geq W_t(x)$  for all  $t \geq 0$  with  $\mathfrak{J}_t(\xi|t, x_t)$  defined in Lemma A.1. Indeed, invoking (48),

$$\mathcal{D}W_t(x) = \mathbb{E}\left[\mathfrak{J}_t(\xi|t, x_t) \middle| x_t = x\right] \geq W_t(x).$$

We prove the claim of  $\mathfrak{J} \geq W$  by performing induction on  $t$ . For  $t = T$ , it is clearly true since  $\mathfrak{J}_T = r_T = W_T$  by the definition. Suppose for  $s \geq t + 1$ , the claim  $\mathfrak{J}_s \geq W_s$  holds. Then at time  $t$ , according to Lemma A.1,

$$\begin{aligned} \mathfrak{J}_t(\xi|t, x_t) &= \inf_{a_t \in A_t} \left\{ \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})] - W_{t+1}(x_{t+1}) + \mathfrak{J}_t(\xi|t, x_{t+1}) \right\} \\ &\geq \inf_{a_t \in A_t} \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})]. \end{aligned}$$

As  $W$  is a subsolution sequence, we have

$$\mathfrak{J}_t(\xi|t, x_t) \geq \inf_{a_t \in A_t} \mathbb{E}[r_t(x_t, a_t, \xi_t) + W_{t+1}(x_{t+1})] \geq W_t(x_t).$$

That completes the induction loop.

(ii) Consider a subsolution sequence  $W$  such that  $\mathcal{D}W = W$ . We use induction again to prove this part. At the last period  $T$ , we know that  $(\mathcal{D}W)_T(x) = r_T(x)$  for all  $x$  according to the definition of the operator  $\mathcal{D}$ . Hence,

$$W_T(x) = (\mathcal{D}W)_T(x) = r_T(x) = V_T(x).$$

In words,  $W$  and  $V$  coincide at  $T$ . Now we assume this claim also holds for  $s \geq t + 1$  for some  $t$ . By Proposition 3.2, the sequence  $\mathcal{D}W$  constitutes a subsolution. Therefore,

$$\begin{aligned} (\mathcal{D}W)_t(x) &\geq \inf_{a_t \in A_t} \mathbb{E}[r_t(x_t, a_t, \xi_t) + (\mathcal{D}W)_{t+1}(f_t(x_t, a_t, \xi_t))|x_t = x] \\ &= \inf_{a_t \in A_t} \mathbb{E}[r_t(x, a_t, \xi_t) + W_{t+1}(f_t(x, a_t, \xi_t))]. \end{aligned}$$



In addition, the right hand side of the above inequality equals to, by the induction hypothesis,

$$\inf_{a_t \in A_t} \mathbb{E}[r_t(x, a_t, \xi_t) + V_{t+1}(f_t(x, a_t, \xi_t))] = V_t(x).$$

where the last equality is due to the fact that  $V$ , as the true value function, should satisfy the Bellman equation. These lead to  $(\mathcal{D}W)_t(x) \geq V_t(x)$ . On the other hand,  $(\mathcal{D}W)_t(x) \leq V_t(x)$  because of the weak duality property. Thus, we should have  $(\mathcal{D}W)_t(x) = V_t(x)$ , which completes the induction loop.

(iii) First, we claim that, if for some  $n$  and  $t$  the equality  $(\mathcal{D}^n W)_{t+1}(x) = V_{t+1}(x)$  holds for all  $x$ , then

$$(\mathcal{D}^k W)_t(x) = V_t(x),$$

for any  $k \geq n + 1$ . This claim can be easily proved by induction. We omit the detail here for the interest of space. Once this claim is established, noting that  $(\mathcal{D}^1 W)_T(x) = V_T(x)$  is true, it is easy to see that  $(\mathcal{D}^2 W)_{T-1}(x) = V_{T-1}(x)$  must be true for all state  $x$ . Using the above claim again, we can reach the following conclusion:

$$(\mathcal{D}^3 W)_{T-2}(x) = V_{T-2}(x) \quad \text{and} \quad (\mathcal{D}^3 W)_{T-1}(x) = V_{T-1}(x).$$

Repeatedly using the above argument leads to, for a general  $k$ ,

$$(\mathcal{D}^k W)_t(x) = V_t(x) \text{ for } t \geq T + 1 - k.$$

In particular, when  $k = T + 1$ , we have  $(\mathcal{D}^{T+1} W)_t(x) = V_t(x)$  for  $t \geq 0$ . The theorem statement is proved.  $\square$

## B The DDP Method in LQC

We need the following technical lemma in calculating the duality of LQC.

**Lemma B.1** *We consider the quadratic programming*

$$J_t = \sum_{s=t}^{T-1} (x_s^{tr} Q_s x_s + a_t^{tr} R_t a_t + 2\alpha_t^{tr} x_t + 2\beta_t^{tr} a_t) + x_T^{tr} Q_T x_T,$$

*with the equality constraints*

$$x_{t+1} = D_t x_t + B_t a_t + \xi_t,$$

*where  $\xi_t \in \mathbb{R}^n$ ,  $\alpha_t \in \mathbb{R}^m$ ,  $\beta_t \in \mathbb{R}^m$  are given vectors,  $Q_t \in \mathbb{R}^{n \times n}$  and  $R_t \in \mathbb{R}^{m \times m}$  are positive semi-definite symmetric and positive definite symmetric matrix, respectively. Then, the optimal solution and minimum cost are given by*

$$\begin{aligned} a_t &= -L_t x_t - \theta_t^{-1} m_t, \\ J_t &= x_t^{tr} K_t x_t + 2n_t^{tr} x_t + \sum_{s=t}^{T-1} (\xi_s^{tr} K_{s+1} \xi_s + 2n_{s+1}^{tr} \xi_s - m_s^{tr} \theta_s^{-1} m_s), \end{aligned}$$

with

$$\begin{aligned} m_t &= B_t^{tr} (K_{t+1} \xi_t + n_{t+1}) + \beta_t, \\ n_t &= (D_t - B_t L_t)^{tr} [n_{t+1} + K_{t+1} \xi_t] + \alpha_t - L_t^{tr} \beta_t, \quad 0 \leq t \leq T-1, \quad n_T = 0. \end{aligned}$$

$\theta_t$ ,  $L_t$  and  $K_t$  are defined as

$$\begin{aligned} K_t &= D_t^{tr} (K_{t+1} - K_{t+1} B_t \theta_t^{-1} B_t^{tr} K_{t+1}) D_t + Q_t, \quad t = 0, \dots, T-1. \quad K_T = Q_T. \\ L_t &= \theta_t^{-1} B_t^{tr} K_{t+1} D_t, \quad \theta_t = R_t + B_t^{tr} K_{t+1} B_t. \end{aligned}$$

*Proof of Lemma B.1.* This statement can be established as a straightforward application of the well known analytical expression of the solution to a quadratic program with equality constraints; see Nocedal and Wright (1999), Chapter 16.  $\square$

Now we proceed to demonstrate how to apply the DDP algorithm to the LQC problem in detail.

– **Problem description:** Solve

$$\begin{aligned} \min_{\alpha \in \mathcal{A}_{\mathbb{F}}} \mathbb{E} \left[ \sum_{t=0}^{T-1} (x_t^{tr} Q_t x_t + \alpha_t^{tr} R_t \alpha_t) + x_T^{tr} Q_T x_T \right], \\ \text{s.t.} \quad x_{t+1} = D_t x_t + B_t \alpha_t + \xi_t, \quad t = 0, \dots, T-1. \end{aligned}$$

– **Solution:** It is well known that the above control problem admits closed form solutions. For  $t = 0, \dots, T-1$ , the optimal policy should be  $\alpha_t^*(x) = -L_t x$ , where the matrix  $L_t \in \mathbb{R}^{m \times n}$  is given by

$$L_t = (R_t + B_t^{tr} K_{t+1} B_t)^{-1} B_t^{tr} K_{t+1} D_t.$$

Here all matrices  $K_t \in \mathbb{R}^{n \times n}$  are positive semidefinite symmetric, and we can use the following recursive relationship to determine them:

$$\begin{aligned} K_T &= Q_T; \\ K_t &= D_t^{tr} (K_{t+1} - K_{t+1} B_t (R_t + B_t^{tr} K_{t+1} B_t)^{-1} B_t^{tr} K_{t+1}) D_t + Q_t, \quad t = 0, \dots, T-1. \end{aligned}$$

Under such a linear policy, the optimal cost function equals

$$V_t(x) = x^{tr} K_t x + \sum_{s=t}^{T-1} \mathbb{E} [\xi_s^{tr} K_{s+1} \xi_s].$$

*Proof of Proposition 3.4.* Consider a policy of the linear form

$$\alpha_t(x) = P_t x + E_t. \tag{50}$$

The subsequent calculation shows that we can achieve the optimal policy and value function of the LQC problem in two iterations by the DDP algorithm.

1. **First iteration.** Under linear policy (50), it is easy to verify that the cost-to-go function is quadratic with respect to states:

$$W_t^0(x) = x^{tr} H_t x + 2F_t^{tr} x + C_t,$$

with

$$\begin{aligned} H_t &= Q_t + P_t^{tr} R_t P_t + (D_t + B_t P_t)^{tr} H_{t+1} (D_t + B_t P_t), & H_T &= Q_T, \\ F_t &= P_t^{tr} R_t^{tr} E_t + (D_t + B_t P_t)^{tr} H_{t+1} B_t E_t, & F_T &= 0, \\ C_t &= C_{t+1} + E_t^{tr} R_t E_t + E_t^{tr} B_t^{tr} H_{t+1} B_t E_t + \mathbb{E}[\xi_t H_{t+1} \xi_t] + 2F_{t+1}^{tr} B_t E_t, & C_T &= 0. \end{aligned}$$

Given  $W_t^0$ , we can construct the penalty function  $z_t^1(a, \xi)$  by

$$\begin{aligned} z_t^1(a, \xi) &= \sum_{s=t}^{T-1} \left\{ \mathbb{E}[W_{s+1}^0(D_s x_s + B_s a_s + \xi_s)] - W_{s+1}^0(D_s x_s + B_s a_s + \xi_s) \right\} \\ &= \sum_{s=t}^{T-1} \left\{ -2\xi_s^{tr} (F_s + H_{s+1}(D_s x_s + B_s a_s)) - \xi_s^{tr} H_{s+1} \xi_s + \mathbb{E}[\xi_s^{tr} H_{s+1} \xi_s] \right\}. \end{aligned}$$

Then, the dual value in the first iteration satisfies

$$\begin{aligned} \underline{V}_t^1(x) &= \mathbb{E} \left[ \inf_{a \in A|t} \left\{ \sum_{s=t}^{T-1} \left( x_s^{tr} Q_s x_s + a_s^{tr} R_s a_s - 2\xi_s^{tr} (F_s + H_{s+1}(D_s x_s + B_s a_s)) \right. \right. \right. \\ &\quad \left. \left. \left. - \xi_s^{tr} H_{s+1} \xi_s + \mathbb{E}[\xi_s^{tr} H_{s+1} \xi_s] \right) + x_T^{tr} Q_T x_T \right\} \middle| x_t = x \right]. \end{aligned} \quad (51)$$

Using Lemma B.1, we can explicitly solve the inner optimization problem in (51). It is a quadratic program. That leads to

$$\underline{V}_t^1(x) = V_t(x) - \mathbb{E} \left[ \sum_{s=t}^{T-1} m_s^{tr} (R_s + B_s^{tr} K_{s+1} B_s)^{-1} m_s \right], \quad t = 0, \dots, T-1,$$

with

$$\begin{aligned} m_t &= B_t^{tr} (K_{t+1} - H_{t+1}) \xi_t + B_t^{tr} n_{t+1}, \\ n_t &= (D_t - B_t L_t)^{tr} [n_{t+1} + (K_{t+1} - H_{t+1}) \xi_t], \quad 0 \leq t \leq T-1, \quad n_T = 0. \end{aligned}$$

2. **Second iteration.** Note that  $\underline{V}_t^1(x)$  is represented as the optimal value function  $V_t(x)$  minus some constant. Hence, it is easy to see that  $z_t^2(a, \xi)$  is the optimal penalty function if we use  $\underline{V}_t^1(x)$  to construct it. From this observation, we can calculate out that the dual value after the second iteration satisfy

$$\underline{V}_t^2(x) = V_t(x), \quad \alpha_t^2(x) = \alpha_t^*(x). \square$$

## C DC Optimization

In this appendix we briefly review some primary facts about DC functions and the related optimization problem. A function  $f$  is called a DC function if there exist convex functions,  $g$  and  $h$ :  $\mathbb{R}^n \rightarrow \mathbb{R}$  such that  $f$  can be decomposed to the difference between  $g$  and  $h$ :

$$f(x) = g(x) - h(x), \quad \forall x \in \mathbb{R}^n.$$

The set of DC functions has a very rich structure. For instance, Lemma C.1 points out that the class of DC functions is closed under some algebraic operations such as addition, multiplication, and max/min.

**Lemma C.1 (Theorem 4.1 in Horst, Pardalos, and Thoai (2000))** *If  $f_1$  and  $f_2$  are two DC functions, then the following functions are also DC:*

- (a)  $\lambda_1 f_1(x) + \lambda_2 f_2(x)$  for any constants  $\lambda_1$  and  $\lambda_2$ ,
- (b)  $\max\{f_1(x), f_2(x)\}$  and  $\min\{f_1(x), f_2(x)\}$ ,
- (c)  $f_1(x)f_2(x)$ .

The standard form of a DC programming problem is given by

$$\begin{aligned} \min \quad & f_0(x) - g_0(x) \\ \text{s.t.} \quad & f_i(x) - g_i(x) \leq 0, \quad i = 1, \dots, m, \\ & x \in \mathcal{X}, \end{aligned} \tag{52}$$

where  $\mathcal{X} \in \mathbb{R}^n$  is a nonempty closed convex set, and  $f_i$ 's,  $g_i$ 's are all convex in  $\mathcal{X}$ . Recently a sequential-convex-programming based DC algorithm and its variations emerge as an effective approach to solving the problem. The idea of this approach is to create a sequence of values  $\{x^k\}$  by solving convex programs sequentially so that  $\{x^k\}$  converges to a local minimum of (52). Given a convex function  $g$ , a real vector  $v$  is called its *subgradient* at  $x$  if  $v$  satisfies

$$g(y) \geq g(x) + v^T(y - x) \quad \text{for all } y,$$

where  $v^T$  is the transpose of vector  $v$ . Let  $\partial g(x)$  be the set of all the subgradients of function  $g$  at  $x$ . Using this notation, we can present the overarching structure of the method in the following table:

**Table II: A Sequential Convex Programming Method**

- **Step 0.** Choose  $x^0 \in \mathcal{X}$  arbitrarily. Set  $k = 0$ .
- **Step 1.** Compute  $s_{g_i}^k \in \partial g_i(x^k)$  for  $i = 0, 1, \dots, m$ .
- **Step 2.** Solve

$$x^{k+1} \in \arg \min_{y \in \mathcal{C}(x^k, \{s_{g_i}^k\}_{i=1}^m)} \{f_0(y) - [g_0(x^k) + (s_{g_0}^k)^T(y - x^k)]\}$$

with the feasible set  $\mathcal{C}(x^k, \{s_{g_i}^k\}_{i=1}^m)$  being given by

$$\mathcal{C}(x^k, \{s_{g_i}^k\}_{i=1}^m) = \{y \in \mathcal{X} : f_i(y) - [g_i(x^k) + (s_{g_i}^k)^T(y - x^k)] \leq 0, \quad i = 1, \dots, m\}.$$

- **Step 3.** Set  $k \leftarrow k + 1$  and go to Step 1.

Note that in Step 2, we linearize all the convex functions  $g_i$ ,  $i = 1, \dots, m$ , through their subgradients, thereby relaxing the original problem into a tractable convex program. A number of literature shows that the resulted sequence  $\{x^k\}$  converge to a KKT point of (52) under some regularity conditions; see, e.g., see Yuille and Rangarajan (2003), Sriperumbudur and Lanckriet (2009), Lu (2016), and Boyd and Vandenberghe (2004).

## D Convergence of the Monte Carlo DDP Algorithm

### D.1 Review of the algorithm

Let us go through the major steps of the regression-based DDP algorithm proposed in Section 4. It is summarized in the following table.

**Table III: Implementation Details of Regression Based Monte Carlo DDP**

- **Step 0.** Initialization:

- **Step 0a.** Choose a sequence of distribution functions  $(G_1, \dots, G_T)$  and a set of basis functions  $\{\psi_1, \dots, \psi_M\}$ .
- **Step 0b.** Simulate states for each period  $t$  from these distributions:  $x_t^{(l)} \sim G_t$  for  $1 \leq l \leq L$  and  $1 \leq t \leq T$ .
- **Step 0c.** Construct the initial approximation

$$\widehat{\mathfrak{Y}}_t^0(x) := \sum_{m=1}^M \widehat{\beta}_{t,m}^0 \psi_m(x), \quad 1 \leq t \leq T-1.$$

One way to do it is to evaluate the value of a policy  $\alpha$  of being at state  $x_t^{(l)}$  for all  $l$  and  $t$  and use the basis functions to extrapolate these values to the entire state space. See Section 9.1 of

Powell (2011) for the discussion on sampling and approximating the value of a policy.

- **Step 1.** Use the regression method to implement the dual iteration:

- **Step 1a.** Starting with the approximation from the last iteration:

$$\widehat{\mathfrak{Y}}_t^{n-1}(x) := \sum_{m=1}^M \widehat{\beta}_{t,m}^{n-1} \psi_m(x), \quad 1 \leq t \leq T-1,$$

define a penalty function sequence such that  $\mathfrak{z}_T^n(a, \xi) = 0$  and

$$\mathfrak{z}_t^n(a, \xi) = \sum_{s=t}^{T-1} \left\{ \mathbb{E}[r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x_s, a_s, \xi_s))] - (r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x_s, a_s, \xi_s))) \right\}$$

for any  $0 \leq t \leq T-1$ , with  $a = (a_0, \dots, a_{T-1}) \in A$  and  $\xi = (\xi_0, \dots, \xi_{T-1})$ .

- **Step 1b.** At each point  $x_t^{(l)}$ , simulate one path of  $\xi^{(l)}|t = (\xi_t^{(l),t}, \xi_{t+1}^{(l),t}, \dots, \xi_{T-1}^{(l),t})$  independently and solve the optimization program (25-26) for  $\mathfrak{J}_{t,n}^{(l)}$ .
- **Step 1c.** Use the least-square method to fit the data  $(x_t^{(1)}, \mathfrak{J}_{t,n}^{(1)}), \dots, (x_t^{(L)}, \mathfrak{J}_{t,n}^{(L)})$  to obtain a new expansion on the dual:

$$\widehat{\mathfrak{Y}}_t^n(x) := \sum_{m=1}^M \widehat{\beta}_{t,m}^n \psi_m(x), \quad 1 \leq t \leq T-1,$$

where  $\widehat{\beta}_t^n = (\widehat{B}_{\psi\psi}^t)^{-1} \widehat{B}_{\mathfrak{J}\psi}^{t,n}$  whenever  $\widehat{B}_{\psi\psi}^t$  is invertible. Here the  $(i, j)$ -element of matrix  $\widehat{B}_{\psi\psi}^{t,n}$  and the  $k$ -th element of vector  $\widehat{B}_{\mathfrak{J}\psi}^{t,n}$  are defined in (23) and (24), respectively. See the discussion below for the case in which the numerical inversion  $\widehat{B}_{\psi\psi}^{t,n}$  is not stable.

- **Step 1d.** At  $x_0$ , simulate  $L$  independent paths of  $\xi^{(l)}|0 = (\xi_1^{(l),0}, \xi_2^{(l),0}, \dots, \xi_T^{(l),0})$ ,  $1 \leq l \leq L$ , and solve the optimization program (25-26) with  $t = 0$  for  $\mathfrak{J}_{0,n}^{(l)}$ . Let

$$\widehat{\mathfrak{Y}}_0^n(x_0) = \frac{1}{L} \sum_{l=1}^L \mathfrak{J}_{0,n}^{(l)}.$$

- **Step 2.** Let  $n = n + 1$  and go to Step 1.

In the implementation of Step 1c, we find that  $\widehat{B}_{\psi\psi}^{t,n}$  could be nearly singular for some sampled  $(x_{t,n}^{(1)}, \dots, x_{t,n}^{(L)})$ . That will result in numerical instability on  $\widehat{\beta}_t^n$ , and in turn, the final dual output  $\widehat{\mathfrak{Y}}_0^{T+1}$ . To prevent  $\widehat{\mathfrak{Y}}_0^{T+1}$  from being extremely large or small due to the singularity of  $\widehat{B}_{\psi\psi}^{t,n}$ , we truncate the output at a pre-specified sufficiently large  $K$  in the numerical experiments, i.e.,

$$\widehat{\mathfrak{Y}}_0^{T+1}(x) = \max \left\{ -K, \min \left\{ K, \frac{1}{L} \sum_{l=1}^L \mathfrak{J}_{0,T+1}^{(l)} \right\} \right\}.$$

Lemma D.6 provides an upper bound on the probability that the matrix  $\widehat{B}_{\psi\psi}^{t,n}$  is close to singularity. As both  $L$  and  $M$  tend to infinity, the probability of near-singular  $\widehat{B}_{\psi\psi}^{t,n}$  will vanish.

The output of our regression based algorithm  $\{\widehat{\mathfrak{Y}}_t(x), 0 \leq t \leq T\}$  can also be used to simulate for an upper-bound estimate for the true value of the original problem. The key steps are summarized in Table IV. Note that all the policies are suboptimal. It is obvious to see that  $\widehat{\mathfrak{Y}}_0$  will converge to one upper bound for the true value as  $K \rightarrow +\infty$ . Furthermore, we may construct a confidence interval based on  $\widehat{\mathfrak{Y}}_0$  and  $\widehat{\mathfrak{Y}}_0$ . Let  $\underline{\sigma}$  and  $\bar{\sigma}$  be the sample standard deviations of  $\{\mathfrak{J}_0^{(l)}, l = 1, \dots, L\}$  in Step 1d of Table III and  $\{\sum_t r_t(x_t^k, \mathbf{a}_t^k), k = 1, \dots, K\}$  in Step 3 of Table IV. Then, we can form the following interval:

$$\left( \widehat{\mathfrak{Y}}_0 - z_{\delta/2} \frac{\underline{\sigma}}{\sqrt{L}}, \widehat{\mathfrak{Y}}_0 + z_{\delta/2} \frac{\bar{\sigma}}{\sqrt{K}} \right), \quad (53)$$

with  $z_\delta$  being the  $1 - \delta$  quantile of the standard normal distribution. By Theorem 4.5 and Remarks 4.6, 4.7, this interval (53) will provide a valid asymptotic confidence interval for  $V_0$ .

**Table IV: Direct Policy Valuation**

- **Step 0.** Initialization: start from the initial state  $x_0$  and choose a large number  $K$ .
- **Step 1.** Do for  $k = 1, \dots, K$ 
  - **Step 1a.** Set  $t = 0$  and let  $x_t^k = x_0$ .
  - **Step 1b.** At  $x_t^k$ , solve the best action  $\mathbf{a}_t^k$ , given the value function at the next step is approximated by  $\widehat{\mathfrak{V}}_{t+1}$ . That is,

$$\mathbf{a}_t^k = \arg \min_{a_t \in A_t} \mathbb{E} \left[ r_t(x_t^k, a_t, \xi_t) + \widehat{\mathfrak{V}}_{t+1}(f_t(x_t^k, a_t, \xi_t)) \right].$$

- **Step 1c.** Simulate  $\xi_t^k$  and generate the state for the next step through  $x_{t+1}^k = f_t(x_t^k, \mathbf{a}_t^k, \xi_t^k)$ .
  - **Step 1d.** Set  $t \leftarrow t + 1$  and go to Step 1b until  $t = T$ .
- **Step 3.** Compute

$$\widehat{\mathfrak{V}}_0(x_0) := \frac{1}{K} \sum_{k=1}^K \sum_t r_t(x_t^k, \mathbf{a}_t^k).$$

## D.2 One example of exploration pitfall

As noted in Section 4, the state sampler  $G$  is crucial to ensure the convergence of the DDP algorithm. This subsection presents one example to illustrate a possible exploration pitfall if we use a policy-dependent sampler to draw the states on which we estimate the dual values.

Consider the following 2-period SDP problem:

$$\min_u \mathbb{E} \left[ \sum_{t=0}^2 -(x_t - 10)^+ | x_0 = x \right].$$

Here, the control  $u_t$  can only be taken from the set  $\{0, 1, 2\}$  and the dynamic satisfies

$$x_{t+1} = 20 + 10u_t(u_t - 2) - u_t\xi_t = \begin{cases} 20, & u_t = 0 \\ 10 - \xi_t, & u_t = 1 \\ 20 - 2\xi_t, & u_t = 2 \end{cases}.$$

The random noise  $\xi_t$  follows the uniform distribution  $U(0, 10)$ . It is easy to see that the optimal value functions of the problem at  $t = 0, 1, 2$  are given by

$$V_2(x) = -(x - 10)^+, \quad V_1(x) = -10 - (x - 10)^+, \quad \text{and} \quad V_0(x) = -20 - (x - 10)^+,$$

respectively. And the corresponding optimal policy is  $u_t(x) = 0$  for all  $t$  and  $x$ .

Suppose that the set of basis functions we take is

$$\Psi(x) = [\psi_1(x), \psi_2(x), \psi_3(x)] := [1, x, (10 - x)^+].$$

And we are given by an initial policy  $u_t = 1$  for all  $t = 0, 1, 2$ ; that is, the policy always selects the action of 1 no matter what state and period the planner is at. Instead of using an independent state sampler as suggested in Step0a of Table III, let us consider the situation that we rely on such  $u$  to drive the system to obtain the states that we may estimate the dual values later. Denote them by  $(x_t^{(1)}, \dots, x_t^{(L)})$ ,  $t = 0, 1, 2$ . Note that all of them are in  $(0, 10)$ .

Evaluating the value of this policy on these states, we know that all the values are  $\widehat{\mathfrak{V}}_t^0(x_t^{(l)}) = 0$ . If we use the regression technique to extrapolate these values to the entire state space, we need to solve

$$\inf_{\beta} \frac{1}{L} \sum_{l=1}^L \left( \Psi(x_t^l) \beta - \widehat{\mathfrak{V}}_t^0(x_t^l) \right)^2 \quad (54)$$

for the regression coefficients  $\hat{\beta}_t^0$ . It is easy to see that this is an underdetermined problem in the sense that infinitely many  $\beta$  are the minimizer of the term on the right hand side of (54).

Take one solution  $\hat{\beta}_{t,i}^0 = 0$  with  $i = 1, 2, 3$  for all  $t$ ; that is, we extrapolate  $\widehat{\mathfrak{V}}_t^0(x) = 0$  to the entire space as the approximate value used in Step 1a of Table III. Substitute it into the expression of the penalty. Following Step 1b in Table III, we solve the inner optimization problem (25-26) at  $x_t^{(l)} \in (0, 10)$  and obtain

$$\mathfrak{J}_{0,1}^{(l)} = -20, \quad \mathfrak{J}_{1,1}^{(l)} = -10, \quad \mathfrak{J}_{2,1}^{(l)} = 0. \quad (55)$$

Note that all these values have nothing to do with the random noise  $\xi$ . Use  $\mathfrak{J}_{1,1}^{(l)}$  as an example to explain how the above is calculated. As  $\widehat{\mathfrak{V}}_t^0(x) = 0$  for all  $t$ , the penalty function  $\mathfrak{J}_t^1(u, \xi)$  should also be zero. Then

$$\begin{aligned} \mathfrak{J}_{1,1}^{(l)}(x_1^{(l)}, \xi_1^{(l),1}) &= \inf_{u_1} \{ -(x_1^{(l)} - 10)^+ - (x_2^{(l)} - 10)^+ \} \\ &= \inf_{u_1} \{ -(x_1^{(l)} - 10)^+ - (20 + 10u_1(u_1 - 2) - u_1\xi_1^{(l),1} - 10)^+ \}. \end{aligned}$$

Apparently,  $u_1 = 0$  is the optimal solution to this inner optimization problem. We thus have

$$\mathfrak{J}_{1,1}^{(l)}(x_1^{(l)}, \xi_1^{(l),1}) = (x_1^{(l)} - 10)^+ - 10 = -10$$

because  $x_1^{(l)} \in (0, 10)$ .

Under (55), after we fit these values using the basis functions according to the Step 1c in Table III, we know that

$$\hat{\beta}_0^1 = [-20, 0, 0], \quad \hat{\beta}_1^1 = [-10, 0, 0], \quad \hat{\beta}_2^1 = [0, 0, 0].$$



That is,

$$\widehat{\mathfrak{V}}_0^1(x) = -20, \quad \widehat{\mathfrak{V}}_1^1(x) = -10, \quad \widehat{\mathfrak{V}}_2^1(x) = 0$$

for all  $x$ . Repeat the calculation for more rounds of dual operation and we find that the dual value will not change, i.e.,  $\widehat{\mathfrak{V}}_t^n(x) = \widehat{\mathfrak{V}}_t^1(x)$  for all  $x$  and  $t = 0, 1, 2$ . No convergence to the optimal value function will occur.

The above example shows that using control policies to generate the representative states may lead our DDP algorithm to be stuck in a suboptimal solution. The cause is that all the sampled states we select at the beginning are in  $(0, 10)$  and no one falls in  $(10, 20)$ , the other part of the state space. Lacking the related information in  $(10, 20)$ , the extrapolation from the regression cannot produce correct estimation for the value in that interval.

### D.3 Proof of Theorem 4.5

Now we turn to prove Theorem 4.5. Below we will use  $C$  to represent a generic constant, which is independent of  $M$  and  $L$ . Note that it may change step by step. In the theorem statement, we also use the following concept of Lebesgue constant. Consider a sequence of basis functions  $\{\psi_m(x), m \geq 1\}$ . Given a function  $f$  such that  $\|f\|_\infty \neq 0$  and  $\|f\|_\infty < \infty$ , we use the standard least square method to find a proper expansion of  $\{\psi_m(x), m \geq 1\}$  to approximate  $f$ ; that is, let

$$\widehat{\beta}_f = \arg \min_{\alpha} \mathbb{E}^G[\|f(x) - \Psi_M^{tr}(x)\alpha\|^2]$$

and then  $f \approx \Psi_M^{tr}\widehat{\beta}_f$ .

**Definition D.1 (Lebesgue constant)** *Define*

$$l_M = \sup \left\{ \frac{\|\Psi_M^{tr}(x)\widehat{\beta}_f\|_\infty}{\|f\|_\infty} : \|f\|_\infty \neq 0, \|f\|_\infty < \infty \right\}. \quad (56)$$

#### D.3.1 Technical Lemmas

We need to establish several lemmas first.

**Lemma D.2** *For any function  $f(x)$  and  $g(x)$ ,*

$$\inf_x (f(x) + g(x)) \geq \inf_x f(x) + \inf_x g(x)$$

and

$$\inf_x f(x) - \inf_x g(x) \geq \inf_x (f(x) - g(x)).$$

**Lemma D.3** *For any  $x, y \in \mathbb{R}$ , let constant*

$$K \geq |y|.$$

*Then we have*

$$\left| \max \left\{ -K, \min \{K, x\} \right\} - y \right| \leq |x - y|.$$

*Proof of Lemma D.3.* It can be easily verified .  $\square$

From Assumption 4.1 and 4.2, we can establish the non-multicollinearity of basis functions as shown in the following lemma.

**Lemma D.4** *Under Assumption 4.1 and 4.2, the smallest eigenvalue of matrix  $B_{\psi\psi}^t$  is bounded away from zero uniformly in  $M$ .*

*Proof of Lemma D.4.* Note that  $B_{\psi\psi}^t = \mathbb{E}^G[\Psi(X_t)\Psi^{tr}(X_t)]$  is a nonnegative definite matrix. Thus, its smallest eigenvalue satisfies

$$\lambda_{\min}(B_{\psi\psi}^t) = \min_{\|w\|_2=1} w^{tr} \mathbb{E}^G[\Psi(X_t)\Psi^{tr}(X_t)]w. \quad (57)$$

Moreover, by Assumption 4.2, there exists an  $\epsilon > 0$  such that  $dG/dF(x) > \epsilon$  for  $x \in \mathcal{X}$ . We have

$$\mathbb{E}^G[\Psi(X_t)\Psi^{tr}(X_t)] = \int_{\mathbb{R}^n} \Psi(x)\Psi^{tr}(x) \frac{dG}{dF}(x) dF(x) \geq \epsilon \int_{\mathbb{R}^n} \Psi(x)\Psi^{tr}(x) dF(x). \quad (58)$$

The orthogonality of the basis functions in Assumption 4.1 implies that the right hand side of the above inequality is given by  $\epsilon \cdot I$ , where  $I$  is an identity matrix. For any vector  $w \in \mathbb{R}^n$ , the inequality (58) implies that

$$w^{tr} \mathbb{E}^G[\Psi(X_t)\Psi^{tr}(X_t)]w \geq \epsilon w^{tr} w.$$

In conjunction with (57), we have

$$\lambda_{\min}(B_{\psi\psi}^t) \geq \epsilon. \square$$

Consider one sequence of i.i.d. random vectors  $X_1, \dots, X_L \in \mathbb{R}^d$  and another sequence of i.i.d. random variables  $Y_1, \dots, Y_L \in \mathbb{R}$ . Suppose that all of  $(X_i)_{1 \leq i \leq L}$  are square integrable and their second moments are bounded above by a constant. Furthermore,  $(Y_i)_{1 \leq i \leq L}$  is assumed to be essentially bounded, i.e., there exists  $\|Y\|_\infty$  such that

$$\max_{1 \leq i \leq L} |Y_i| \leq \|Y\|_\infty.$$

Then, we have

**Lemma D.5** *There exists a constant  $C$ , independent of  $L$  and  $d$ , such that*

$$\mathbb{E} \left[ \left\| \frac{1}{L} \sum_{l=1}^L X_l Y_l - \mathbb{E}[X_l Y_l] \right\|_2 \right] \leq \frac{C\sqrt{d}}{\sqrt{L}} \|Y\|_\infty.$$

*Proof of Lemma D.5.* Let  $X_l^k$  denote the  $k$ -th element of vector  $X_l$ . According to Jensen's inequality,

$$\begin{aligned} \mathbb{E} \left[ \left\| \frac{1}{L} \sum_{l=1}^L X_l Y_l - \mathbb{E}[X_l Y_l] \right\|_2 \right] &= \mathbb{E} \left[ \left\{ \sum_{k=1}^d \left( \frac{1}{L} \sum_{l=1}^L X_l^k Y_l - \mathbb{E}[X_l^k Y_l] \right)^2 \right\}^{1/2} \right] \\ &\leq \left\{ \mathbb{E} \left[ \sum_{k=1}^d \left( \frac{1}{L} \sum_{l=1}^L X_l^k Y_l - \mathbb{E}[X_l^k Y_l] \right)^2 \right] \right\}^{1/2}. \end{aligned}$$

Observe that

$$\mathbb{E}\left[\left(\frac{1}{L}\sum_{l=1}^L X_l^k Y_l - \mathbb{E}[X_l^k Y_l]\right)^2\right] = \frac{1}{L^2}\sum_{l=1}^L \mathbb{E}\left[\left(X_l^k Y_l - \mathbb{E}[X_l^k Y_l]\right)^2\right].$$

Each summand on the right hand side of above equality satisfies

$$\mathbb{E}\left[\left(X_l^k Y_l - \mathbb{E}[X_l^k Y_l]\right)^2\right] \leq \mathbb{E}\left[\left(X_l^k Y_l\right)^2\right] \leq C\|Y\|_\infty^2,$$

if we take

$$C = \max_{1 \leq k \leq d} E[(X_l^k)^2].$$

Accordingly, we have

$$\mathbb{E}\left[\left\|\frac{1}{L}\sum_{l=1}^L X_l Y_l - \mathbb{E}[X_l Y_l]\right\|_2\right] \leq \frac{C\sqrt{d}}{\sqrt{L}}\|Y\|_\infty. \square$$

The next lemma gives bound on the probability that the sample matrix  $\widehat{B}_{\psi\psi}^{t,n}$  deviates from its mean  $B_{\psi\psi}^t$ . More precisely, given a  $\delta > 0$ , for any time  $t$  and iteration  $n$ , let

$$A_t^n(\delta) = \{\|I - (B_{\psi\psi}^t)^{-1}\widehat{B}_{\psi\psi}^{t,n}\|_2 \geq \delta\},$$

where  $I$  is the identity matrix. We have

**Lemma D.6** *There exists a constant  $C$ , independent of  $M$  and  $L$ , such that for any  $\delta$ ,*

$$\mathbb{P}(A_t^n(\delta)) \leq 2M \exp\left\{-\frac{L\delta^2}{CM^2}\right\}.$$

*Proof of Lemma D.6.* It is Lemma 2.1 in Chen and Christensen (2015). This inequality is also known as the matrix Bernstein inequality in the literature; see also Tropp (2012).  $\square$

In the following lemma we develop an upper bound estimate on the distance between sample value  $\mathfrak{J}_{t,n}(\xi|t, x)$  and the optimal value  $V_t(x)$ . To be more precisely,

**Lemma D.7** *Given the initial state  $x_t = x$  and the truncated randomness sequence  $\xi|t = (\xi_t, \dots, \xi_{T-1})$ , the corresponding optimization problem  $\mathfrak{J}_{t,n}(\xi|t, x)$ , satisfies*

$$\left\|\mathfrak{J}_{t,n}(\xi|t, x) - V_t(x)\right\|_\infty \leq 2 \sum_{s=t+1}^T \left\|\widehat{\mathfrak{V}}_s^{n-1}(x) - V_s(x)\right\|_\infty.$$

*Proof of Lemma D.7.* Recall that  $\mathfrak{J}_{t,n}(\xi|t, x)$  is defined by

$$\mathfrak{J}_{t,n}(\xi|t, x) = \inf_{a \in A|t} \left( \sum_{s=t}^{T-1} r_s(x_s, a_s, \xi_s) + r_T(x_T) + \mathfrak{z}_t^n(a, \xi) \right)$$

with

$$\mathfrak{J}_t^n(a, \xi) = \sum_{s=t}^{T-1} \left\{ \mathbb{E} \left[ r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{V}}_{s+1}^{n-1}(x_{s+1}) \right] - \left( r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{V}}_{s+1}^{n-1}(x_{s+1}) \right) \right\}.$$

Following similar arguments as the proof of Lemma A.1, we can show that  $\mathfrak{J}_{t,n}$  admits the following recursive representation:

$$\mathfrak{J}_{t,n}(\xi|t, x) = \inf_{a \in A_t} \left( \mathbb{E} \left[ r_t(x_t, a, \xi_t) + \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) \right] - \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) + \mathfrak{J}_{t+1,n}(\xi|t+1, f(x, a, \xi_t)) \right).$$

By Lemma D.2, we know that

$$\begin{aligned} & \mathfrak{J}_{t,n}(\xi|t, x) \\ & \geq \inf_{a \in A_t} \mathbb{E} \left[ r_t(x_t, a, \xi_t) + \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) \right] + \inf_{a \in A_t} \left\{ \mathfrak{J}_{t+1,n}(\xi|t+1, f(x, a, \xi_t)) - \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) \right\}, \\ & =: J_1 + J_2. \end{aligned} \tag{59}$$

Consider the part of  $J_1$  on the right hand side of (59). Note that  $V_t(x)$  satisfies the Bellman equation,

$$V_t(x) = \inf_{a \in A_t} \mathbb{E} \left[ r_t(x, a, \xi_t) + V_{t+1}(f_t(x, a, \xi_t)) \right]. \tag{60}$$

Then, the difference between  $J_1$  and  $V_t(x)$  should be

$$\begin{aligned} & J_1 - V_t(x) \\ & = \inf_{a \in A_t} \mathbb{E} \left[ r_t(x_t, a, \xi_t) + \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) \right] - \inf_{a \in A_t} \mathbb{E} \left[ r_t(x, a, \xi_t) + V_{t+1}(f_t(x, a, \xi_t)) \right] \\ & \geq \inf_{a \in A_t} \mathbb{E} \left[ \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) - V_{t+1}(f_t(x, a, \xi_t)) \right], \end{aligned}$$

where the inequality in the last line is because of Lemma D.2. Furthermore, since  $f_t(x, a, \xi_t) \in \mathcal{X}$  for any state  $x$ , action  $a$  and random noise  $\xi_t$ , we have

$$\widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) - V_{t+1}(f_t(x, a, \xi_t)) \geq - \sup_{x \in \mathcal{X}} \left| \widehat{\mathfrak{V}}_{t+1}^{n-1}(x) - V_{t+1}(x) \right| = - \left\| \widehat{\mathfrak{V}}_{t+1}^{n-1}(x) - V_{t+1}(x) \right\|_{\infty}.$$

Taking expectation with respect to  $\xi_t$  and taking infimum over all possible actions  $a \in A_t$  on both side of above inequality will lead to

$$\inf_{a \in A_t} \mathbb{E} \left[ \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) - V_{t+1}(f_t(x, a, \xi_t)) \right] \geq - \left\| \widehat{\mathfrak{V}}_{t+1}^{n-1}(x) - V_{t+1}(x) \right\|_{\infty}.$$

That implies,

$$J_1 - V_t(x) \geq - \left\| \widehat{\mathfrak{V}}_{t+1}^{n-1}(x) - V_{t+1}(x) \right\|_{\infty}. \tag{61}$$

Next we turn to  $J_2$ , the second part on the right hand of (59). Substitute the definition of  $\mathfrak{J}_{t+1,n}(\xi|t+1, f(x, a, \xi_t))$  into  $J_2$ . After some term rearrangements, we obtain

$$\begin{aligned} J_2 & = \inf_{a \in A_t} \left\{ \mathfrak{J}_{t+1,n}(\xi|t+1, f(x, a, \xi_t)) - \widehat{\mathfrak{V}}_{t+1}^{n-1}(f_t(x, a, \xi_t)) \right\} \\ & = \inf_{a \in A_t} \left\{ \sum_{s=t+1}^{T-1} \left( \mathbb{E} \left[ r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{V}}_{s+1}^{n-1}(x_{s+1}) \right] - \widehat{\mathfrak{V}}_s^{n-1}(x_s) \right) + \left( r_T(x_T) - \widehat{\mathfrak{V}}_T^{n-1}(x_T) \right) \right\}. \end{aligned} \tag{62}$$

Applying Lemma D.2 to the first term on the right hand side of the above equality,

$$\begin{aligned}
& \inf_{a \in A|t} \left\{ \sum_{s=t+1}^{T-1} \left( \mathbb{E} \left[ r_s(x_s, a_s, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(x_{s+1}) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x_s) \right) \right\} \\
& \geq \sum_{s=t+1}^{T-1} \inf_{a \in A|t} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x_s, a, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x_s, a, \xi_s)) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x_s) \right\} \\
& \geq \sum_{s=t+1}^{T-1} \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right\}. \tag{63}
\end{aligned}$$

Here the last inequality is obvious because every summand of the sum in the second line, as a function of state variable  $x$ , is bounded below by its minimum over the space  $\mathcal{X}$ . Similarly, we have

$$r_T(x_T) - \widehat{\mathfrak{Y}}_T^{n-1}(x_T) \geq \inf_{x \in \mathcal{X}} \left\{ r_T(x) - \widehat{\mathfrak{Y}}_T^{n-1}(x) \right\}. \tag{64}$$

From (62-64),

$$J_2 \geq \sum_{s=t+1}^{T-1} \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right\} + \inf_{x \in \mathcal{X}} \left\{ r_T(x) - \widehat{\mathfrak{Y}}_T^{n-1}(x) \right\}. \tag{65}$$

We add and subtract the optimal value function  $V$  simultaneously in every summand of the sum on the right hand side of (65). This operation will not change its value. That is,

$$\begin{aligned}
& \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right\} \\
& = \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + V_{s+1}(f_s(x, a, \xi_s)) + \left( \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) - V_{s+1}(f_s(x, a, \xi_s)) \right) \right] \right. \\
& \quad \left. - V_s(x) + \left( V_s(x) - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right) \right\} \\
& \geq \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + V_{s+1}(f_s(x, a, \xi_s)) \right] - V_s(x) \right\} + \inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \left( \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) - V_{s+1}(f_s(x, a, \xi_s)) \right) \right\} \\
& \quad + \inf_{x \in \mathcal{X}} \left( V_s(x) - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right), \tag{66}
\end{aligned}$$

where we use Lemma D.2 again to obtain the last inequality. Thanks to the Bellman equation, we know that the first term on the right hand side of the inequality (66) is 0. In addition, following similar arguments leading to (61), we can establish

$$\inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \left( \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) - V_{s+1}(f_s(x, a, \xi_s)) \right) \right\} \geq - \left\| \widehat{\mathfrak{Y}}_{s+1}^{n-1}(x) - V_{s+1}(x) \right\|_{\infty}$$

and

$$\inf_{x \in \mathcal{X}} \left( V_s(x) - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right) \geq - \left\| \widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x) \right\|_{\infty}.$$

As a consequence, we have

$$\inf_{x \in \mathcal{X}} \left\{ \inf_{a \in A_s} \mathbb{E} \left[ r_s(x, a, \xi_s) + \widehat{\mathfrak{Y}}_{s+1}^{n-1}(f_s(x, a, \xi_s)) \right] - \widehat{\mathfrak{Y}}_s^{n-1}(x) \right\} \geq - \left\| \widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x) \right\|_{\infty} - \left\| \widehat{\mathfrak{Y}}_{s+1}^{n-1}(x) - V_{s+1}(x) \right\|_{\infty}.$$

Summing the above inequality over  $s = t + 1$  to  $T - 1$ , (65) implies that

$$J_2 \geq -2 \sum_{s=t+2}^T \|\widehat{\underline{\mathfrak{Y}}}_s^{n-1}(x) - V_s(x)\|_\infty - \|\widehat{\underline{\mathfrak{Y}}}_{t+1}^{n-1}(x) - V_{t+1}(x)\|_\infty. \quad (67)$$

Hence,

$$\mathfrak{J}_{t,n}(\xi|t, x) - V_t(x) = J_1 - V_t(x) + J_2 \geq -2 \sum_{s=t+1}^T \|\widehat{\underline{\mathfrak{Y}}}_s^{n-1}(x) - V_s(x)\|_\infty.$$

To finish the proof, we need to derive the upper bound for  $\mathfrak{J}_{t,n}(\xi|t, x) - V_t(x)$ . By Assumption 4.4, let  $a^*(x)$  be the optimal solution to the Bellman equation; that is,

$$a_t^* = \arg \inf_{a \in A_t} \mathbb{E} \left[ r_t(x_t, a, \xi_t) + V_{t+1}(x_{t+1}) \mid x_t = x \right].$$

Such a policy  $a_t^*(x)$  must be a suboptimal solution to the optimization in the definition of  $\mathfrak{J}_{t,n}(\xi|t, x)$ . Therefore,

$$\mathfrak{J}_{t,n}(\xi|t, x) \leq \sum_{s=t}^{T-1} \left\{ \mathbb{E} \left[ r_s(x_s^*, a_s^*(x_s^*), \xi_s) + \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) \right] - \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) \right\} + r_T(x_T^*), \quad (68)$$

with  $x_{t+1}^* = f_t(x_t^*, a_t^*(x_t^*), \xi_t)$ . On the other hand, we may rewrite  $V_t(x_t^*)$  using the following telescoping sum:

$$V_t(x_t^*) = \sum_{s=t}^{T-1} \left\{ V_s(x_s^*) - V_{s+1}(x_{s+1}^*) \right\} + r_T(x_T^*).$$

Note that  $V_T(\cdot) \equiv r_T(\cdot)$ . By the Bellman equation, for all  $s = t, \dots, T_1$  and  $x_s^*$ ,

$$V_s(x_s^*) = \mathbb{E} \left[ r_s(x_s^*, a_s^*(x_s^*), \xi_s) + V_{s+1}(x_{s+1}^*) \right].$$

Therefore,

$$V_t(x_t^*) = \sum_{s=t}^{T-1} \left\{ \mathbb{E} \left[ r_s(x_s^*, a_s^*(x_s^*), \xi_s) + V_{s+1}(x_{s+1}^*) \right] - V_{s+1}(x_{s+1}^*) \right\} + r_T(x_T^*). \quad (69)$$

Subtract the above two relation (68) and (69),

$$\begin{aligned} & \mathfrak{J}_{t,n}(\xi|t, x_t^*) - V_t(x_t^*) \\ & \leq \sum_{s=t}^{T-1} \left\{ \mathbb{E} \left[ \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) - V_{s+1}(x_{s+1}^*) \right] + V_{s+1}(x_{s+1}^*) - \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) \right\}. \end{aligned}$$

Following the similar procedures leading to (61), for  $t \leq s \leq T - 1$ , we can show that

$$\mathbb{E} \left[ \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) - V_{s+1}(x_{s+1}^*) \right] \leq \|\widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x) - V_{s+1}(x)\|_\infty$$

and

$$V_{s+1}(x_{s+1}^*) - \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x_{s+1}^*) \leq \|V_{s+1}(x) - \widehat{\underline{\mathfrak{Y}}}_{s+1}^{n-1}(x)\|_\infty.$$

Consequently, we have

$$\mathfrak{J}_{t,n}(\xi|t, x) - V_t(x) \leq 2 \sum_{s=t+1}^T \left\| \widehat{\underline{\mathfrak{Y}}}_s^{n-1}(x) - V_s(x) \right\|_\infty.$$

In summary, the combination of these two bounds implies

$$\left\| \mathfrak{J}_{t,n}(\xi|t, x) - V_t(x) \right\|_\infty \leq 2 \sum_{s=t+1}^T \left\| \widehat{\underline{\mathfrak{Y}}}_s^{n-1}(x) - V_s(x) \right\|_\infty. \square$$

From the Lemma D.7, it turns out that

**Corollary D.8** *Let*

$$\underline{\mathfrak{Y}}_t^n(x) = \mathbb{E}[\mathfrak{J}_{t,n}(\xi|t, x)].$$

*Then we have*

$$\|\underline{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty \leq 2 \sum_{s=t+1}^T \|\widehat{\underline{\mathfrak{Y}}}_s^{n-1}(x) - V_s(x)\|_\infty. \quad (70)$$

*Proof of Corollary D.8.* This can be easily verified by Jensen's inequality.  $\square$

In the next lemma, we attempt to bound the sampling error

$$\|\Psi_M^{tr}(x)\widehat{\beta}_t^n - \Psi_M^{tr}(x)\beta_t^n\|_\infty$$

when  $\widehat{B}_{\psi\psi}^{t,k}$  gives a “good” approximation to  $B_{\psi\psi}^t$ . To be precise, define event  $A(\delta, n)$  to be

$$A(\delta, n) = \bigcup_{\substack{1 \leq k \leq n, \\ T-k+1 \leq t \leq T}} A_t^k(\delta) = \bigcup_{\substack{1 \leq k \leq n, \\ T-k+1 \leq t \leq T}} \{\|I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,k}\|_2 \geq \delta\}.$$

**Lemma D.9** *Let*

$$\delta = \frac{1}{2M^{1/2}L^{1/4}}$$

*in the above definition of  $A(\delta, n)$ . There exists a constant  $C$ , independent of  $M$  and  $L$ , such that for  $1 \leq n \leq T$  and  $T - n + 1 \leq t \leq T$ ,*

$$\mathbb{E} \left[ \mathbf{1}_{A(\delta, n)^c} \cdot \left\| \Psi_M^{tr}(x)(\widehat{\beta}_t^n - \beta_t) \right\|_\infty \right] \leq C \left( \frac{M^{3/2}}{L^{1/4}} \right) \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_\infty \cdot \mathbf{1}_{A(\delta, n-1)^c} \right].$$

*Proof of Lemma D.9.* Recall that if  $A$  is a real symmetric matrix, then all the eigenvalues of this matrix is real. In this proof, we use  $\lambda_{\max}(A)$  and  $\lambda_{\min}(A)$  to denote the maximum and minimum eigenvalue for a general symmetric  $A$ . By the definitions of  $\widehat{\beta}_t^n$  and  $\beta_t^n$ , we have

$$\widehat{\beta}_t^n = (\widehat{B}_{\psi\psi}^{t,n})^{-1} \cdot \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \quad \text{and} \quad \beta_t^n = (B_{\psi\psi}^t)^{-1} \mathbb{E}[\Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}].$$

Hence,

$$\Psi_M^{tr}(x)(\widehat{\beta}_t^n - \beta_t^n) = \Psi_M^{tr}(x)(\widehat{B}_{\psi\psi}^{t,n})^{-1} \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} - \Psi_M^{tr}(x)(B_{\psi\psi}^t)^{-1} \mathbb{E}[\Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}].$$

We simultaneously add and subtract

$$\Psi_M^{tr}(x)(B_{\psi\psi}^t)^{-1} \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}$$

on the right hand side of the above equality. That results in

$$\begin{aligned} & \Psi_M^{tr}(x)(\widehat{\beta}_t^n - \beta_t^n) \\ &= \Psi_M^{tr}(x) \left[ (\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1} \right] \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} + \Psi_M^{tr}(x)(B_{\psi\psi}^t)^{-1} \left\{ \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} - \mathbb{E}[\Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}] \right\}. \end{aligned}$$

Then the triangle inequality implies

$$\|\Psi_M^{tr}(x)(\widehat{\beta}_t^n - \beta_t^n)\|_{\infty} \leq \epsilon_{t,n}^{(1)} + \epsilon_{t,n}^{(2)},$$

where  $\epsilon_{t,n}^{(1)}$  and  $\epsilon_{t,n}^{(2)}$  are defined as

$$\epsilon_{t,n}^{(1)} = \sup_{x \in \mathcal{X}} \left| \Psi_M^{tr}(x) \left[ (\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1} \right] \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \right|$$

and

$$\epsilon_{t,n}^{(2)} = \sup_{x \in \mathcal{X}} \left| \Psi_M^{tr}(x)(B_{\psi\psi}^t)^{-1} \left\{ \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} - \mathbb{E}[\Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}] \right\} \right|.$$

By the Cauchy-Schwartz inequality, it is easy to see that  $\epsilon_{t,n}^{(1)}$  is bounded by

$$\begin{aligned} \epsilon_{t,n}^{(1)} &\leq \frac{1}{L} \sum_{l=1}^L \sup_{x \in \mathcal{X}} \left| \Psi_M^{tr}(x) \left[ (\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1} \right] \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \right| \\ &\leq \frac{1}{L} \sum_{l=1}^L \sup_{x \in \mathcal{X}} \left\| \Psi_M^{tr}(x) \right\|_2 \cdot \left\| (\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1} \right\|_2 \cdot \left\| \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \right\|_2. \end{aligned} \quad (71)$$

Under Assumption 4.3, there exists a constant  $C$  such that

$$\sup_{x \in \mathcal{X}} \left\| \Psi_M^{tr}(x) \right\|_2 \leq CM.$$



We next develop an upper bound for the last term on the right hand side of (71). Note that

$$\left\| \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \right\|_2 = \left[ \sum_{m=1}^M (\psi_m(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)})^2 \right]^{\frac{1}{2}} \leq \left[ \sum_{m=1}^M (\psi_m(x_{t,n}^{(l)})^2)^{\frac{1}{2}} \right]^{\frac{1}{2}} \cdot \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty}.$$

By Assumption 4.3,

$$\left[ \sum_{m=1}^M (\psi_m(x_{t,n}^{(l)}))^2 \right]^{\frac{1}{2}} \leq CM.$$

Hence,

$$\left\| \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} \right\|_2 \leq CM \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty}. \quad (72)$$

To bound

$$\|(\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1}\|_2,$$

by Cauchy-Schwarz inequality,

$$\begin{aligned} \left\| (\widehat{B}_{\psi\psi}^{t,n})^{-1} - (B_{\psi\psi}^t)^{-1} \right\|_2 &= \left\| (I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n}) \left( (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right)^{-1} (B_{\psi\psi}^t)^{-1} \right\|_2 \\ &\leq \left\| I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right\|_2 \cdot \left\| \left( (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right)^{-1} \right\|_2 \cdot \left\| (B_{\psi\psi}^t)^{-1} \right\|_2. \end{aligned}$$

From the definition of  $A(\delta, n)$ , we know that

$$\left\| I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right\|_2 \leq (2M^{1/2}L^{1/4})^{-1} \quad (73)$$

on the set of  $A(\delta, n)^c$ . Using Example 5.6.6 in Horn and Johnson (2003),

$$\left\| (B_{\psi\psi}^t)^{-1} \right\|_2 = \lambda_{\max} \left( (B_{\psi\psi}^t)^{-1} \right) = \frac{1}{\lambda_{\min}(B_{\psi\psi}^t)}.$$

As for

$$\left\| \left( (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right)^{-1} \right\|_2,$$

it is well known that

$$\lambda_{\min} \left( (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right) = \min_{\|w\|=1} w^{tr} (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} w; \quad (74)$$

see Theorem 4.2.2 in Horn and Johnson (2003). For any vector  $w$  with  $\|w\| = 1$ ,

$$w^{tr} (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} w = w^{tr} I w + w^{tr} \left( (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} - I \right) w = 1 - w^{tr} \left( I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right) w,$$

where  $I$  is an identity matrix. Hence,

$$\min_{\|w\|=1} w^{tr} (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} w = 1 - \max_{\|w\|=1} w^{tr} \left( I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n} \right) w. \quad (75)$$

On the other hand, it is easy to show that

$$\max_{\|w\|=1} w^{tr} (I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n}) w = \|I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n}\|_2. \quad (76)$$

Combining (73-76) yields

$$1_{A(\delta,n)^c} \cdot \lambda_{\min}((B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n}) \geq 1_{A(\delta,n)^c} \cdot (1 - (2M^{1/2}L^{1/4})^{-1}) \geq \frac{1}{2} \cdot 1_{A(\delta,n)^c}. \quad (77)$$

where the last inequality is due to the fact that  $M, L \geq 1$ . Thus,

$$1_{A(\delta,n)^c} \cdot \left\| \left( \widehat{B}_{\psi\psi}^{t,n} \right)^{-1} - \left( B_{\psi\psi}^t \right)^{-1} \right\|_2 \leq 1_{A(\delta,n)^c} \cdot \frac{1}{\lambda_{\min}(B_{\psi\psi}^t) M^{1/2} L^{1/4}}. \quad (78)$$

By (71), (72), and (78), we have

$$\mathbb{E} \left[ \epsilon_{t,n}^{(1)} \cdot 1_{A(\delta,n)^c} \right] \leq \frac{CM^{3/2}}{\lambda_{\min}(B_{\psi\psi}^t) L^{1/4}} \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty} \cdot 1_{A(\delta,n)^c} \right].$$

From Lemma D.4,  $\lambda_{\min}(B_{\psi\psi}^t)$  is bounded below by some constant. Therefore, the right hand side of the above inequality is further bounded by

$$C \frac{M^{3/2}}{L^{1/4}} \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty} \cdot 1_{A(\delta,n-1)^c} \right],$$

if we change the constant properly.

Using Cauchy-Schwartz inequality again,  $\epsilon_{t,n}^{(2)}$  satisfies

$$\epsilon_{t,n}^{(2)} \leq \sup_{x \in \mathcal{X}} \left\| \Psi_M^{tr}(x) \right\|_2 \cdot \left\| (B_{\psi\psi}^t)^{-1} \right\|_2 \cdot \left\| \frac{1}{L} \sum_{l=1}^L \Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)} - \mathbb{E}[\Psi_M(x_{t,n}^{(l)}) \mathfrak{J}_{t,n}^{(l)}] \right\|_2.$$

Let  $D_{t,n}$  denote the last term of right hand side in above inequality. Note that

$$\sup_{x \in \mathcal{X}} \left\| \Psi_M^{tr}(x) \right\|_2 \leq CM$$

by Assumption 4.3 and

$$\left\| (B_{\psi\psi}^t)^{-1} \right\|_2 = \lambda_{\min}^{-1}(B_{\psi\psi}^t).$$

We have

$$\epsilon_{t,n}^{(2)} \leq \frac{CM}{\lambda_{\min}(B_{\psi\psi}^t)} \cdot D_{t,n}.$$

The definition of  $A(\delta, n)$  implies that

$$A(\delta, n)^c \subseteq A(\delta, n-1)^c.$$

Therefore,

$$\mathbb{E} \left[ \epsilon_{t,n}^{(2)} \cdot 1_{A(\delta,n)^c} \right] \leq \mathbb{E} \left[ \epsilon_{t,n}^{(2)} \cdot 1_{A(\delta,n-1)^c} \right] \leq \frac{CM}{\lambda_{\min}(B_{\psi\psi}^t)} \cdot \mathbb{E} \left[ D_{t,n} \cdot 1_{A(\delta,n-1)^c} \right]. \quad (79)$$

Let  $\mathcal{G}_n$  be a  $\sigma$ -algebra defined as follows,

$$\mathcal{G}_n = \sigma \left( \left\{ (x_{t,k}^{(1)}, \dots, x_{t,k}^{(L)}), (\xi^{(l)}|t) \right\}, 1 \leq k \leq n \right).$$

By the iterated law of conditional expectation, the expectation term on the right hand side of (79) equals

$$\mathbb{E} \left[ D_{t,n} \cdot 1_{A(\delta, n-1)^c} \right] = \mathbb{E} \left[ \mathbb{E} \left[ D_{t,n} \cdot 1_{A(\delta, n-1)^c} \middle| \mathcal{G}_{n-1} \right] \right]. \quad (80)$$

Since the event  $A(\delta, n-1)^c$  is measurable with respect to  $\mathcal{G}_{n-1}$ , we have

$$\mathbb{E} \left[ D_{t,n} \cdot 1_{A(\delta, n-1)^c} \right] = \mathbb{E} \left[ \mathbb{E} \left[ D_{t,n} \middle| \mathcal{G}_{n-1} \right] \cdot 1_{A(\delta, n-1)^c} \right]. \quad (81)$$

Following the proof of Lemma D.5, we can show

$$\mathbb{E} \left[ D_{t,n} \middle| \mathcal{G}_{n-1} \right] \leq \frac{C\sqrt{M}}{\sqrt{L}} \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty}. \quad (82)$$

In light of (79-82),

$$\mathbb{E} \left[ \epsilon_{t,n}^{(2)} \cdot 1_{A(\delta, n)^c} \right] \leq C \frac{M^{3/2}}{\lambda_{\min}(B_{\psi\psi}^t) \sqrt{L}} \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty} \cdot 1_{A(\delta, n-1)^c} \right].$$

Using again the fact that  $\lambda_{\min}(B_{\psi\psi}^t)$  is bounded below, the right hand side of above can be bounded by

$$C \frac{M^{3/2}}{\sqrt{L}} \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty} \cdot 1_{A(\delta, n-1)^c} \right],$$

by changing constant  $C$  properly. Finally we put the upper bounds of  $\epsilon_{t,n}^{(1)}$  and  $\epsilon_{t,n}^{(2)}$  together to conclude

$$\mathbb{E} \left[ 1_{A(\delta, n)^c} \cdot \left\| \Psi_M^{tr}(x)(\widehat{\beta}_t^n - \beta_t) \right\|_{\infty} \right] \leq C \left( \frac{M^{3/2}}{L^{1/4}} \right) \mathbb{E} \left[ \left\| \mathfrak{J}_{t,n}(\xi|t, x) \right\|_{\infty} \cdot 1_{A(\delta, n-1)^c} \right]. \square$$

### D.3.2 Proof of Theorem 4.5

Let

$$\delta = \frac{1}{2M^{1/2}L^{1/4}}$$

as in Lemma D.9 and define  $A(\delta, T)$  accordingly. We decompose

$$\mathbb{E} \left[ \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right] = \mathbb{E} \left[ 1_{A(\delta, T)} \cdot \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right] + \mathbb{E} \left[ 1_{A(\delta, T)^c} \cdot \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right].$$

*Step 1.* We plan to develop a bound for

$$\mathbb{E} \left[ 1_{A(\delta, T)} \cdot \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right].$$

To this end, according to Lemma D.6,

$$\mathbb{P}(A(\delta, T)) \leq \sum_{\substack{1 \leq n \leq T, \\ T-n+1 \leq t \leq T}} \mathbb{P}(\|I - (B_{\psi\psi}^t)^{-1} \widehat{B}_{\psi\psi}^{t,n}\|_2 \geq \delta) \leq T(T+1)M \exp\left(-\frac{L^{1/2}}{CM^3}\right).$$

Since  $V_0(x)$  is bounded by Assumption 4.4 and  $\widehat{\mathfrak{Y}}_0^{T+1}(x)$  is also truncated by pre-specified constant  $K$  as stated in Section D.1, there should exist a constant  $C$  such that

$$\mathbb{E}[1_{A(\delta, T)} \cdot |\widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x)|] \leq C\mathbb{P}(A(\delta, T)) \leq CT(T+1)M \exp\left(-\frac{L^{1/2}}{CM^3}\right). \quad (83)$$

*Step 2.* We intend to establish the relationship between  $\|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty$  and  $\|\widehat{\mathfrak{Y}}_t^{n-1}(x) - V_t(x)\|_\infty$  for  $1 \leq n \leq T$  and  $T-n+1 \leq t \leq T$ . Our claim is that

$$\begin{aligned} & \mathbb{E}[1_{A(\delta, n)^c} \|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty] \\ & \leq (1+l_M)\Delta + C\frac{M^{3/2}}{L^{1/4}} + (2l_M + C\frac{M^{3/2}}{L^{1/4}}) \sum_{s=t+1}^T \mathbb{E}[1_{A(\delta, n-1)^c} \|\widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x)\|_\infty]. \end{aligned} \quad (84)$$

To show this, by adding and subtracting the term  $\Psi_M^{tr}(x)\beta_t^n$  at the same time within  $\|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty$ , we have

$$\begin{aligned} \left\| \widehat{\mathfrak{Y}}_t^n(x) - V_t(x) \right\|_\infty &= \left\| \Psi_M^{tr}(x)\widehat{\beta}_t^n - V_t(x) \right\|_\infty \\ &= \left\| \left( \Psi_M^{tr}(x)\widehat{\beta}_t^n - \Psi_M^{tr}(x)\beta_t^n \right) + \left( \Psi_M^{tr}(x)\beta_t^n - V_t(x) \right) \right\|_\infty. \end{aligned}$$

Then, according to the triangle inequality,

$$\left\| \widehat{\mathfrak{Y}}_t^n(x) - V_t(x) \right\|_\infty \leq \left\| \Psi_M^{tr}(x)\widehat{\beta}_t^n - \Psi_M^{tr}(x)\beta_t^n \right\|_\infty + \left\| \Psi_M^{tr}(x)\beta_t^n - V_t(x) \right\|_\infty. \quad (85)$$

Note that Lemma D.9 provides the upper bound on  $\mathbb{E}[1_{A(\delta, n)^c} \|\Psi_M^{tr}(x)\widehat{\beta}_t^n - \Psi_M^{tr}(x)\beta_t^n\|_\infty]$ . Henceforth we only need to consider how to bound the second part in right hand side of (85).

Let

$$\beta_t = \arg \min_{\alpha} \mathbb{E}^G[(V_t(x) - \Psi_M^{tr}(x)\alpha)^2].$$

We add and subtract the term  $\Psi_M^{tr}(x)\beta_t$  simultaneously in the next equation and use the triangle inequality again,

$$\begin{aligned} \left\| \Psi_M^{tr}(x)\beta_t^n - V_t(x) \right\|_\infty &= \left\| \left( \Psi_M^{tr}(x)\beta_t^n - \Psi_M^{tr}(x)\beta_t \right) + \left( \Psi_M^{tr}(x)\beta_t - V_t(x) \right) \right\|_\infty \\ &\leq \left\| \Psi_M^{tr}(x)\beta_t^n - \Psi_M^{tr}(x)\beta_t \right\|_\infty + \left\| \Psi_M^{tr}(x)\beta_t - V_t(x) \right\|_\infty. \end{aligned} \quad (86)$$

Under the basis function set  $\Psi_M(x)$ ,  $\Psi_M^{tr}(x)(\beta_t^n - \beta_t)$  is the least square estimation of function  $\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)$ . Recall the definition of  $l_M$  in (56). Then

$$\left\| \Psi_M^{tr}(x)\beta_t^n - \Psi_M^{tr}(x)\beta_t \right\|_\infty \leq l_M \|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty.$$

To bound  $\|\Psi_M^{tr}(x)\beta_t - V_t(x)\|_\infty$ , Lemma 2.4 in Chen and Christensen (2015) shows that

$$\|\Psi_M^{tr}(x)\beta_t - V_t(x)\|_\infty \leq (l_M + 1)\Delta,$$

with  $\Delta$  representing the approximation error as defined in the Theorem statement.

Therefore  $\|\Psi_M^{tr}(x)\beta_t^n - V_t(x)\|_\infty$  satisfies

$$\|\Psi_M^{tr}(x)\beta_t^n - V_t(x)\|_\infty \leq l_M \|\underline{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty + (l_M + 1)\Delta. \quad (87)$$

From Lemma D.9, relationship (85) and (87), we have

$$\begin{aligned} & \mathbb{E}\left[1_{A(\delta,n)^c} \|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty\right] \\ \leq & C \frac{M^{3/2}}{L^{1/4}} \mathbb{E}\left[\|\mathfrak{J}_{t,n}(\xi|t,x)\|_\infty 1_{A(\delta,n-1)^c}\right] + l_M \mathbb{E}\left[1_{A(\delta,n)^c} \|\underline{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty\right] + (l_M + 1)\Delta. \end{aligned} \quad (88)$$

We need to bound each term in the last line of above inequality. According to the Corollary D.8, we have

$$\mathbb{E}\left[\left\|\underline{\mathfrak{Y}}_t^n(x) - V_t(x)\right\|_\infty \cdot 1_{A(\delta,n)^c}\right] \leq 2 \sum_{s=t+1}^T \mathbb{E}\left[\left\|\widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x)\right\|_\infty \cdot 1_{A(\delta,n-1)^c}\right]. \quad (89)$$

For  $\|\mathfrak{J}_{t,n}(\xi|t,x)\|_\infty$ , it satisfies

$$\left\|\mathfrak{J}_{t,n}(\xi|t,x)\right\|_\infty \leq \left\|\mathfrak{J}_{t,n}(\xi|t,x) - V_t(x)\right\|_\infty + \left\|V_t(x)\right\|_\infty.$$

As the optimal value function  $V_t(x)$  is bounded on compact set  $\mathcal{X}$  in Assumption 4.4, there exists a constant  $C$  such that

$$\begin{aligned} \mathbb{E}\left[\left\|\mathfrak{J}_{t,n}(\xi|t,x)\right\|_\infty \cdot 1_{A(\delta,n-1)^c}\right] & \leq C + \mathbb{E}\left[\left\|\mathfrak{J}_{t,n}(\xi|t,x) - V_t(x)\right\|_\infty \cdot 1_{A(\delta,n-1)^c}\right] \\ & \leq C + 2 \sum_{s=t+1}^T \mathbb{E}\left[\left\|\widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x)\right\|_\infty \cdot 1_{A(\delta,n-1)^c}\right]. \end{aligned} \quad (90)$$

We combine (88-90),

$$\begin{aligned} & \mathbb{E}\left[1_{A(\delta,n)^c} \|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\|_\infty\right] \\ \leq & (1 + l_M)\Delta + C \frac{M^{3/2}}{L^{1/4}} + (2l_M + C \frac{M^{3/2}}{L^{1/4}}) \sum_{s=t+1}^T \mathbb{E}\left[1_{A(\delta,n-1)^c} \|\widehat{\mathfrak{Y}}_s^{n-1}(x) - V_s(x)\|_\infty\right]. \end{aligned}$$

*Step 3.* From (84) in *Step 2*, we use induction on  $n$  to show that for  $1 \leq t \leq T$  and  $n \geq T - t + 1$ ,

$$\mathbb{E}\left[1_{A(\delta,n)^c} \cdot \left\|\widehat{\mathfrak{Y}}_t^n(x) - V_t(x)\right\|_\infty\right] \leq \left(1 + 2l_M + C \frac{M^{3/2}}{L^{1/4}}\right)^{T-t} \left[(1 + l_M)\Delta + C \frac{M^{3/2}}{L^{1/4}}\right]. \quad (91)$$

We omit the calculation detail in the interest of space.

*Step 4.* In light of the definition of  $\widehat{\mathfrak{Y}}_0^{T+1}(x)$ ,

$$\widehat{\mathfrak{Y}}_0^{T+1}(x) = \max \left\{ -K, \min \left\{ K, \frac{1}{L} \sum_{l=1}^L \mathfrak{J}_{0,T+1}^{(l)} \right\} \right\},$$

we choose constant  $K$  such that  $K \geq |V_0(x)|$ . According to Lemma D.3, we have

$$\left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \leq \left| \frac{1}{L} \sum_{l=1}^L \mathfrak{J}_{0,T+1}^{(l)} - V_0(x) \right| \leq \frac{1}{L} \sum_{l=1}^L \left| \mathfrak{J}_{0,T+1}^{(l)} - V_0(x) \right|.$$

Again we use the Lemma D.7,

$$\mathbb{E} \left[ \mathbf{1}_{A(\delta,T)^c} \cdot \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right] \leq 2 \sum_{t=1}^T \mathbb{E} \left[ \mathbf{1}_{A(\delta,T)^c} \cdot \left\| \widehat{\mathfrak{Y}}_t^T(x) - V_t(x) \right\|_{\infty} \right]. \quad (92)$$

We sum the inequality (91) from  $t = 1$  to  $T$  in iteration  $T$  and derive that

$$\mathbb{E} \left[ \mathbf{1}_{A(\delta,T)^c} \cdot \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right] \leq \left( 1 + 2l_M + C \frac{M^{3/2}}{L^{1/4}} \right)^T \left[ (1 + l_M) \Delta + C \frac{M^{3/2}}{L^{1/4}} \right].$$

*Step 5.* By combining the result of (83) and (92), we conclude that

$$\mathbb{E} \left[ \left| \widehat{\mathfrak{Y}}_0^{T+1}(x) - V_0(x) \right| \right] \leq CT(T+1)M \exp \left( -\frac{L^{1/2}}{CM^3} \right) + \left( 1 + 2l_M + C \frac{M^{3/2}}{L^{1/4}} \right)^T \left[ (1 + l_M) \Delta + C \frac{M^{3/2}}{L^{1/4}} \right].$$

For sufficient small  $\alpha$ , we have

$$M \exp \left( -\frac{L^{1/2}}{CM^3} \right) \leq \left( 1 + 2l_M + C \frac{M^{3/2}}{L^{1/4}} \right)^T \frac{M^{3/2}}{L^{1/4}}.$$

By adjusting the constant  $C$  properly, we obtain the result in Theorem 4.5.  $\square$

## E Supplementary Materials to Section 5

### E.1 Optimal Order Execution Problem:

- **The objective function:**

It is easy to see that minimizing (33) is equivalent to minimizing

$$\mathbb{E} \left[ \sum_{t=1}^T \mathbf{P}_t^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} \right].$$

Note that the constant  $\tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}}$  stands for the cost that the trader would pay for  $\bar{\mathbf{R}}$  shares of assets without the price impacts. This difference thus represents the implementation shortfall of a specific strategy, namely how much more costs the trader may incur during the course of fulfilling the execution target. In the following lemma, we show that it equals (34).

**Lemma E.1** For the trader's problem

$$\min_{\{\mathbf{S}_t, 1 \leq t \leq T\}} \mathbb{E} \left[ \left( \sum_{t=1}^T \mathbf{P}_t^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} \right) \right], \quad (93)$$

subject to the constraints (30-32), it is equivalent to

$$\min_{\{\mathbf{S}_t, 1 \leq t \leq T\}} \mathbb{E} \left[ \sum_{t=1}^T \mathbf{S}_t^{tr} h(\mathbf{S}_t) + \sum_{t=0}^{T-1} (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1} \right].$$

*Proof of Lemma E.1.* Using the relationship (31), we observe that

$$\begin{aligned} \sum_{t=1}^T \mathbf{P}_t^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} &= \sum_{t=1}^T (\tilde{\mathbf{P}}_t + h(\mathbf{S}_t))^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} \\ &= \sum_{t=1}^T \mathbf{S}_t^{tr} h(\mathbf{S}_t) + \sum_{t=1}^T \tilde{\mathbf{P}}_t^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}}. \end{aligned} \quad (94)$$

In addition, applying Abel's summation-by-part formula to  $\sum_{t=1}^T \tilde{\mathbf{P}}_t^{tr} \mathbf{S}_t$ , we know that

$$\begin{aligned} \sum_{t=1}^T \tilde{\mathbf{P}}_t^{tr} \mathbf{S}_t &= \tilde{\mathbf{P}}_0^{tr} \left( \sum_{t=1}^T \mathbf{S}_t \right) + \sum_{t=0}^{T-1} \left( (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \cdot \sum_{j=t+1}^T \mathbf{S}_j \right) \\ &= \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} + \sum_{t=0}^{T-1} (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1}. \end{aligned} \quad (95)$$

Thus, with (94) and (95), we have

$$\sum_{t=1}^T \mathbf{P}_t^{tr} \mathbf{S}_t - \tilde{\mathbf{P}}_0^{tr} \bar{\mathbf{R}} = \sum_{t=1}^T \mathbf{S}_t^{tr} h(\mathbf{S}_t) + \sum_{t=0}^{T-1} (\tilde{\mathbf{P}}_{t+1} - \tilde{\mathbf{P}}_t)^{tr} \mathbf{R}_{t+1}.$$

This verifies the equivalence of these two objective functions. Note the new value function doesn't depend on the variable  $\mathbf{P}$ .  $\square$

#### - The auxiliary LQC problem:

If we ignore the temporary impact  $h(\mathbf{S}_t)$  and remove the nonnegative constraint  $\mathbf{S}_t \geq 0$ , the problem (34) with the constraints (30-32) is equivalent to LQC problem. According to the discussion in Appendix B, the value function  $V_t(\mathbf{X}_t, \mathbf{R}_t)$  and policy  $\mathbf{S}_t^*$  are:

$$V_t(\mathbf{X}_t, \mathbf{R}_t) = \mathbf{X}_t^{tr} \mathbf{W}_t \mathbf{X}_t + \mathbf{R}_t^{tr} \mathbf{Q}_t \mathbf{R}_t + \mathbf{R}_t^{tr} \mathbf{K}_t \mathbf{X}_t + \mathbf{H}_t, \quad (96)$$

$$\mathbf{S}_t^*(\mathbf{X}_t, \mathbf{R}_t) = (\mathbf{I} - \frac{1}{2} \mathbf{Q}_{t+1}^{-1} \mathbf{A}^{tr}) \mathbf{R}_t + \frac{1}{2} \mathbf{Q}_{t+1}^{-1} \mathbf{K}_{t+1} \mathbf{C} \mathbf{X}_t, \quad (97)$$

with

$$\mathbf{Q}_t = -\frac{1}{4}\mathbf{A}\mathbf{Q}_{t+1}^{-1}\mathbf{A}^{tr} + \frac{1}{2}(\mathbf{A} + \mathbf{A}^{tr}), \quad \mathbf{Q}_T = \frac{1}{2}(\mathbf{A} + \mathbf{A}^{tr}). \quad (98)$$

$$\mathbf{W}_t = \mathbf{C}^{tr}\mathbf{W}_{t+1}\mathbf{C} - \frac{1}{4}\mathbf{C}^{tr}\mathbf{K}_{t+1}^{tr}\mathbf{Q}_{t+1}^{-1}\mathbf{K}_{t+1}\mathbf{C}, \quad \mathbf{W}_T = 0. \quad (99)$$

$$\mathbf{K}_t = \mathbf{B} + \frac{1}{2}\mathbf{A}\mathbf{Q}_{t+1}^{-1}\mathbf{K}_{t+1}\mathbf{C}, \quad \mathbf{K}_T = \mathbf{B}. \quad (100)$$

$$\mathbf{H}_t = \mathbf{H}_{t+1} + \mathbb{E}[\boldsymbol{\eta}^{tr}\mathbf{W}_{t+1}\boldsymbol{\eta}], \quad \mathbf{H}_T = 0. \quad (101)$$

Specially if the matrix  $\mathbf{A}$  is symmetric, the optimal policy (97) can be simplified as

$$\mathbf{S}_t^*(\mathbf{X}_t, \mathbf{R}_t) = \frac{1}{2}\mathbf{Q}_{t+1}^{-1}\mathbf{K}_{t+1}\mathbf{C}\mathbf{X}_t + \frac{1}{T-t+1}\mathbf{R}_t.$$

- **Parameter setting:**

To illustrate the numerical results, we consider a case with three assets and a signal vector of two variables. Assume the trader wants to buy  $1 \times 10^5$  shares for each asset within  $T = 20$  periods, i.e.,  $\bar{R}_i = 1 \times 10^5$  for  $i = 1, 2, 3$ . The parameter matrices pertinent to the temporary and permanent impacts are supposed to

$$\mathbf{A} = \begin{bmatrix} 30 & 7 & 3 \\ 7 & 25 & -5 \\ 3 & -5 & 20 \end{bmatrix} \times 10^{-6}, \quad \mathbf{B} = \begin{bmatrix} 5 & 2 \\ 3 & 2 \\ 1 & 4 \end{bmatrix}, \quad \mathbf{C} = \delta \times \begin{bmatrix} 0.8 & 0.1 \\ 0.2 & 0.6 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 2\lambda & 0 & 0 \\ 0 & 2\lambda & 0 \\ 0 & 0 & 2\lambda \end{bmatrix} \times 10^{-5}, \quad \Sigma_\eta = \begin{bmatrix} 1.0 & 0.2 \\ 0.2 & 0.8 \end{bmatrix}.$$

Here we parametrize matrix  $\mathbf{D}$  by  $\lambda$  so that we can examine the effect of the temporary price impact on the optimal execution strategies by varying  $\lambda$ .

## E.2 Inventory Management Problem:

- **Parameters:** The model parameters' values used in the experiments are given by

$$h = 1, \quad m = 4 \text{ or } 9, \quad p = 9 \text{ or } 19, \quad \gamma = 1, \quad T = 30, \quad \mathbf{x}_1 = \mathbf{0}.$$

- **Basis functions:**

For  $L = 4$ , we choose the basis function set as

$$\left\{ 1, \quad (x_{i,t})_{0 \leq i \leq 3}, \quad \mathbb{E}[(x_{0,t} - \tilde{d}_0)^+], \quad \mathbb{E}[((x_{0,t} - \tilde{d}_0)^+ + x_{1,t} - \tilde{d}_1)^+], \right. \\ \mathbb{E}[(((x_{0,t} - \tilde{d}_0)^+ + x_{1,t} - \tilde{d}_1)^+ + x_{2,t} - \tilde{d}_2)^+], \quad \mathbb{E}[((((x_{0,t} - \tilde{d}_0)^+ + x_{1,t} - \tilde{d}_1)^+ + x_{2,t} - \tilde{d}_2)^+ + x_{3,t} - \tilde{d}_3)^+], \\ \left. \mathbb{E}[(((x_{1,t} - \tilde{d}_1)^+ + x_{2,t} - \tilde{d}_2)^+ + x_{3,t} - \tilde{d}_3)^+], \quad \mathbb{E}[((x_{2,t} - \tilde{d}_2)^+ + x_{3,t} - \tilde{d}_3)^+], \quad \mathbb{E}[(x_{3,t} - \tilde{d}_3)^+] \right\}.$$

The expectation is taken over  $(\tilde{d}_i)_{0 \leq i \leq 3}$ , which have the same distribution with  $d_t$  in the system. For  $L = 10$ , we choose 30 basis functions in similar manner as  $L = 4$ . That is, constant 1, one order function  $(x_{i,t})_{0 \leq i \leq 9}$ , the expectation in iteration form from  $\mathbb{E}[(x_{0,t} - \tilde{d}_0)^+]$  to  $\mathbb{E}[((x_{0,t} - \tilde{d}_0)^+ \cdots + x_{9,t} - \tilde{d}_9)^+]$ , and the reverse form from  $\mathbb{E}[(x_{9,t} - \tilde{d}_9)^+]$  to  $\mathbb{E}[((x_{1,t} - \tilde{d}_1)^+ \cdots + x_{9,t} - \tilde{d}_9)^+]$ .



- **Quasi Monte Carlo:**

As mentioned in the main body, for  $L = 10$ , we choose low-discrepancy sequences to perform the nested simulations. To illustrate this, we note that the expectation of basis functions can be written in the form of

$$H(\mathbf{x}_t) = \mathbb{E}[g(\mathbf{x}_t, \mathbf{d})]$$

for some function  $g(\cdot)$  and where state  $\mathbf{x} = [x_{0,t}, \dots, x_{9,t}]$ , geometric distribution  $\mathbf{d} = [\tilde{d}_0, \dots, \tilde{d}_9]$ . Using the inverse transform approach we can easily rewrite  $H(\mathbf{x}_t)$  as

$$H(\mathbf{x}_t) = \mathbb{E}[g(\mathbf{x}_t, \lfloor \log(\mathbf{U}) / \log(1 - p) \rfloor)],$$

where  $\mathbf{U}$  is 10-dimensional vector of independent uniform random variables in range  $(0, 1)$  and  $\lfloor N \rfloor$  stands for the largest integer which is no bigger than  $N$ . We could perform this expectation with respect to  $\mathbf{U}$  by using the low discrepancy sequence. Here we choose 2047 points in Sobol sequence,  $U_1, \dots, U_{2047}$ , and approximate

$$H(\mathbf{x}_t) \approx \sum_{i=1}^{2047} g(\mathbf{x}_t, \lfloor \log(U_i) / \log(1 - p) \rfloor).$$

The detailed discussion about this method, one may refer to Chapter 5 in Glasserman (2004).

## References

- Altman E (1999) *Constrained Markov Decision Processes*, CRC Press, US.
- Belloni A, Chernozhukov V, Chetverikov D, Kato K (2015) Some New Asymptotic Theory for Least Squares Series: Pointwise and Uniform Results. *J. Econom.* **186**: 345–366.
- Chen X, Christensen TM (2015) Optimal Uniform Convergence Rates and Asymptotic Normality for Series Estimators Under Weak Dependence and Weak Conditions. *J. Econom.* **188**: 447–465.
- DeVore RA, Lorentz GG (1993) *Constructive Approximation*, Springer-Verlag, Berlin, Germany.
- Dufour F, Prieto-Rumeau T (2012) Approximation of Markov decision processes with general state space. *J. Math. Analysis App.* **388**: 1254–1267.
- Glasserman P (2004) *Monte Carlo Methods in Financial Engineering*, Springer-Verlag, New York, USA.
- Horn R, Johnson C (2003). *Matrix Analysis*, Cambridge University Press, UK.
- Horst R, Pardalos PM, Thoai NV (2000) *Introduction to Global Optimization*, 2nd Edition. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Huang JZ (2003) Local asymptotics for polynomial spline regression. *Ann. Stat.* **31**: 1600–1635.

- Kushner HJ, Dupuis P (2001) *Numerical Methods for Stochastic Control Problems in Continuous Time*, Springer, US.
- Nocedal J, Wright SJ (1999) *Numerical Optimization*. Springer-Verlag, New York.
- Saldi N, Linder T, Yuksel S (2018) *Finite Approximations in Discrete-Time Stochastic Control*, Springer, US.
- Schumaker L (1981) *Spline Functions: Basic Theory*, John Wiley & Sons, New York.
- Tropp JA (2012) User-Friendly Tail Bounds for Sums of Random Matrices. *Found. Comput. Math.* **12**: 389–434.
- Timan AF (1963) *Theory of Approximation of Functions of a Real Variable*, MacMillan, New York.
- Zygmund A (2002) *Trigonometric Series*. Cambridge Mathematical Library.