Imperial College London
Department of Mathematics

Duality and Policy Gradient Methods for Stochastic Control Problems with Controlled Diffusions

Ashley Davey

Thesis presented for exmaination for the degree of
Doctor of Philosophy in Mathematics
of Imperial College London

March 2023

Statement of Originality

I certify that this thesis, and the research to which it refers, are the product of my own work, and that any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Ashley Davey

Copyright Declaration

Acknowledgements

I would like to extend my heartfelt thanks to the following people, without whom this thesis would not have been possible.

First and foremost, I would like to thank my supervisor Professor Harry Zheng for his support and valuable discussions throughout my PhD. His guidance and support were fundamental in this achievement. Not only did I benefit from his experience and tutorage, but he also enabled collaboration and useful discussions with a wide range of experts in the field – a particularly difficult challenge given the pandemic. In particular, I would like to thank Professor Michael Monoyios. I learned a lot through our discussions during our paper collaboration.

I would additionally like to thank Dr Eyal Neuman and Dr Panos Parpas for their useful input in my research. They were my examiners throughout my PhD and provided interesting ideas and advice during our discussions.

I would like to thank everyone in my CDT cohort for their support, and Professor Philip Treleaven for funding and his supervisory role for my PhD. I benefited greatly from the seminars and discussions within our cohort.

Finally, I would like to thank my friends, family, and my girlfriend for their unwavering support throughout my PhD. They always believed in me and kept me going, and I cannot begin to express my gratitude to them.

Abstract

In this thesis we develop numerical algorithms for stochastic control problems where the state processes is an Itô process that depends on the control process in both the drift and diffusion functions.

We derive sufficient conditions for convergence of a proximal policy gradient method (PPGM) for the stochastic linear quadratic control problem. The optimal control can be uniquely determined by solving a Ricatti equation. We prove that the control induced by the policy gradient method converges to the optimal control. Considering convergence analysis in the general case, we study the underlying backwards stochastic differential equations using Malliavin Calculus, and determine some conditions under which convergence of the control process can be established.

To implement the PPGM method, we use concepts in machine learning methods, and extend our algorithms using duality theory. Using deep learning methods, the algorithms developed are scalable to high dimensional problems, with a reasonable runtime. Duality allows us to solve an auxiliary control problem simultaneously to the original primal problem, which allows us to exploit the dual-primal relations between the associated processes, and form tight bounds of the value function. In certain cases solving the dual problem directly bypasses complexity within the primal problem. We further use duality to implement algorithms that can be applied to non-Markovian control problems.

Keywords: Stochastic control, deep learning, primal/dual BSDEs, HJB equation, utility maximisation, linear quadratic, PPGM, convergence analysis, constrained optimisation.

Contents

1 Preliminaries **19**

1.1 Notation 19

1.2 Model Formulation 20

1.3 The Dynamic Programming Principle 22

1.4 The HJB Equation 22

1.5 Stochastic Maximum Principle 24

1.6 Malliavin Calculus 25

1.7 Neural Networks and Deep Reinforcement Learning 27

1.8 Stochastic Gradient Descent 29

1.9 Organisation and Contributions 31

2 Duality and Machine Learning Algorithms for Stochastic Control Problems **33**

2.1 The Deep Controlled 2BSDE Method 38

2.2 Utility Maximisation 43

2.2.1 Primal Problem 44

2.2.2 Dual Problem 46

2.3 Duality Methods 47

2.3.1 The Dual DC2BSDE Method for the Markovian Case 48

2.3.2 The Deep SMP Method 51

2.4 Random Income 55

2.4.1 Primal Problem 55

2.4.2 Dual Problem 58

3 Convergence of the PPGM algorithm for the LQ Problem **63**

3.1 Problem Formulation . . . 65

3.2 Proximal Policy Gradient Method . . . 67

3.3 Main Result . . . 69

3.4 Sufficient Conditions for the Constrained Case . . . 73

3.4.1 Showing $[\phi^k]_1$ is Uniformly Bounded . . . 73

3.4.2 Convergence in $|.|_0$ norm . . . 82

3.4.3 Proof of Main Result . . . 90

3.5 Convergence of PGM for Unconstrained Case . . . 92

3.5.1 Showing $[\phi^k]_1$ is Uniformly Bounded . . . 100

3.5.2 Showing $|\phi^k|_0$ is Uniformly Bounded . . . 104

3.5.3 Showing $\mathcal{L}$ is a Contraction in $|\cdot|_0$ . . . 106

3.5.4 Proof of Main Result . . . 113

4 Numerical Examples **118**

4.1 Utility Maximisation . . . 119

4.1.1 Methodology Comparison . . . 126

4.1.2 Quadratic Utility . . . 130

4.2 Investment-Consumption with Randomly Terminating Income . . . 143

4.2.1 Finding the true state process . . . 143

4.2.2 Numerical Examples . . . 145

4.2.2.1 Power Utility . . . 145

4.3 LQ Problem . . . 150

4.3.1 Unconstrained . . . 150

4.3.1.1 Implementation and an Example . . . 151

4.3.1.2 Dimensionality . . . 155

4.3.1.3 Assumptions . . . 156

4.3.2 Constrained . . . 158

4.3.2.1 Equivalence of PGM and DCBSDE . . . 158

4.3.2.2 Dimensionality . . . 162

4.3.2.3 Neural Network Approximation Capabilities . . . 162

5 Conclusions **165**

A Additional Material **172**

A.1 Auxiliary Results . . . 172

A.2 List of Constants . . . 179

List of Tables

4.1 Value approximation and runtime at certain iteration steps for the primal and dual deep controlled 2BSDE methods applied to the cone-constrained power utility problem. . . . 124

4.2 Value approximation and runtime at certain iteration steps for the primal and dual deep controlled 2BSDE methods applied to the ball-constrained log utility problem. . . . 126

4.3 Comparison of the Primal DC2BSDE and (primal value approximation of the) DSMP algorithm to the PVO and Hybrid algorithms for the unconstrained Power Utility problem. . . . 128

4.4 Value, error and runtime of the Deep 2BSDE method and primal DC2BSDE method for the unconstrained Power Utility problem. . . . 130

4.5 Accuracy and runtime for the deep controlled 2BSDE method for various terminal times T and numbers of time steps N applied to the unconstrained power utility problem in a stochastic volatility model. . . . 140

4.6 Accuracy and runtime for the deep SMP method for various terminal times T and numbers of time steps N applied to the unconstrained power utility problem in a stochastic volatility model. . . . 141

4.7 Value approximations for the DC2BSDE and DCSMP algorithms applied to the unconstrained non-HARA utility problem in a stochastic volatility model. . . . 142

4.8 Duality gap for the deep SMP method for various terminal timed T and numbers of time steps N applied to the cone-constrained power utility problem with path dependent volatility. . . . 144

4.9 Results for $a = 0, \eta = 0$ for power utility. . . . 147

4.10 Results for $a = 0, \eta = 2$ for power utility. . . . 148

4.11 Results for $a = 0.1, \eta = 0$ for power utility. . . . 148

4.12 Results for $a = 0.1, \eta = 2$ for power utility. . . . 149

List of Figures

4.1 Loss functions against iteration step for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem. . . . 121

4.2 Two simulations of the value process and value-derivative process for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem. The displayed dual processes are those implied by the duality relations (2.21). 121

4.3 Approximation and relative error of the value function against number of time steps N for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem. Note in the left graph that the Y values on the axis are shifted down by 2.253. 122

4.4 Approximation of the value function and bsde loss against iteration step for the primal and dual deep controlled 2BSDE methods applied to the cone constrained power utility problem. 124

4.5 The quadratic utility U for varying (α, H). 131

4.6 Control error (in $\mathcal{H}^2$) and value approximation of the primal and dual DC2BSDE methods and the DSMP method for the unconstrained quadratic utility problem. 138

4.7 Sensitivity of initial value for power utility. 148

4.8 A sample path of random income state and control processes, $T = 0.5$, $N = 20$ with $\eta = 2$, $a = 1.0$ for power utility. The sample termination time is indicated by the vertical dotted line. 149

4.9 Initial control strategies and value for different initial wealths for power utility. . 151

4.10 Relative control error of the unconstrained PGM method applied to the LQ problem, compared to approximated and theoretical convergence rates. 154

4.11 Value function approximation, in terms of initial value and error of the process a^k in $\mathcal{L}^\infty$. 155

4.12 Run time in seconds against dimension $n = m$. 156

4.13 Control error vs r for large values of r with $m = 5$, and values error against r for small values of r with $m = 1$. 158

4.14 Relative Error and control loss function against iteration step for the cone constrained LQ problem. 161

4.15 3 Simulations of the state and control process for the cone constrained LQ problem. 161

4.16 3 Simulations of the value and adjoint processes for the cone constrained LQ problem. 162

4.17 Runtime and accuracy against control dimension for the cone constrained LQ problem. 163

4.18 Control neural network and solution function at $t = 0.98$ for the unconstrained LQ problem. 164

Chapter 1

Preliminaries

1.1 Notation

For two measure spaces (A, μ_A) and (B, μ_B), let $\mathcal{B}(A; B)$ denote the set of measurable functions from A to B. Fix $T < \infty$. Let $(E, |\cdot|)$ be a Euclidean space, and let $\mathcal{B}_E = \mathcal{B}([0,T]; E)$ be the set of progressively measurable processes valued in E. Then for any $p \geq 2$ and $0 \leq t \leq T < \infty$ define

$$\mathcal{S}^p(t,T;E) = \left\{ Y \in \mathcal{B}_E \colon \|Y\|_{\mathcal{S}^p} := \mathbb{E}\left[\sup_{t \leq s \leq T} |Y_s|^p\right]^{\frac{1}{p}} < \infty \right\},$$
$$\mathcal{H}^p(t,T;E) = \left\{ Z \in \mathcal{B}_E \colon \|Z\|_{\mathcal{H}^p} := \mathbb{E}\left[\left(\int_t^T |Z_s|^2 ds\right)^{\frac{p}{2}}\right]^{\frac{1}{p}} < \infty \right\}.$$

Here and in the sequel, we denote $|\cdot|$ as the Euclidean norm in the appropriate space. We then denote $\mathcal{S}^p(E) = \mathcal{S}^p(0,T;E)$ and $\mathcal{H}^p(E) = \mathcal{H}^p(0,T;E)$.

Let (S, μ) be some measure space, and $f \colon S \to E$ a function. We denote the $\mathcal{L}^\infty$ and $\mathcal{L}^p$, $p \geq 1$, norms as

$$\|f\|_\infty := \sup_{\omega \in \Omega} |f(\omega)|,$$
$$\|f\|_p := \left(\int_\Omega |f(\omega)|^p d\omega\right)^{\frac{1}{p}}.$$

we then define , for $p \in [1,\infty]$,

$$\mathcal{L}^p(S) = \{f \in \mathcal{B}(S,\mathbb{R})\colon \|f\|_p \leq \infty\},$$
$$\mathcal{L}^p\,(0,T;E) = \{f \in \mathcal{B}_E \text{ progressively measurable} : \|f\|_p < \infty\}.$$

1.2 Model Formulation

Let $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0\leq t\leq T}, \mathbb{P}\right)$ be a filtered probability space with terminal horizon $T \in (0,\infty)$. We consider the $\mathcal{F}_t$-adapted controlled diffusion process $(X_t^u)_{0\leq t\leq T}$ on $\mathbb{R}^n$, $n \in \mathbb{N}$, defined by $X_0^u = x_0 \in \mathbb{R}^n$ and

$$dX_t^u = b_t\left(X_t^u, u_t\right) dt + \sigma_t\left(X_t^u, u_t\right) dW_t, \qquad 0 \leq t \leq T, \tag{1.1}$$

where W is an $\mathbb{R}^d$-valued Brownian motion for some $d \in \mathbb{N}$ and $u = (u_t)_{0\leq t\leq T}$ is a progressively measurable process, valued in a closed convex set $U \subset \mathbb{R}^m$ for some $m \in \mathbb{N}$. We have the drift and volatility functions $b\colon [0,T]\times\mathbb{R}^n\times U \to \mathbb{R}^n$ and $\sigma\colon [0,T]\times\mathbb{R}^n\times U \to \mathbb{R}^{n\times d}$ are assumed to be measurable and uniformly Lipschitz in x. This condition guarantees us a unique strong solution to this SDE. Here X^u represents our state process, and u is the control. We consider only admissible controls, that are contained in the set

$$\mathcal{U} := \left\{u \in \mathcal{H}^2\left(U\right) : \mathbb{E}\left[\int_0^T \left|b_t\left(0,u_t\right)\right|^2 + \left|\sigma_t\left(0,u_t\right)\right|^2 dt\right] < \infty\right\}.$$

This control is chosen to optimise the state process via minimising a running cost function $f\colon [0,T]\times\mathbb{R}^n\times U \to \mathbb{R}$ and a terminal cost $g\colon \mathbb{R}^n \to \mathbb{R}$, which are measurable functions with g either lower bounded or satisfying some quadratic growth condition. In order to have a well defined finite value function, we need to define

$$\mathcal{U}\left(t,x\right) := \left\{u \in \mathcal{U}\colon \mathbb{E}\left[\int_t^T \left|f_s\left(X_s^{t,x,u}, u_s\right)\right| ds\right] < \infty\right\}, \qquad (t,x) \in [0,T]\times\mathbb{R}^n$$

where $(X_s^{t,x,u})_{t\leq s\leq T}$ is a controlled diffusion satisfying

$$X_s^{t,x,u} = b_s\left(X_s^{t,x,u}, u_s\right) ds + \sigma_s\left(X_s^{t,x,u}, u_s\right) dW_s, \qquad t \leq s \leq T, \tag{1.2}$$

such that $X_t^{t,x,u} = x$. Assuming this set is non-empty, one wishes to find the value function

$$v(t,x) := \inf_{u \in \mathcal{U}(t,x)} \mathbb{E}\left[\int_t^T f_s\left(X_s^{t,x,u}, u_s\right) ds + g\left(X_T^{t,x,u}\right)\right]. \tag{1.3}$$

We use this set throughout for optimisation, but in the sequel we simply use the notation $u \in U$ to indicate that u is a progressively measurable process valued in this set, with the correct integrability conditions. In particular we look for a Markovian control, that is, $u_s = \phi_s(X_s^u)$ for some measurable $\phi \in \mathcal{B}([0,T] \times \mathbb{R}^n; U)$. The functional representation ϕ is approximated by neural networks, which depend on a parameter set $\Theta \in \mathbb{R}^\rho$ for some (potentially large) $\rho \in \mathbb{N}$. This reduces the infinite dimensional problem to finding a finite optimal set of parameters to maximise the cost function

$$J(u) := \mathbb{E}\left[\int_0^T f_s\left(X_s^{0,x_0,u}, u_s\right) ds + g\left(X_T^{0,x_0,u}\right)\right], \qquad u \in \mathcal{U}. \tag{1.4}$$

To minimise this we use a variation of a gradient descent algorithm, but the loss function used is different for each time step. In some cases additional neural networks is used, to approximate the value function and its derivatives.

In some settings we will consider a non-Markovian structure, where the coefficients b, σ may depend on the entire path of X^u, and we may not be able to define $v(t,\cdot)$ for $t > 0$. In the algorithms treating this problem structure, we will simply find $\hat{u}$ and the optimal cost $J(\hat{u}) = v(0, x_0)$.

In our algorithms, like most numerical methods we will consider the discrete problem corresponding to problem (1.3). In particular we will consider a Euler-Maruyama scheme. We introduce a partition $0 = t_0 < t_1 < \ldots < t_{N-1} < t_N = T$ of $[0,T]$ into $N \in \mathbb{N}$ time steps. We then define the discrete time process $(X_k^u)_{k=0}^N$ by $X_0^u = x$ and for $k = 0, 1, 2, \ldots, N-1$ we define

$$X_{k+1}^u = X_k^u + (t_{k+1} - t_k)\, b_{t_k}(X_k^u, u_k) + \sqrt{t_{k+1} - t_k}\,\sigma_{t_k}(X_k^u, u_k)\,\Delta W_k, \tag{1.5}$$

where $\Delta W_0, \ldots, \Delta W_{N-1}$ are i.i.d standard multivariate normal $\mathcal{N}_d(0, \mathbb{1}_{d,d})$ random variables, with $\mathbb{1}_{d,d}$ the identity matrix of size d. The process $(u_k)_{k=0}^{N-1}$ is a piecewise constant control, which we try to optimise in order to solve the discrete control problem

$$v_N\left(t_k,x\right) := \inf_{u_j\in U}\mathbb{E}\left[\sum_{j=k}^{N-1}\left(t_{j+1}-t_j\right)f_{t_j}\left(X_j^u,u_j\right)+g\left(X_N^u\right)\right], \qquad x\in\mathbb{R}^n, k=0,\ldots,N-1. \tag{1.6}$$

Under suitable assumptions we have $v_N \to v$ as $N\to\infty$, see for example Henry-Labordere et al. (2016).

1.3 The Dynamic Programming Principle

The Dynamic Programming Principle (DPP) tells us that at a fixed time $t\in[0,T]$, we can find the optimal control u_t^* independently of any previously chosen control. To be specific, it is formulated here as the following.

Theorem 1.3.1 (Dynamical Programming Principle)**.** *Let $(t,x)\in[0,T]\times\mathbb{R}^n$. Then for all $t<s\leq T$ we have*

$$v\left(t,x\right)=\sup_{u\in\mathcal{U}(t,x)}\mathbb{E}\left[\int_t^s f_r\left(r,X_r^{t,x,u},u_r\right)dr+v\left(s,X_s^{t,x,u}\right)\right].$$

This is a weak version of this theorem, but will suffice as motivation for algorithms in the sequel. The interpretation of this theorem is that we can first find $u\left(s,x\right)$ for any $x\in\mathbb{R}^n$, for some $s\in[0,T]$, then we can use this to find $u\left(t,x\right)$ for $t\leq s$ and any $x\in\mathbb{R}^n$.

1.4 The HJB Equation

The HJB equation is often an important tool for finding solutions to the control problem (1.3). The value function v satisfies the following non-linear PDE.

$$\frac{\partial v}{\partial t}\left(t,x\right)=-\inf_{u\in U}F_t\left(x,u,\partial_x v\left(t,x\right),\partial_{xx}^2 v\left(t,x\right)\right), \qquad (t,x)\in[0,T)\times\mathbb{R}^n, \tag{1.7}$$

$$v\left(T,x\right)=g\left(x\right), \qquad x\in\mathbb{R}^n, \tag{1.8}$$

where ∂_x, ∂^2_{xx} denotes the first and second partial derivatives with respect to x respectively, and

$$F_t(x,u,y,\gamma) := b_t(x,u)^\top y + \frac{1}{2}\mathrm{tr}\left(\sigma_t(x,u)\,\sigma_t(x,u)^\top \gamma\right) + f_t(x,u), \tag{1.9}$$

for $(t,x,u,y,\gamma) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^{n\times n}$ is called the Hamiltonian of the control problem. Here, $\sigma_t(x,u)^\top$ denotes the transpose of $\sigma_t(x,u)$. This PDE can be very difficult to solve directly, especially if there is no closed form of the right hand side. Our method will aim to minimise these Hamiltonians for each $t \in [0,T]$ and $x \in \mathbb{R}^n$, rather than dealing with the cost function itself like has been done in other papers, e.g. Bachouch et al. (2021); Han and E (2016); Huré et al. (2021). The following theorem justifies this approach, as it gives conditions on which the Hamiltonian-maximising control is in fact an optimal control.

Theorem 1.4.1 (Verification Theorem (Pham, 2009, p. 47))**.** *Let $w \in \mathcal{C}^{1,2}([0,T) \times \mathbb{R}^n) \cap \mathcal{C}^0([0,T], \mathbb{R}^n)$ satisfy a quadratic growth condition, and let v be defined as in (1.3).*

(i) Suppose that

$$\begin{aligned} \frac{\partial w}{\partial t}(t,x) &= -\inf_{u\in U} F_t\left(x,u,\partial_x w(t,x), \partial^2_{xx} w(t,x)\right), && (t,x) \in [0,T) \times \mathbb{R}^n, \\ w(T,x) &\geq g(x), && x \in \mathbb{R}^n, \end{aligned}$$

Then $w \geq v$ on $[0,T] \times \mathbb{R}^n$.

(ii) Suppose further that $w(T,\cdot) = g$, and there exists a measurable function $u^* \colon [0,T] \times \mathbb{R}^n \to U$ such that, for all $(t,x) \in [0,T] \times \mathbb{R}^n$,

$$\begin{aligned} \frac{\partial w}{\partial t}(t,x) &= -\inf_{u\in U} F_t\left(x,u,\partial_x w(t,x), \partial^2_{xx} w(t,x)\right) \\ &= -F_t\left(x,u^*(t,x), \partial_x w(t,x), \partial^2_{xx} w(t,x)\right), \end{aligned}$$

and the SDE (1.2) with $\phi = u^*$ admits a unique solution X^{t,x,u^*} such that the process $\left(u^*\left(s, X^{t,x,u^*}_s\right)\right)_{t\leq s\leq T}$ lies in $\mathcal{U}(t,x)$. Then $w = v$ on $[0,T] \times \mathbb{R}^n$ and u^* is an optimal Markovian control.

To use this theorem to estimate the optimal control, we must find good approximations of the first and second derivatives of the value function. It is this problem that is often hindered by the curse of dimensionality, as methods such as finite-differences are often used in the literature. In the proceeding chapters we will see how to overcome this difficulty.

1.5 Stochastic Maximum Principle

In addition to a PDE, there is also an equivalence between the control problem (1.3) and a first order BSDE, given by the Stochastic Maximum Principle (SMP). This is a useful step in establishing the relations between derivatives of the value function, which will assist in their approximation.

Given $u \in \mathcal{U}$, define the adjoint BSDE pair $(Y^u, Z^u) \in \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{H}^2\left(\mathbb{R}^{n\times d}\right)$ by

$$\begin{aligned} dY_t^u &= -\partial_x H_t\left(X_t^u, u_t, Y_t^u, Z_t^u\right) dt + Z_t^u dW_t, \qquad 0 \le t \le T, \\ Y_T^u &= \nabla g\left(X_T^u\right), \end{aligned} \tag{1.10}$$

where ∇ denotes the Jacobian and H is the generalised Hamiltonian of the system and is given by, for $t \in [0,T]$, $x, y \in \mathbb{R}^n$, $z \in \mathbb{R}^{n\times d}$

$$H_t(x, u, z, y) := b_t(x,u)^\top y + \operatorname{tr}\left(\sigma_t(x,u)^\top z\right) + f_t(x,u). \tag{1.11}$$

We presume that H is differentiable in x.

Theorem 1.5.1 (Stochastic Maximum Principle (Pham, 2009, p. 150-151))**.** *Let $u \in \mathcal{U}$. Let $(X^u, Y^u, Z^u) \in \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{H}^2(\mathbb{R}^{n\times d})$ be solutions to (1.1) and (1.10). Suppose that for all $t \in [0,T]$*

$$H\left(t, X_t^u, u_t, Y_t^u, Z_t^u\right) = \sup_{a\in A} H\left(t, X_t^u, a, Y_t^u, Z_t^u\right) \tag{1.12}$$

and the function $(x,u) \mapsto H_t\left(x, u, Y_t^u, Z_t^u\right)$ is concave for all $t \in [0,T]$, then u is an optimal control. Furthermore, suppose that

$$v \in \mathcal{C}^{1,3}\left([0,T) \times \mathbb{R}^n\right) \cup \mathcal{C}^0\left([0,T] \times \mathbb{R}^n\right),$$

and there exists an optimal control $u^* \in \mathcal{U}$ with associated controlled diffusion X^{u^*}. Then the *pair*

$$\left(Y_t^{u^*}, Z_t^{u^*}\right) = \left(\partial_x v\left(t, X_t^{u^*}\right), \partial_{xx}^2 v\left(t, X_t^{u^*}\right)\sigma_t\left(X_t^{u^*}, u_t^*\right)\right), \qquad 0 \leq t \leq T,$$

is a solution to the adjoint BSDE (1.10) with $u = u^*$.

1.6 Malliavin Calculus

We now give some results regarding the Malliavin Differentiability of SDEs and BSDEs.

Definition 1.6.1 (Malliavin Derivative ($p = 2$))**.** Let E be a Euclidean space, and let $\mathcal{S}(E)$ denote the set of random variables ξ of the form $\xi = \psi\left(W\left(h^1\right), \ldots, W\left(h^k\right)\right)$ for some $k \in \mathbb{N}$, where $\psi \in C_b^\infty\left(\mathbb{R}^k; E\right)$ is a smooth function with bounded derivatives, $h^i \in \mathcal{L}^2([0,T];\mathbb{R})$ for $i = 1, \ldots, k$ and $W\left(h^i\right) = \int_0^T h_s^i dW_s$. Define the derivative of ξ as

$$\mathcal{D}_s \xi := \sum_{j=1}^k \partial_{x_i}\psi\left(W\left(h^1\right), \ldots, W\left(h^k\right)\right) h_\theta^j, \qquad 0 \leq s \leq T,$$

and define the norm, for $p \geq 1$

$$||\xi||_{1,p} := \mathbb{E}\left[|\xi|^p + \left(\int_0^T |D_\theta \xi|^2 d\theta\right)^{\frac{p}{2}}\right]^{\frac{1}{p}}, \qquad \xi \in S(E).$$

We define the **Malliavin derivative** as the unique extension of $\mathcal{D}$ to the closure $\mathbb{D}^{1,2}(E) := \overline{\mathcal{S}(E)}$, with $p = 2$.

Definition 1.6.2 (Generalised Derivative)**.** Let E be a Euclidean space. For a function $f\colon \mathbb{R}^n \to E$ we define the **generalised derivative** of f as a function δf such that

$$\lim_{\epsilon \to 0} \int_{\mathbb{R}^n} |\nabla f^\epsilon(x) - \delta f(x)| dx = 0,$$

where for $\epsilon > 0$, f^ϵ is a mollification of f given (for example) by the following

$$\begin{aligned} f^\epsilon(x) &= \eta_\epsilon * f(x) = \int_{\mathbb{R}^n} \eta^\epsilon(y) f(x-y)\, dy, \\ \eta^\epsilon(y) &= \frac{1}{\epsilon}\eta\left(\frac{y}{\epsilon}\right), \\ \eta(z) &= \lambda \exp\left(-\left(1 - |z|^2\right)^{-1}\right) \mathbb{1}_{|x|<1}, \end{aligned}$$

for $x, y, z \in \mathbb{R}^n$. The constant $\lambda \geq 0$ is chosen such that $\int_{\mathbb{R}^n} \eta(z)\, dz = 1$.

Theorem 1.6.3 (Imkeller et al. (2019) Theorem 3.5 and Corollary 3.4)**.** *Consider the following n-dimensional SDE.*

$$X_t = \eta + \int_0^t b_s(X_s)\, ds + \int_0^t \sigma_s(X_s)\, dW_s, \qquad 0 \leq t \leq T,$$

where $\eta \in \mathcal{L}^2(\mathbb{R})$ and the functions $b\colon [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ and $\sigma\colon [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ are deterministic. Assume the following.

1. *The processes $b^0 := b(0)$ and $\sigma^0 := \sigma(0)$ satisfy*

$$\left(\int_0^T |b_t^0| dt\right)^2 + \int_0^T |\sigma_t^0|^2 dt < \infty.$$

2. *b and σ are uniformly Lipschitz in x.*

For $s \in [0,T]$ let δb_s, $\delta\sigma_s$ be the generalised derivatives of b_s and σ_s respectively. Then $X_t \in \mathbb{D}^{1,2}(\mathbb{R}^n)$ for all $t \in [0,T]$ and $\mathcal{D}_s X_t = 0$ for $s > t$, while for $s \leq t$ a version of $\mathcal{D}_s X_t$ is given by

$$\mathcal{D}_s X_t = \sigma_s(X_s) + \int_s^t \delta b_r(X_r)\, \mathcal{D}_s X_r dr + \int_s^t \delta\sigma_r(X_r)\, \mathcal{D}_s X_r dW_r.$$

Theorem 1.6.4 (El Karoui et al. (1997) Theorem 5.3)**.** *Consider the following n-dimensional non-linear BSDE, with 1-dimensional Brownian motion.*

$$Y_t = \xi + \int_t^T f_s(Y_s, Z_s)\, ds - \int_0^T Z_s dW_s, \qquad 0 \leq t \leq T,$$

where $Y, Z \in \mathcal{H}^2(\mathbb{R}^n)$ for some measurable $f\colon \Omega \times [0,T] \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$. Assume the following:

1. *$\xi \in \mathcal{L}^4(\mathbb{R}^n) \cap \mathbb{D}^{1,2}(\mathbb{R}^n)$.*

2. *$f_s(\omega, \cdot, \cdot)$ is continuously differentiable for all $\omega \in \Omega$.*

3. *$f_s(y,z) \in \mathbb{D}^{1,2}(\mathbb{R}^n)$ for any $s \in [0,T]$, $y, z \in \mathbb{R}^n$, and $f^0 := f(0,0)$ satisfies*

$$\mathbb{E}\left[\int_0^T |f_t^0|^4 dt\right] < \infty.$$

4. *The Malliavin Derivatives of ξ and f satisfy*

$$\mathbb{E}\left[\int_0^T |\mathcal{D}_\theta \xi|^2 d\theta\right] < \infty,$$
$$\mathbb{E}\left[\int_0^T |\left(\mathcal{D}_\theta f\right)_t (Y_t, Z_t)|^2 d\theta\right] < \infty \qquad 0 \leq t \leq T.$$

5. *The Malliavin Derivative $\mathcal{D}f$ satisfies, for any $t \in [0,T]$, $s \in [t,T]$ and $y_1, y_2, z_1, z_2 \in \mathbb{R}^n$*

$$\left|\left(D_s f\right)_t (y_1, z_1) - \left(D_s f\right)_t (y_2, z_2)\right| \leq K_{s,t}\left(|y_1 - y_2| + |z_1 - z_2|\right),$$

for any $s \in [0,T]$, where $(K_{s,t})_{t\in[0,T]}$ is a non negative process satisfying

$$\mathbb{E}\left[\int_0^T |K_{s,t}|^4 ds\right] < \infty, \qquad 0 \leq t < T.$$

Then $Y, Z \in \mathcal{L}^2\left(0,T;\mathbb{D}^{1,2}\left(\mathbb{R}^n\right)\right)$ and a version of $\left(\mathcal{D}_s Y, \mathcal{D}_s Z\right)_{s\in[0,T]}$ satisfies $\mathcal{D}_s Y_t = \mathcal{D}_s Z_t = 0$ for $s > t$ and for $s \leq t$

$$\mathcal{D}_s Y_t = \mathcal{D}_s \xi + \int_t^T \left(\partial_y f_r\left(Y_r, Z_r\right)\mathcal{D}_s Y_r + \partial_z f_r\left(Y_r, Z_r\right)\mathcal{D}_s Z_r + \left(\mathcal{D}_s f_r\right)\left(Y_r, Z_r\right)\right) dr - \int_t^T \mathcal{D}_s Z_r dW_r.$$

Furthermore, $\left(\mathcal{D}_t Y_t\right)_{t\in[0,T]}$ is a version of Z.

1.7 Neural Networks and Deep Reinforcement Learning

Before describing the algorithms used to solve this problem in detail, we first look at what we mean by a neural network. We give a brief outline here, see for example Anthony and Bartlett (2009) for an exposition. Suppose we wish to approximate a function $\phi\colon \mathbb{R}^p \to \mathbb{R}^q$ for some $p, q \in \mathbb{N}$. We use the following algorithm to construct the approximating function. This neural network has a predetermined amount of 'layers' $L \in \mathbb{N}$ and 'hidden nodes' $\ell \in \mathbb{N}$. We call these the hyper-parameters of the network in the sequel.

Let $X \in \mathbb{R}^p$ be our input. For $i = 1, \ldots, L$ define the linear function f_i by

$$f_i\left(x\right) = A_i x + b_i.$$

We have $A_1 \in \mathbb{R}^{\ell\times p}$, $A_j \in \mathbb{R}^{\ell\times\ell}$ for $j = 2, \ldots, L-1$ and $A_L \in \mathbb{R}^{q\times\ell}$ for the matrices. For the vectors we have $b_j \in \mathbb{R}^{\ell}$ for $j = 1, \ldots, L-1$ and $b_L \in \mathbb{R}^{q}$. These form the unknowns of our function, and are what we need to choose to correctly approximate our function. We combine these into a set $\theta = (A_i, b_i)_{i=1}^{L}$ of parameters. Informally speaking, the number of hidden nodes ℓ corresponds to us 'expanding' the input into a higher dimensional space. One could think of each node as a piece of information we obtain from the input. The more nodes we have the more information we get, hence the better we can approximate the function. We wish to compose these linear functions in series, but we need a nonlinear function in between them. Let $h \colon \mathbb{R} \to \mathbb{R}$ be such a function that we apply element wise to the output of each linear function. The approximating function is therefore

$$N_\theta (X) \coloneqq f_L \circ h \circ f_{L-1} \circ \ldots \circ f_2 \circ h \circ f_1 (X)$$

In our setting we choose $h(x) = \tanh(x)$. The choice of this function for this scheme is rather arbitrary, as there seems to be no clear preferred function here. A comparison of popular activation functions in provided in Chapter 4. This non-polynomial mapping is essential in ensuring the density of these networks within the space of continuous functions, as is shown in (Leshno et al., 1993, Theorem 1) which formalises the informal argument above about information into a density argument about the space of neural networks. Further more, if ϕ is Lipschitz then we can bound the approximation error in terms of the hyper-parameters of the network (Bach, 2017, Proposition 6). Note that no nonlinear function is applied at the final step, as our activation function is not surjective. The parameter vector $\theta \in \mathbb{R}^{\rho}$, where $\rho = \rho(p, q)$ is defined below, is a parametrisation of the A_i and b_i matrices and is optimised at the training stage of our algorithm. We made no assumptions on the regularity of ϕ, but our approximation is almost everywhere differentiable. In practise this is sufficient for gradient descent algorithms, in which derivatives of this network with respect to its parameters must be taken. We use the `Python` package `Tensorflow` to implement this neural network procedure. This methodology reduces an optimisation over an infinite dimensional space to a finite dimensional one. However, the dimension can still be high. For this neural network, the parameter space dimension is

$$\rho(p, q) = (L-1)\,\ell^2 + \ell\,(L + p + q) + q, \tag{1.13}$$

a quadratic in the number of hidden nodes, and linear in the number of layers and the input dimension. Associated to each neural network N_θ is a loss function $\mathcal{L}(\theta)$ which is a function of the parameters forming the network. Finding the parameters that minimise this function equates to optimising the network itself. In our implementation, we use the ADAM algorithm Kingma and Ba (2014), which is an adaptation of stochastic gradient descent, implemented in `Tensorflow`. We furthermore use mini batches, which means the number of simulations we use to calculate the loss function is small, and batch normalisation, meaning that we scale our input to the neural network Ioffe and Szegedy (2015).

1.8 Stochastic Gradient Descent

Suppose we have a neural network with parameter set $\theta \in \mathbb{R}^\rho$, and a loss function $\mathcal{L}$ to compare it against. The next part of this method is to improve our network in order to optimize $\mathcal{L}$. The typical method to do this is a variant of gradient descent. A vanilla gradient descent algorithm would perform the update

$$\theta_{\text{new}} = \theta_{\text{old}} - \lambda \frac{\partial \mathcal{L}}{\partial \theta}(\theta_{\text{old}})$$

until we achieve convergence. Here, $\lambda > 0$ is the so called learning rate, and is assumed to be sufficiently small. While this is an effective method for convex functions (or other regular functions with unique minimisers), in a more general setting this variable sequence may end up at a local minimum. Also, from a practical perspective, it may be computationally costly to find the gradient with respect to the outcome of every simulation at every step. For these reasons, in practise we use a variant called stochastic gradient descent. In this method, we choose a random batch of sample paths for each update, evaluate $\mathcal{L}$ based on these paths, and then take derivatives. Since less paths are used to compute $\mathcal{L}$, it is faster to run. Also, using less paths to find $\mathcal{L}$ naturally causes a higher variance, which causes high variations in the derivatives that we apply. However, this may be a good thing, as it may allow one to 'jump' from one local minimum to another, better one. Of course this does not completely solve our problem, and finding ways of getting around this is a very active area of research in computer science. There are many other variants of gradient descent, for example there are many methods that look at how fast our variables are approaching a minimum, and adjust the learning rate accordingly, increasing the efficiency of the algorithm. In our implementations, we will use one such algorithm, called `ADAM`, but for simplicity we will use gradient descent when describing our algorithms. The interested reader can find out more about `ADAM` in (Kingma and Ba, 2014). The update formula for the parameters changes more gradually than a stochastic gradient descent algorithm. In the algorithm we wish to minimise the expectation of the random function $\mathcal{L}$ that depends on a parameter set θ and some random noise ω (for example, paths of Brownian motion). At each step we will generate a batch of noisy variables of size B which are used to compute the expectation. In the algorithm we use the notation g^2 to indicate the element-wise square of the vector g.

Algorithm 1: ADAM Algorithm with mini batches (Kingma and Ba, 2014, p. 2)

Input: $\theta^0, \mathcal{L}$

Result: The parameters $\hat{\theta}$ that minimise $\mathbb{E}\left[\mathcal{L}\left(\theta\right)\right]$

Initialise $m^0 = 0, v^0 = 0, j = 0$;

while θ^j *not converged* **do**

  Generate ω_i for $i = 1, \dots B$;

  $L^j\left(\theta\right) = \frac{1}{B}\sum_{i=1}^{B}\mathcal{L}\left(\theta^j; \omega_i\right)$;

  Set $\theta^{j+1}, m^{j+1}, v^{j+1} = \text{ADAM}(\theta^j, L^j, m^j, v^j, j)$;

  $j = j + 1$;

end

return θ^j;

Algorithm 2: ADAM(θ, L, m, v, j)

Input: θ, L, m, v, j (and hyper parameters $\lambda, \beta_1, \beta_2, \epsilon$)

Set $g = \nabla L\left(\theta\right)$;

Set $\tilde{m} = \beta_1 m + \left(1 - \beta_1\right) g$;

Set $\tilde{v} = \beta_2 v + \left(1 - \beta_2\right) g^2$;

Set $\hat{m} = \tilde{m}\left(1 - \beta_1^j\right)^{-1}$;

Set $\hat{v} = \tilde{v}\left(1 - \beta_2^j\right)^{-1}$;

Set $\hat{\theta} = \theta - \lambda\hat{m}\left(\sqrt{\hat{v}} + \epsilon\right)^{-1}$;

return $\hat{\theta}, \hat{m}, \hat{v}$;

The update of θ is based on a bias corrected adapted first moment estimate $\hat{m}$ and a bias corrected raw second moment estimate $\hat{v}$. The convex combination levels are governed by $\beta_1, \beta_2 \in [0, 1)$, for which default values of 0.9 and 0.99 are given respectively. To avoid potential singularities from a small second moment, we include a small ϵ term in the denominator when we divide by it, with default value of 10^{-8}. The learning rate is denoted by λ. This value will change as the iteration increases, and on a model by model basis. The batch size B is typically small, either 32 or 64, allowing for fast derivative computation. Our algorithms will be based on repeated use of algorithm 2.

1.9 Organisation and Contributions

The remainder of this thesis is organised as follows. In Chapter 2 we introduce ML methods for solving stochastic control problems. We develop methods solving the 2BSDEs associated to the HJB equation, and the adjoint equation associated to the SMP. Given solutions to these BSDEs we use optimality conditions to define a scheme for finding the optimal control using neural networks. We apply convex duality methods to develop algorithms for solving the dual problem, as well as the primal problem.

In Chapter 3 we analyse the convergence of the PPGM algorithm for stochastic control problems, which can be seen as a more theoretical version of our primal DC2BSDE algorithm. We study the application of PPGM to the LQ problem, and develop sufficient conditions under which convergence can be established. We perform separate analysis for the unconstrained and constrained problems, as the unconstrained case can be simplified to a study of ODEs, while the constrained case requires Malliavin calculus methods.

In Chapter 4 we perform numerical analysis of all the algorithms developed in the previous sections. We apply the algorithms to a range of problems, and numerical results show high accuracy and low runtime of the proposed methods. Our methods are additionally compared to existing methods in the literature.

The main contributions of this thesis are the following.

- We develop novel machine learning algorithms based on 2BSDE representations of the HJB equations, and optimality conditions arising from the SMP. To find the derivative of the value function with respect to an approximate control, we solve a series of BSDEs, which was new in the literature at the time.
- We use methods of duality theory to develop algorithms that solve both the primal and dual versions of particular control problems. Convex duality methods allow us to deter-

mine tight bounds of the value function, and can be applied to non-Markovian control problems.

- We analyse the PPGM framework for the LQ problem, and give sufficient conditions of linear convergence in a general setting of constrained optimisation, where the diffusion of the state is explicitly dependent on the control process. To do this we study the regularity of the adjoint BSDE pair using Malliavin calculus.

- Numerical results suggest our algorithms are favourable comparable with the existing literature in terms of run time and accuracy.

Chapter 2

Duality and Machine Learning Algorithms for Stochastic Control Problems

In this chapter we propose algorithms that combine modern machine learning practises with theoretical stochastic control principles to solve a range of control problems with high accuracy, scalability with dimension, and low computational cost. We propose deep learning methods to solve the utility maximisation problem, as an application of a more general stochastic control solver, in a wide range of markets with arbitrary convex control constraints. Dynamic portfolio optimisation has been extensively studied within the field of mathematical finance, see Karatzas and Shreve (1998); Pham (2009) for exposition.

There is a natural crossover between machine learning and stochastic control problems, as they both involve searching for features within a data set. For stochastic control the data set is a diffusion process that is influenced by a controlling process, and some underlying Brownian motion. The controlling process is chosen with the aim to optimise some cost function of the state and control process along a time horizon. Often the optimal control is a function of the underlying state process, a so-called Markovian control. The concept of a Markovian control is very similar to methods of deep learning. In this setting the control is the output of a neural network, which is a composition of non-linear and linear functions, parametrised by the linear and affine coefficients of the linear functions. This neural network takes the dataset as input, and the parameter set will be chosen to minimise some loss function, that takes the dataset and output of the neural network as input. It is in this area of crossover that this chapter resides.

A typical drawback of numerical methods for stochastic control problems is that their complexity increases dramatically with increases in the state and control dimensions. Often some grid method or exhaustive search of the underlying state space is required to find optimal trajectories, which is computationally inefficient for high dimensional state spaces. It is claimed that machine learning methods provide a solution for this dimensionality problem Grohs et al. (2018). Numerical results show efficient solving of PDEs with the state space dimension in the thousands Beck et al. (2019a,b); E et al. (2017). These PDE solvers form the basis of our generic stochastic control problem solver, which is the first algorithm presented in this chapter. The problems are transformed into forward-backward stochastic differential equations (FBSDEs) using the Feynman-Kac formula and the Stochastic Maximum Principle (SMP). The PDE that we solve is the Hamilton-Jacobi-Bellman (HJB) equation, a highly nonlinear PDE that is often impossible to solve analytically, or even numerically if the dimensionality is high. This chapter introduces a control variable into the PDE, that is constrained to some convex set. This drastically increases the complexity and takes the problem out of the scope of Beck et al. (2019b); E et al. (2017) unless the control can be found analytically. We show how machine learning can overcome this difficulty and produce highly accurate value approximations with comparatively low runtime.

The FBSDE based methods developed in this chapter are direct extension of the first and second order deep BSDE solvers introduced in Beck et al. (2019b); E et al. (2017). In these papers, a system of FBSDEs is solved by simulating all processes in the forward direction, driven by a neural network process which is optimised to ensure the terminal condition holds. The extension offered by this chapter is an additional driving neural network process, representing a control process. The convenience of adapting this BSDE method to stochastic control problems is that the control process can be optimised using the value function approximations implied by the 2BSDE solver, via the Hamiltonian of the system. In the 2BSDE solver of Beck et al. (2019b), there are two unknown process that must be found with neural networks. This suggests that our method has three unknowns, but we can use the structure of the value problem to derive one of these processes and therefore introduce control without increasing the number of unknown processes.

There are some papers that describe generic algorithms for solving stochastic control problems in the literature. Han and E (2016) directly maximises the gains function at the terminal time with machine learning and is applicable to many problems. However, the computation costs may be high, as one introduces a discretisation with many time steps or a high dimensional control space, and it only outputs the value at time 0. Huré et al. (2021) finds a local approximation of the value function with machine learning. However, it only holds for Markovian problems and the iterative nature of this value function approximation may lead to error propagation. Instead of directly optimising the value function, the first method presented in this chapter maximises the Hamiltonian associated with the system. There are two key benefits of doing this. Firstly, it reconciles the potential errors of Han and E (2016); Huré et al. (2021) where the algorithm gets stuck at a local optimiser. To find the value function (given a control) we ensure that the terminal condition of the HJB equation holds, which involves minimising the least squares error of the associated processes at the terminal time. The global minimum of this error is zero by construction, ensuring that we can directly assess the quality of our approximation for any generic control problem. Secondly, the control is optimised using the Hamiltonian of the system, which is a function of only what is known at one time. This optimisation is faster than back propagating through a discretisation of the processes to compute the derivatives as in Han and E (2016). This is even more important when the control space is high dimensional, which is normally true in portfolio optimisation problems. To implement this Hamiltonian maximisation procedure, one needs to know derivatives of the value function which can be computed accurately with deep learning based on the 2BSDE method in Beck et al. (2019b).

There are a number of papers in the literature that utilise the deep BSDE methods of Beck et al. (2019b); E et al. (2017). For example, Exarchos and Theodorou (2018); Pereira et al. (2019); Wang et al. (2019) first find an explicit formula for the optimal control in terms of the value function and its derivatives and then substitute this control into the HJB equation and solve the resulting PDE with the deep 2BSDE method. However, these algorithms cannot be applied in full generality like our method. Exarchos and Theodorou (2018); Pereira et al. (2019) are reliant on the diffusion of the state process being independent of the control, which leads to a semi-linear PDE and a representation with a first order BSDE. This framework is inappropriate for portfolio selection problems as the control naturally appears in the diffusion term. Wang et al. (2019) alleviates this restriction by deriving an explicit formula for the optimal control that is required to be unconstrained. Our approach is more general in the sense that it does not require any analytical solutions. In the literature these other algorithms are called "deep 2BSDE controllers" since the BSDE solutions are used to form the control. Our algorithm is called the deep controlled 2BSDE method as each control and state pair defines its own 2BSDE which we try to solve. Another paper developed at the same time as ours, Ji et al. (2022) develops primal algorithms utilising BSDE solutions and the SMP in a similar way to ours. They propose 3 algorithms, one of which being similar to the primal DC2BSDE method described here. The key difference of our work is the introduction of duality, which allows us to develop alternative algorithms for solving the control problem.

The numerical method of solving these control problems fits in with a more general framework of an actor-critic model, where the 'actor' tries to find a control policy for a given problem (often with some random element to force 'exploration' of the state space), and the 'critic' tries to find the value associated to this policy, see for example Jia and Zhou (2022); Wang et al. (2020). In our setting, the 'critic' finds the solutions to a BSDE to determine information about the value function derivatives evaluated along paths of a particular state process, and the 'actor' uses this information to update the control.

A primal-dual method is introduced in Henry-Labordere (2017) for solving semi-linear PDEs. The deep 2BSDE method is used to find a lower bound for the solution. An upper bound is found by solving a maximisation control problem on the path space, that is derived using the Legendre transform. The approach in Henry-Labordere (2017) is not directly related to our notion of convex duality using martingale methods. There are also some papers on applications of deep neural networks in mathematical finance, including optimal stopping (Becker et al. (2019)), asset pricing with transaction cost (Gonon et al. (2020)), and optimal transport (Eckstein and Kupper (2019)). Furthermore, Buehler et al. (2019); De Spiegeleer et al. (2018); Liu et al. (2019) consider the use of machine learning for model calibration, hedging and pricing, and are not directly related to our work.

The utility maximisation problem involves maximising concave functions, so we can naturally apply convex duality methods. Duality is a powerful tool that can be used in complete or incomplete markets with a range of convex constraints on the control process. The unique aspect of our work is the combination of deep learning methodology with convex duality. The benefits of introducing duality are threefold. Firstly, the dual problem is a minimisation problem, and solving it independently of the primal problem naturally leads to an upper bound of the value function. Sometimes the dual problem is easier to solve, since primal control constraints are characterised by penalties in the dual state process, leading the dual problem to be potentially unconstrained. Secondly, the BSDEs associated to the dual problem can be related to that of the primal problem, giving us the ability to compare our numerical results and check their accuracy. Again, as per the Hamiltonian maximisation, this benefits us since we know the error between the dual and primal processes should be zero, which allows us to easily measure the quality of our results. Additionally, solving the primal and dual problem simultaneously allows us to generate upper and lower bounds of the value function, giving us a confidence interval which is often missing in other methods in the literature. In some cases may skip one of the problems if they are difficult to solve, as solving the dual problem automatically gives us the solution to the primal problem and vice versa. Thirdly, inherent in the dual formulation are certain optimality conditions that are derived from the SMP. These conditions can be used separately from the HJB equation to find the optimal control. We use these conditions as the basis for our second algorithm, called the deep SMP method, which is valid for non-Markovian control problems with convex structure. This greatly increases the range of utility maximisation problems that can be solved using the methods of this chapter. In a Markovian setting we can compare the two algorithms presented in the chapter. It should be noted here that results for dynamical programming principles, under which our 2BSDE scheme is formulated, exist in the literature for non-Markovian problems. Non-Markovian optimal stopping problems and their 2BSDE counterparts are studied in Karoui and Tan (2013); Possamaï et al. (2018). The problem is reformulated into an optimisation in the space of probability measures, and the canonical probability space is expanded to allow a dynamical programming principle and a characterising path dependent PDE to be established, see Zhang (2017) for an exposition. However, it is not clear how a machine learning based solver could be used in this abstract setting.

The convex duality method works for complete and incomplete markets. A common incomplete market model is the Heston stochastic volatility model in which the underlying variance of the stock follows some mean-reversion SDE with its own source of randomness. This model may be seen as a 2-dimensional stock market, where only one asset is traded. This framework allows us to use our deep learning method to solve it. We compare our results to Ma et al. (2020) in which tight bounds of the value function are derived for an unconstrained optimisation problem. Ma et al. (2020) is reliant on a convenient guess of the dual control, so is not applicable to other methods where such a guess is not available, unlike our method. Another multidimensional model we consider is a path-dependent volatility model.

The material in this chapter is adapted from a paper Davey and Zheng (2022). The main contributions of this paper are the following. In the deep controlled 2BSDE algorithm, we present a Markovian stochastic control problem solver that combines the HJB equation with the deep 2BSDE method, but without the need for an analytical formula for the optimal control. We apply this algorithm to both the primal and dual problems which give us tight bounds on the value function. In the deep SMP method, we exploit the dual formulation of the utility maximisation problem to create a novel algorithm for non-Markovian optimisation problems using deep learning. Instead of dealing with the gains function, we solve the adjoint BSDE and use machine learning to satisfy the dual optimality conditions. To the best of our knowledge, we are the first to use this methodology in the literature.

The chapter is outlined as follows. Section 2.1 introduces the primal DC2BSDE method for the primal problem. Section 2.2 describes the utility maximisation problem as an example of the DC2BSDE method, and a motivational example for the proceeding dual approaches. Section 2.3.1 introduces the dual DC2BSDE method and the Deep SMP Method for the utility maximisation problem. Section 2.4 extends our example to the investment-consumption problem with randomly terminating income.

2.1 The Deep Controlled 2BSDE Method

The machine learning methodology will attempt to find the optimal control and value function using the HJB equation (1.7) (in particular, its BSDE representation from Feymann-Kac), the adjoint equation (1.10) and the Hamiltonian condition (1.12). We look to solve the BSDEs, which can be coupled into a 2BSDE, with machine learning techniques in the spirit of Beck et al. (2019b) (the algorithm presented therein is named the "Deep 2BSDE" method, thus inspiring our adapted Deep Controlled 2BSDE method). The trick here is to simulate all processes in the forward direction, and use the terminal condition as a loss function. The control process u is designed as a neural network, taking in the state process and time as input, which means that we are searching for a Markovian control. We then approximate $\partial_{xx} v\left(t, X_t^u\right)$ to be an additional neural network. The loss functions are determined by satisfaction of the terminal conditions for the BSDEs, and the Hamiltonian minimisation condition (1.12).

Remark 1. In this algorithm we propose a control u of the form $u_t = N_\theta\left(t, X_t^u\right)$, for some parameter set θ. One may then ask if this control is indeed admissible. It can be shown that a neural network is a Lipschitz function as long as the activation function is Lipschitz (though a global Lipschitz constant for every neural network in the training process may not be established), and therefore the required integrability properties of the state process and cost function of the control problem are satisfied, so we have $u \in \mathcal{U}$.

As motivation, suppose $v \in \mathcal{C}^{1,3}\left([0,T) \times \mathbb{R}^n\right) \cup \mathcal{C}^0\left([0,T] \times \mathbb{R}^n\right)$ (we do not need to assume this to solve the BSDEs in the sequel, they may simply represent viscosity solutions to the HJB if this condition is not satisfied). Let u^* be the optimal control, with associated controlled diffusion X^{u^*}. Define the processes, for $t \in [0,T]$,

$$\begin{aligned} V_t^{u^*} &= v\left(t, X_t^{u^*}\right), \\ Y_t^{u^*} &= \partial_x v\left(t, X_t^{u^*}\right), \\ \Gamma_t^{u^*} &= \partial_{xx}^2 v\left(t, X_t^{u^*}\right), \\ Z_t^{u^*} &= \partial_{xx}^2 v\left(t, X_t^{u^*}\right) \sigma_t\left(X_t^{u^*}, u_t^*\right). \end{aligned}$$

Applying Ito's Lemma to V^{u^*}, and using the HJB equation (1.7), we have the BSDE

$$\begin{aligned} dV_t^{u^*} &= -f_t\left(X_t^{u^*}, u_t^*\right) dt + Y_t^{u^*} dW_t, \qquad 0 \le t \le T, \\ dV_T^{u^*} &= g\left(X_T^{u^*}\right). \end{aligned} \tag{2.1}$$

Furthermore, by the SMP (Theorem 1.5.1), we have the BSDE

$$\begin{aligned} dY_t^{u^*} &= -\partial_x H_t\left(X_t^{u^*}, u_t^*, Y_t^{u^*}, Z_t^{u^*}\right) dt + Z_t^{u^*} dW_t, \qquad 0 \le t \le T \\ dY_T^{u^*} &= \nabla g\left(X_T^{u^*}\right). \end{aligned} \tag{2.2}$$

In practise, we do not know what u^* is. Instead, we start at some $u \in \mathcal{U}$. Then we solve the

2BSDE, for $t \in [0,T]$,

$$
\begin{aligned}
dV_t^u &= -f_t\left(X_t^u, u_t\right)dt + Y_t^u dW_t,\\
dV_T^u &= g\left(X_T^u\right)\\
dY_t^u &= -\partial_x H_t\left(X_t^u, u_t, Y_t^u, Z_t^u\right)dt + Z_t^u dW_t,\\
dY_T^u &= \nabla g\left(X_T^u\right).
\end{aligned}
\tag{2.3}
$$

and use (1.12) as a loss function.

Remark 2. Again, a neural network is a Lipschitz function, as long as the activation function is Lipschitz. Therefore the 2BSDE (2.3) admits a unique strong solution $(V^u, Y^u, Z^u) \in \mathcal{S}^2(\mathbb{R}) \times \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{H}^2\left(\mathbb{R}^{n\times d}\right)$ if u is chosen to be a Markovian neural network control.

Now we describe the details of the algorithm to solve the control problem. We introduce parameters $\theta_i^0 \in \mathbb{R}^{\rho(n,m)}$, $\lambda_i^0 \in \mathbb{R}^{\rho\left(n,n^2\right)}$, $v_0^0 \in \mathbb{R}$ and $y_0^0 \in \mathbb{R}^n$, then use the following Euler-Maruyama Scheme. Set $X_0^0 = x_0$, $V_0^0 = v_0^0$ and $Y_0^0 = y_0^0$ then for $i = 0, \dots N-1$ let $u_i^0 = N_{\theta_i^0}\left(X_i^0\right)$, $\Gamma_i^0 = N_{\lambda_i^0}\left(X_i^0\right)$ and

$$
\begin{aligned}
X_{i+1}^0 &= X_i^0 + \left(t_{i+1} - t_i\right) b_{t_i}\left(X_i^0, u_i^0\right) + \sigma_{t_i}\left(X_i^0, u_i^0\right)\Delta W_i\\
V_{i+1}^0 &= V_i^0 - \left(t_{i+1} - t_i\right) f_{t_i}\left(X_i^0, u_i^0\right) + \left(Y_i^0\right)^\top \sigma_{t_i}\left(X_i^0, u_i^0\right)\Delta W_i\\
Y_{i+1}^0 &= Y_i^0 - \left(t_{i+1} - t_i\right)\partial_x H_{t_i}\left(X_i^0, u_i^0, Y_i^0, \Gamma_i^0 \sigma_{t_i}\left(X_i^0, u_i\right)\right) + \Gamma_i^0 \sigma_{t_i}\left(X_i^0, u_i\right)\Delta W_i,
\end{aligned}
\tag{2.4}
$$

where ΔW_i is a multivariate normal $\mathcal{N}_d\left(0, \left(t_{i+1} - t_i\right)\mathbb{1}_d\right)$ random variable. We have used minimal notation for these processes for clarity, but one should remember that these explicitly depend on the choice of parameters λ_i^0, θ_i^0 and the initial points v_0^0, z_0^0.

Remark 3. We use a different neural network at every time step, as opposed to a single network that takes in state and time as inputs. While this increases the number of parameters (assuming all networks use the same number of layers and hidden nodes), it also decreases the run time. This is because in the back propagation step, extra time is required to take derivatives of the network parameters, since they are involved in every time step. Even when using a high number of time steps, the increase in time outweighs the penalty of having more networks which leads to more memory usage.

Once we have repeated this iteration $N-1$ times, we arrive at time $t_N = T$ but realise that we have not satisfied the terminal conditions. We want to choose the neural networks and initial start points to reduce the expected error, which amounts to minimising the loss function

$$\mathcal{L}_1\left(\Lambda^0\right) := \mathbb{E}\left[\left|V_N^0 - g\left(X_N^0\right)\right|^2 + \gamma_1\left|Y_N^0 - \nabla g\left(X_N^0\right)\right|^2\right], \tag{2.5}$$

where $\Lambda^0 \in \mathbb{R}^{1+n+\rho\left(n,n^2\right)}$ is a vector representation of $\{v_0^0, y_0^0\} \cup (\lambda_i^0)_{i=0}^{N-1}$. This is referred to as the BSDE loss function in the sequel. When discussing numerical implementation here and in the sequel, $\mathbb{E}$ denotes the sample average.

In addition, for each time step the arbitrary control choice u^0 is not optimal, and may not be in the constraint space U, so we must also maximise the loss functions

$$\mathcal{L}_2\left(\theta_i^0, i\right) := \mathbb{E}\left[F_{t_i}\left(X_i^0, u_i^0, Y_i^0, \Gamma_i^0\right) + \gamma_2 \text{dist}\left(u_i^0, U\right)\right], \tag{2.6}$$

for $i = 0, \ldots, N-1$, where dist is the distance function

$$\text{dist}\left(u, U\right) = \sup_{z \in \mathcal{U}} |z - u|^2.$$

Where $|\cdot|$ denotes the euclidean norm (we use the square as it is unnecessary to take the square root of this vector, as would be done using the norm). The function $\mathcal{L}_2$ is referred to as the control loss function in the sequel. Again, we evaluate a sample average of this loss function. The coefficients $\gamma_1, \gamma_2 > 0$ are some tuning parameters that can be chosen. In our case, we choose $\gamma_1 = 0.5$, and $\gamma_2 = 100$. We proceed to optimise these loss functions in turn, starting with our randomly initialised 0-superscript parameters. Now suppose we have just finished step $j \in \mathbb{N}$.

The first sub-step is to improve the terminal condition loss. We generate V_N^j, Y_N^j and X_N^j using Λ^j and θ_i^j for $i = 1, \ldots, N-1$, which are carried forward from the previous step. We then make an improvement using one step of the ADAM algorithm, against the loss function $\mathcal{L}_1$, resulting in the updated parameters Λ^{j+1}. We also keep track of the moment estimates that are present in the algorithm as in Kingma and Ba (2014).

The second half of the iteration step is to improve the control process. We can first simulate the processes u^j and X^j using our previous parameter set $\Theta^j = \left(\theta_i^j\right)_{i=0}^{N-1}$, then simulate Y^j and Γ^j using the updated parameters Λ^{j+1}, and finally make the following improvement using one step of the ADAM algorithm. For each $i = 0, \ldots, N-1$ we update θ_i^j using the ADAM algorithm against $\mathcal{L}_2(\cdot, i)$, resulting in θ_i^{j+1}. We track the moment estimates and move to step $j+2$ and repeat this process. We do this for each path of a batch of simulated paths of size $B \in \mathbb{N}$.

These two steps can be repeated until the new parameters are sufficiently close to the old parameters after an update, at which point we deem the algorithm to have converged. We find that making small increments at each step, ensuring that the learning rate required for the ADAM algorithm is smaller for the control step than the BSDE step, we have good convergence results.

Remark 4. At each time step we build 2 neural networks, which have 4 layers, with the two hidden layers consisting of $n+10$ nodes. The parameters are optimised using the ADAM algorithm, using a batch size of 64 Brownian paths. We choose to initialise each parameter near 0, though a prior guess may be used, and the rate of convergence is naturally much higher when the initial guess is closer to the true value. Substituting $p = n$, $L = 4$ and $l = n + 10$ into (1.13) leads to a parameter set of size $\rho(n, q) = 4n^2 + 74n + 340 + q(n + 11)$, where q is the output dimension of the network. The number of neural network parameters is therefore at most cubic in the state dimension and linear in the control dimension.

Remark 5. We use a different loss function for the BSDE and control networks in the algorithm. Another option would be combining these loss functions into one single loss function that both networks try to minimise. However, this would be slower as the back propagation step would take longer, particularly for the control network. It may also lead to false solutions, for example minimising $\mathcal{L}_1$ by making both V_N^j and $g\left(X_N^j\right)$ close to zero.

Algorithm 3 outlines this scheme explicitly. When we use the notation ADAM(), we refer to one step of the ADAM Algorithm (Algorithm 1). The output is given as the value at time 0, but a byproduct of this scheme is pathwise evaluations of the value function and its derivatives, as well as the optimal control. This algorithm terminated when Λ^j, Θ^j are deemed to have 'converged', this could be once some maximum number of iterations are hit, or if the loss functions do not change much with respect to iteration.

An implementation of this scheme (and the dual / SMP adaptations in the sequel) in Python with Tensorflow 1 can be found at `https://github.com/Ashley-Davey/ML-For-Stochastic-Control`.

Algorithm 3: Primal Deep Controlled 2BSDE

Input: $x_0 \in \mathbb{R}^n$

Result: $v(0, x_0)$

Initialise $V_0^0 \sim \text{Unif}([-0.1, 0.1])$ and $Y_0^0 \sim \text{Unif}([-0.1, 0.1]^n)$;

for $i = 0, 1, 2, \ldots, N-1$ **do**

  Initialise $\theta_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho(n,m)}\right)$, $\lambda_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho\left(n,n^2\right)}\right)$;

end

Set $\Lambda^0 = \{V_0^0, Y_0^0\} \cup \{\lambda_i^0\}_{i=0}^{N-1}$;

Set $\Lambda^0 = \{\theta_i^0\}_{i=0}^{N-1}$;

Initialise $m^0 = v^0 = 0$, $j = 0$ and $m_i^0 = v_i^0 = 0$ for $i = 0, \ldots, N-1$;

while Λ^j, Θ^j *not converged* **do**

  Generate ΔW_i^k for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  Generate $X^{j,k}, V^{j,k}, Y^{j,k}, u^{j,k}, \Gamma^{j,k}$ using (2.4) and (Λ^j, Θ^j) for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  Set $\mathcal{L}_1^j(\Lambda^j) = \frac{1}{B}\sum_{k=1}^B \left|V_N^{j,k} - g\left(X_N^{j,k}\right)\right|^2 + \gamma_1 \left|Y_N^{j,k} - \nabla g\left(X_N^{j,k}\right)\right|^2$;

  Set $\Lambda^{j+1}, m^{j+1}, v^{j+1} = \text{ADAM}(\Lambda^j, \mathcal{L}_1^j, m^j, v^j, j)$;

  Regenerate $X^{j,k}, V^{j,k}, Y^{j,k}, u^{j,k}, \Gamma^{j,k}$ using (2.4) and $(\Lambda^{j+1}, \Theta^j)$ for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  for $i = 0, 1, 2, \ldots, N-1$ **do**

    Set $\mathcal{L}_2^j\left(\theta_i^j, i\right) = \frac{1}{B}\sum_{k=1}^B F_{t_i}\left(X_i^{j,k}, u_i^{j,k}, Y_i^{j,k}, \Gamma_i^{j,k}\right) + \gamma_2 \text{dist}(u_i^{j,k}, U)$;

    Set $\theta_i^{j+1}, m_i^{j+1}, v_i^{j+1} = \text{ADAM}(\theta_i^j, \mathcal{L}_2^j(\cdot, i), m_i^j, v_i^j, j)$;

  end

  Set $\Theta^{j+1} = \{\theta_i^{j+1}\}_{i=0}^{N-1}$;

  Set $j = j + 1$;

end

Output V_0^j;

2.2 Utility Maximisation

In this section we describe the standard portfolio optimisation problem.

2.2.1 Primal Problem

Let $W = (W_t)_{0 \le t \le T}$ be a standard m dimensional Brownian motion on the natural filtered probability space $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0 \le t \le T}, \mathbb{P}\right)$ augmented with all $\mathbb{P}$-null sets for some fixed finite terminal horizon $T > 0$. Consider an $m+1$ dimensional stock market, consisting of a risk-free bond with interest rate r_t, $t \in [0,T]$, and a collection S of risky stocks defined by $S_0 = 1$ and

$$dS_t = \operatorname{diag}(S_t)\left(\mu_t dt + \Sigma_t dW_t\right), \qquad 0 \le t \le T, \tag{2.7}$$

where $\operatorname{diag}(S_t)$ is an $m \times m$ matrix with diagonal elements from S_t and all other elements zero, and r, μ and Σ are uniformly bounded, progressively measurable processes valued in $\mathbb{R}$, $\mathbb{R}^m$ and $\mathbb{R}^{m\times m}$, respectively, with μ_t and Σ_t representing the drifts and volatilities of the m stock prices at time $t \in [0,T]$, respectively. The matrix Σ_t is invertible and Σ_t^{-1} is uniformly bounded for all $t \in [0,T]$. An agent invests a proportion π_t of the wealth X_t^π in the risky stocks and the rest in the risk-free bond at time $t \in [0,T]$. The 1-dimensional wealth process X evolves as the geometric Brownian motion

$$dX_t^\pi = X_t^\pi \left(r_t + \pi_t^\top \left(\mu_t - r_t \mathbb{1}_m\right)\right) dt + X_t^\pi \pi_t^\top \Sigma_t dW_t, \qquad 0 \le t \le T, \tag{2.8}$$

with $X_t^\pi = x_0$, where $\mathbb{1}_m \in \mathbb{R}^m$ is a vector of ones. We call π an admissible control if π is a progressively measurable process valued in $K \subset \mathbb{R}^m$, a non-empty closed convex set, almost surely for almost every $t \in [0,T]$, and there exists a unique strong solution X^π to the SDE (2.8). The set of all admissible controls π is denoted by $\mathcal{K}$. The set K corresponds to trading restrictions or incompleteness of the Market. For example, in one of our examples we take $K = [0,\infty)^m$ to be only non-negative controls. As X^π is also non-negative (as long as $x_0 > 0$), this constraint corresponds to the rule of no short-selling.

The investor wishes to maximise the expected utility of the terminal wealth, that is, to find

$$V := \sup_{\pi \in \mathcal{K}} \mathbb{E}\left[U\left(X_T^\pi\right)\right], \tag{2.9}$$

where $U\colon [0,\infty) \to \mathbb{R}$ is a utility function that is twice continuously differentiable, strictly increasing, strictly concave and satisfies $U'(0) = \infty$ and $U'(\infty) = 0$. We say that state and control processes X^{π^*}, π^* are optimal if $V = \mathbb{E}\left[U\left(X_T^{\pi^*}\right)\right]$.

We first suppose that the coefficients r, μ, Σ are deterministic. For our machine learning framework, we have the process $X^u = X^\pi$ defined by (1.1), controlled by $u = \pi$, and the primal 2BSDE system (2.3) where we take $g(x) = U(x)\mathbb{1}_{x>0}$ for $x \in \mathbb{R}$, $f = 0$ and define the drift and diffusion functions by

$$b_t(x,\pi) = x\left(r_t + \pi^\top(\mu_t - r_t\mathbb{1}_m)\right),$$
$$\sigma_t(x,\pi) = x\pi^\top\Sigma_t,$$

for $(t,x,\pi) \in [0,T] \times \mathbb{R} \times \mathbb{R}^m$. As this is a maximisation problem, we alter the loss function (2.6) for the control slightly. We set

$$\mathcal{L}_2\left(\theta_i^0, i\right) := \mathbb{E}\left[-F_{t_i}\left(X_i^0, u_i^0, Y_i^0, \Gamma_i^0\right) + \gamma_2 \mathrm{dist}\left(u_i^0, K\right)\right],$$

for $i = 1, \ldots, N$ and $\theta_i \in \mathbb{R}^{\rho(1,m)}$.

Example 2.2.1. Now, in the general case we may not have deterministic coefficients. Consider for example the Heston problem. In this model, our agent invests in a 2-dimensional stock market, out of which one asset S is traded, and is defined by

$$dS_t = S_t(r + \theta\Sigma_t)\,dt + S_t\sqrt{\Sigma_t}dW_t^S, \qquad 0 \le t \le T.$$

The constants $r, \theta > 0$ represent the risk-free rate and market price of risk respectively. The volatility process Σ has the dynamics

$$d\Sigma_t = \kappa\left(\bar{\Sigma} - \Sigma_t\right)dt + \xi\sqrt{\Sigma_t}dW_t^\Sigma, \qquad 0 \le t \le T.$$

The constant $\kappa > 0$ represents the speed at which the process reverts to the long-time average volatility, given by $\bar{\Sigma} > 0$. The constant $\xi > 0$ is a variance parameter. The Brownian motions W^S and W^Σ are correlated with correlation parameter $\rho \in [-1,1]$. When the agent invests a proportion π of their wealth X^π in this market, their wealth process (2.8) evolves as

$$dX_t^\pi = X_t^\pi(r + \pi_t\theta\Sigma_t)\,dt + X_t^\pi\pi_t\sqrt{\Sigma_t}dW_t^S, \qquad 0 \le t \le T.$$

To transform this to a Markovian setting, the 2-dimensional process (X^π, Σ) is the state process controlled by the $K \subset \mathbb{R}$ valued process π. Consider the maximum utility problem given by, for $(t, x, \Sigma) \in [0, T] \times \mathbb{R}^2$

$$u(t, x, \Sigma) = \sup_{\pi \in \mathcal{K}} \mathbb{E}\left[U\left(X_T^\pi\right) \mid \left(X_0^\pi, \Sigma_0\right) = (x, \Sigma)\right], \tag{2.10}$$

for some utility function U. This setup fits in the framework of (1.3) with $d = 2$, $n = 2$, $m = 1$ and $g(x, \Sigma) := U(x) \mathbb{1}_{x>0}$ for $(x, \Sigma) \in \mathbb{R}^2$ and $f = 0$. The drift and diffusion functions are given by

$$\begin{aligned} b_t(x, \Sigma, \pi) &= \left(x(r + \pi\theta\Sigma), \kappa\left(\bar{\Sigma} - \Sigma\right)\right)^\top, \\ \sigma_t(x, \Sigma, \pi) &= \begin{pmatrix} x\pi\sqrt{\Sigma} & 0 \\ \rho\xi\sqrt{\Sigma} & \sqrt{1-\rho^2}\xi\sqrt{\Sigma} \end{pmatrix}, \end{aligned}$$

for $(t, x, \Sigma) \in [0, T] \times \mathbb{R}^2$. It is a usual assumption that the drift and diffusion functions are Lipschitz functions of the state and control variable, but this is not the case here due to the square root appearing in the diffusion. This will invalidate current theoretical convergence results.

2.2.2 Dual Problem

We now define the dual problem for the utility maximization problem (2.9), see (Li and Zheng, 2018, Section 2) for more details. The basis for a dual problem relies on the Legendre-Fenchel Transform. Define

$$\tilde{U}(\tilde{x}) = \sup_{x>0}\left\{U(x) - x\tilde{x}\right\}, \quad \tilde{x} > 0.$$

The concavity and differentiability of U ensure that $\tilde{U}$ is well defined and is twice continuously differentiable, strictly decreasing and strictly convex on $(0, \infty)$. The $\mathbb{R}$-valued dual process $\tilde{X}^{\tilde{x},\tilde{\pi}}$ is defined by

$$d\tilde{X}_t^{\tilde{x},\tilde{\pi}} = -\tilde{X}_t^{\tilde{x},\tilde{\pi}}\left(r_t + \delta_K\left(\tilde{\pi}_t\right)\right)dt - \tilde{X}_t^{\tilde{x},\tilde{\pi}}\left(\theta_t + \Sigma_t^{-1}\tilde{\pi}_t\right)^\top dW_t, \qquad 0 \le t \le T, \tag{2.11}$$

with $\tilde{X}_0^{\tilde{x},\tilde{\pi}} = \tilde{x}$, where $\tilde{x}$ is non-negative, $\theta_t := \Sigma_t^{-1}\left(\mu_t - r_t\mathbb{1}_m\right)$ is the market price of risk, and $\tilde{\pi}$ is a $\mathbb{R}^m$-valued dual control process which is progressively measurable and ensures there exists a unique strong solution $\tilde{X}^{\tilde{x},\tilde{\pi}} \in \mathcal{S}^2(\mathbb{R})$ to the SDE (2.44). The function δ_K is the support function of $-K$, defined by

$$\delta_K(z) = \sup_{\pi \in \mathcal{K}} \left\{ -\pi^\top z \right\}, \qquad z \in \mathbb{R}^m.$$

We choose this specific dual problem, as it can be shown that the processes $X^\pi \tilde{X}^{\tilde{x},\tilde{\pi}}$ is a supermartingale for any choice of π, $\tilde{\pi}$ and $\tilde{x}$ and we have the following weak duality relation (Li and Zheng, 2018, p. 5)

$$V \leq \tilde{V} := \inf_{\tilde{x}>0} \left\{ \inf_{\tilde{\pi} \in \mathbb{R}^m} \left\{ \mathbb{E}\left[\tilde{U}\left(\tilde{X}_T^{\tilde{x},\tilde{\pi}} \right) \,\middle|\, \tilde{X}_0^{\tilde{x},\tilde{\pi}} = \tilde{x} \right] \right\} + x_0 \tilde{x} \right\}. \tag{2.12}$$

Solving the dual problem would give us an upper bound of the value function, that can be combined with a lower bound by the primal problem to produce a confidence interval for the value function. If we have an equality in (2.12), then we can find the value function by alternatively solving the dual problem.

2.3 Duality Methods

In this section we introduce deep learning algorithms that use duality. For the dual DC2BSDE algorithm, we consider a more general setting for which the utility dual problem (2.12) is a special case.

We may define the following dual value function

$$\tilde{u}(0, \tilde{x}) = \inf_{x \in \mathbb{R}^n} \left\{ u(0, x) + x^\top \tilde{x} \right\}, \qquad \tilde{x} \in \mathbb{R}^n.$$

This formulation can be reversed, so if we know $\tilde{u}(0, \cdot)$, then we can find $u(0, \cdot)$ via

$$u(0, x) = \sup_{\tilde{x} \in \mathbb{R}^n} \left\{ \tilde{u}(0, \tilde{x}) - x^\top \tilde{x} \right\}, \qquad x \in \mathbb{R}^n. \tag{2.13}$$

We can describe $\tilde{u}$ as a value function for a particular stochastic control problem that can be solved using the method of Section 2.1. From this we describe two options to recover the primal value function. The first method involves solving the optimisation problem (2.13) via the introduction of a third loss function to optimise the initial dual variable $\tilde{x}$. This is referred to as the dual DC2BSDE method. The second option is to instead optimise the dual control process using optimality conditions arising from the SMP. This changes the machine learning algorithm slightly, as we no longer optimise using the Hamiltonian as a loss function. The key benefit of this method is the theory is applicable to non-Markovian problems, allowing us to solve control problems involving path-dependant coefficients.

2.3.1 The Dual DC2BSDE Method for the Markovian Case

When the dual problem in (2.13) is Markovian it has a similar structure to the primal problem (1.3), so we can adapt our machine learning method to solve it. We define a variable $\tilde{x}_0^0$ with which the state process begins, as opposed to starting at x_0. Suppose $\tilde{X}^{\tilde{x},\tilde{u}}$ is driven by a control $\tilde{u}$, a progressively measurable process, valued in $\tilde{U} \subset \mathbb{R}^m$, and has deterministic and measurable drift and diffusion functions $\tilde{b}\colon [0,T]\times\mathbb{R}^n\times\tilde{U} \to \mathbb{R}^n$ and $\tilde{\sigma}\colon [0,T]\times\mathbb{R}^n\times\tilde{U} \to \mathbb{R}^{n\times d}$ respectively. Then, in analogue to (2.3), we have solutions $\left(\tilde{V}^{\tilde{x},\tilde{u}}, \tilde{Y}^{\tilde{x},\tilde{u}}, \tilde{\Gamma}^{\tilde{x},\tilde{u}}\right) \in \mathcal{S}^2(\mathbb{R}) \times \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{H}^2(\mathbb{R}^{n\times n})$ to the 2BSDE

$$\begin{aligned}
d\tilde{X}_t^{\tilde{x},\tilde{u}} &= \tilde{b}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)dt + \tilde{\sigma}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)dW_t \\
d\tilde{V}_t^{\tilde{x},\tilde{u}} &= -\tilde{f}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)dt + \left(\tilde{Y}_t^{\tilde{x},\tilde{u}}\right)^{\top}\tilde{\sigma}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)dW_t \\
d\tilde{Y}_t^{\tilde{x},\tilde{u}} &= -\partial_x\tilde{H}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t, \tilde{Y}_t^{\tilde{x},\tilde{u}}, \tilde{\Gamma}_t^{\tilde{x},\tilde{u}}\sigma_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)\right)dt + \tilde{\Gamma}_t^{\tilde{x},\tilde{u}}\tilde{\sigma}_t\left(\tilde{X}_t^{\tilde{x},\tilde{u}}, \tilde{u}_t\right)dW_t,
\end{aligned} \tag{2.14}$$

with the initial condition $\tilde{X}_0^{\tilde{x},\tilde{u}} = \tilde{x}$ and terminal conditions $\tilde{V}_T^{\tilde{x},\tilde{u}} = \tilde{g}\left(\tilde{X}_T^{\tilde{x},\tilde{u}}\right)$ and $\tilde{Y}_T^{\tilde{x},\tilde{u}} = \nabla\tilde{g}\left(\tilde{X}_T^{\tilde{x},\tilde{u}}\right)$, where $\tilde{f}$ and $\tilde{g}$ are the dual running and terminal reward functions respectively. The function $\tilde{H}$ is defined as in (1.11) but with the corresponding dual functions. We discretise this 2BSDE and optimise using the dual Hamiltonian, denoted here as $\tilde{F}$, in the same way. Introduce parameters $\theta_i^0 \in \mathbb{R}^{\rho(n,m)}$, $\lambda_i^0 \in \mathbb{R}^{\rho\left(n,n^2\right)}$, $i = 1,\ldots,N$ and set $\tilde{X}_0^0 = \tilde{x}_0$, $\tilde{V}_0^0 = v_0^0$ and $\tilde{Y}_0^0 = y_0^0$ then for $i = 0,\ldots N-1$ let $u_i^0 = N_{\theta_i^0}\left(\tilde{X}_i^0\right)$, $\tilde{\Gamma}_i^0 = N_{\lambda_i^0}\left(\tilde{X}_i^0\right)$ and

$$\begin{aligned}
\tilde{X}_{i+1}^0 &= \tilde{X}_i^0 + (t_{i+1}-t_i)\,b_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0\right) + \sigma_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0\right)\Delta W_i \\
\tilde{V}_{i+1}^0 &= \tilde{V}_i^0 - (t_{i+1}-t_i)\,f_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0\right) + \left(\tilde{Y}_i^0\right)^{\top}\sigma_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0\right)\Delta W_i \\
\tilde{Y}_{i+1}^0 &= \tilde{Y}_i^0 - (t_{i+1}-t_i)\,\partial_x H_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0, \tilde{Y}_i^0, \tilde{\Gamma}_i^0\sigma_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i\right)\right) + \tilde{\Gamma}_i^0\sigma_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i\right)\Delta W_i,
\end{aligned} \tag{2.15}$$

where ΔW_i is a multivariate normal $\mathcal{N}_d\left(0,(t_{i+1}-t_i)\,\mathbb{1}_d\right)$ random variable. We have analogous loss functions $\tilde{\mathcal{L}}_1$ and $\tilde{\mathcal{L}}_2$ for this scheme, defined by

$$\tilde{\mathcal{L}}_1\left(\Lambda^0\right) := \mathbb{E}\left[\left|\tilde{V}_N^0 - \tilde{g}\left(\tilde{X}_N^0\right)\right|^2 + \gamma_1\left|\tilde{Y}_N^0 - \nabla\tilde{g}\left(\tilde{X}_N^0\right)\right|^2\right], \tag{2.16}$$

where $\Lambda^0 \in \mathbb{R}^{1+n+\rho\left(n,n^2\right)}$ is a vector representation of $\{\tilde{v}_0^0, \tilde{y}_0^0\} \cup (\lambda_i^0)_{i=0}^{N-1}$.

$$\tilde{\mathcal{L}}_2\left(\theta_i^0, i\right) := \mathbb{E}\left[-\tilde{F}_{t_i}\left(\tilde{X}_i^0, \tilde{u}_i^0, \tilde{Y}_i^0, \tilde{\Gamma}_i^0\right) + \gamma_2 \mathrm{dist}\left(\tilde{u}_i^0, \tilde{U}\right)\right]. \tag{2.17}$$

Optimising these solves the dual value problem and finds $\tilde{u}$. We now introduce a third loss function to recover the primal value function from this system

$$\tilde{\mathcal{L}}_3\left(\tilde{x}_0^0\right) = -\left(\mathbb{E}\left[\tilde{g}\left(\tilde{X}_N^0\right)\right] - (\tilde{x}_0^0)^\top x_0\right). \tag{2.18}$$

Since the primal problem is described as a minimisation problem, the dual problem, and the duality optimisation problem become maximisations, as we see in our examples in the sequel. Hence the respective loss functions are multiplied by -1. Algorithm 4 gives the details of this method.

Example 2.3.1. For the utility maximisation problem (2.12), we can apply the dual method with

$$\begin{aligned}
\tilde{b}_t\left(\tilde{x}, \tilde{\pi}\right) &= -\tilde{x}\left(r_t + \delta_K\left(\tilde{\pi}\right)\right), \\
\tilde{\sigma}_t\left(\tilde{x}, \tilde{\pi}\right) &= -\tilde{x}\left(\theta_t + \Sigma_t^{-1}\tilde{\pi}\right)^\top, \\
\tilde{f}_t\left(\tilde{x}, \tilde{\pi}\right) &= 0, \\
\tilde{g}\left(\tilde{x}\right) &= \tilde{U}\left(\tilde{x}\right)\mathbb{1}_{x>0},
\end{aligned}$$

for $(t, \tilde{x}, \tilde{\pi}) \in [0,T] \times \mathbb{R} \times \mathbb{R}^m$. In some cases, $\delta_K(z)$ can be infinite for some $z \notin \tilde{K} \subset \mathbb{R}^m$. In these cases we remove this term and consider the constrained optimisation over $\tilde{K}$.

Algorithm 4: Dual Deep Controlled 2BSDE

Input: $x_0 \in \mathbb{R}^n$

Result: $v(0, x_0)$

Initialise $\tilde{V}_0^0 \sim \text{Unif}([-0.1, 0.1])$, $\tilde{x}_0^0 \sim x_0 + \text{Unif}([-0.1, 0.1]^n)$ and $\tilde{Y}_0^0 \sim \text{Unif}\left([-0.1, 0.1]^d\right)$;

for $i = 0, 1, 2, \ldots, N-1$ **do**

  Initialise $\theta_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho(n,m)}\right)$, $\lambda_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho\left(n,n^2\right)}\right)$;

end

Set $\Lambda^0 = \{\tilde{V}_0^0, \tilde{Y}_0^0\} \cup \{\lambda_i^0\}_{i=0}^{N-1}$;

Set $\Theta^0 = \{\theta_i^0\}_{i=0}^{N-1}$;

Initialise $m^0 = v^0 = 0$, $m_x^0 = 0$, $v_x^0 = 0$, $j = 0$ and $m_i^0 = v_i^0 = 0$ for $i = 0, \ldots, N-1$;

while $\Lambda^j, \Theta^j, \tilde{x}_0^j$ *not converged* **do**

  Generate ΔW_i^k for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  Generate $\tilde{X}^{j,k}, \tilde{V}^{j,k}, \tilde{Y}^{j,k}, \tilde{u}^{j,k}, \tilde{\Gamma}^{j,k}$ using (2.15) and $\left(\Lambda^j, \Theta^j, \tilde{x}_0^j\right)$ for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  Set $\tilde{\mathcal{L}}_1^j(\Lambda^j) = \frac{1}{B}\sum_{k=1}^B \left|\tilde{V}_N^{j,k} - g\left(\tilde{X}_N^{j,k}\right)\right|^2 + \gamma_1 \left|\tilde{Y}_N^{j,k} - \nabla g\left(\tilde{X}_N^{j,k}\right)\right|^2$;

  Set $\Lambda^{j+1}, m^{j+1}, v^{j+1} = \text{ADAM}(\Lambda^j, \tilde{\mathcal{L}}_1^j, m^j, v^j, j)$;

  Regenerate $\tilde{X}^{j,k}, \tilde{V}^{j,k}, \tilde{Y}^{j,k}, \tilde{u}^{j,k}, \tilde{\Gamma}^{j,k}$ using (2.15) and $\left(\Lambda^{j+1}, \Theta^j, \tilde{x}_0^j\right)$ for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  for $i = 0, 1, 2, \ldots, N-1$ **do**

    Set $\tilde{\mathcal{L}}_2^j\left(\theta_i^j, i\right) = -\frac{1}{B}\sum_{k=1}^B F_{t_i}\left(\tilde{X}_i^{j,k}, \tilde{u}_i^{j,k}, \tilde{Y}_i^{j,k}, \tilde{\Gamma}_i^{j,k}\right) + \gamma_2 \text{dist}(\tilde{u}^{j,k}, \tilde{U})$;

    Set $\theta_i^{j+1}, m_i^{j+1}, v_i^{j+1} = \text{ADAM}(\theta_i^j, \tilde{\mathcal{L}}_2^j(\cdot, i), m_i^j, v_i^j, j)$;

  end

  Set $\Theta^{j+1} = \{\theta_i^{j+1}\}_{i=0}^{N-1}$;

  Set $\tilde{\mathcal{L}}_3^j\left(\tilde{x}_0^j\right) = x_0^\top \tilde{x}_0^j - \frac{1}{B}\sum_{k=1}^B \tilde{g}\left(\tilde{X}_N^j\right)$;

  Set $\tilde{x}_0^{j+1}, m_x^{j+1}, v_x^{j+1} = \text{ADAM}(\tilde{x}_0^j, \tilde{\mathcal{L}}_3^j(\cdot), m_x^j, v_x^j, j)$;

  Set $j = j + 1$;

end

Output $\tilde{V}_0^j - x_0^\top \tilde{x}_0^j$;

2.3.2 The Deep SMP Method

In this section we consider the Utility Maximisation problem (2.9) in the case that r, μ and Σ may not be deterministic. The utility problem and its dual problem then become non-Markovian. This means we cannot write the value function as a measurable function of state and time nor apply the dynamical programming principle, hence the HJB equation is invalid. However, we can still employ the SMP, and it is shown in Li and Zheng (2018) that there exist certain optimality conditions for a non-Markovian utility maximisation problem, which we use to define a new algorithm, named the deep SMP method. We again consider the utility problem, written in this setting as

$$v(t) = \sup_{\pi \in \mathcal{K}} \mathbb{E}\big[U\left(X_T^\pi\right) \,\big|\, \mathcal{F}_t\big], \qquad 0 \leq t \leq T.$$

We cannot use Itô's formula for this value process as it cannot be written as a measurable function of the wealth process X^π, defined as in (2.8) and controlled by π, but we can still use the SMP, for both the primal and dual problems. In this setting, for the primal problem the BSDE (1.10) becomes $(Y^\pi, Z^\pi) \in \mathcal{S}^2(\mathbb{R}) \times \mathcal{H}^2(\mathbb{R}^m)$ such that

$$\begin{aligned} dY_t^\pi &= -\Big[\left(r_t + \pi_t^\top \Sigma_t \theta_t\right) Y_t^\pi + \left(Z_t^\pi\right)^\top \Sigma_t^\top \pi_t\Big]\, dt + \left(Z_t^\pi\right)^\top dW_t, \qquad 0 \leq t \leq T, \\ Y_T^\pi &= U'\left(X_T^\pi\right) \end{aligned} \tag{2.19}$$

Similarly, we can define the dual state $\tilde{X}^{\tilde{x},\tilde{\pi}}$ as per (2.11), controlled by $(\tilde{x}, \tilde{\pi})$. The dual adjoint equation is the following BSDE with unknown processes $\left(\tilde{Y}^{\tilde{x},\tilde{\pi}}, \tilde{Z}^{\tilde{x},\tilde{\pi}}\right) \in \mathcal{S}^2(\mathbb{R}) \times \mathcal{H}^2(\mathbb{R}^m)$

$$\begin{aligned} d\tilde{Y}_t^{\tilde{x},\tilde{\pi}} &= \left[\left(r_t + \delta_K\left(\tilde{\pi}_t\right)\right) \tilde{Y}_t^{\tilde{x},\tilde{\pi}} + \left(\tilde{Z}_t^{\tilde{x},\tilde{\pi}}\right)^\top \left(\theta_t + \Sigma_t^{-1} \tilde{\pi}_t\right)\right] dt + \left(\tilde{Z}_t^{\tilde{x},\tilde{\pi}}\right)^\top dW_t, \qquad 0 \leq t \leq T, \\ \tilde{Y}_T^{\tilde{x},\tilde{\pi}} &= \tilde{U}'\left(\tilde{X}_T^{\tilde{x},\tilde{\pi}}\right). \end{aligned} \tag{2.20}$$

We use the following primal-dual relations, derived in Li and Zheng (2018).

Theorem 2.3.2 (Li and Zheng (2018), Theorem 12)**.** *Suppose that $\tilde{x}^*$ and $\tilde{\pi}^*$ are optimal dual controls, with the corresponding dual state process $\tilde{X}^{\tilde{x}^*,\tilde{\pi}^*}$ and dual adjoint BSDE solutions $\tilde{Y}^{\tilde{x}^*,\tilde{\pi}^*}, \tilde{Z}^{\tilde{x}^*,\tilde{\pi}^*}$. Then the primal state process X^{π^*} and primal adjoint BSDE solutions Y^{π^*}, Z^{π^*}*

corresponding to the optimal primal control π^* satisfy

$$X_t^{\pi^*} = -\tilde{Y}_t^{\tilde{x}^*,\tilde{\pi}^*}, \qquad Y_t^{\pi^*} = \tilde{X}_t^{\tilde{x}^*,\tilde{\pi}^*}, \tag{2.21}$$

for all $t \in [0, T]$.

We now state the optimality conditions for the dual problem in terms of these processes.

Theorem 2.3.3 (Li and Zheng (2018), Theorem 11)**.** *Let $(\tilde{x}, \tilde{\pi})$ be an admissible dual control. Then $(\tilde{x}, \tilde{\pi})$ is optimal if and only if the solutions $\left(\tilde{Y}^{\tilde{x},\tilde{\pi}}, \tilde{Z}^{\tilde{x},\tilde{\pi}}\right)$ of (2.20) satisfy the following*

1. $\tilde{Y}_0^{\tilde{x},\tilde{\pi}} = -x_0$,
2. $\frac{\left(\Sigma_t^\top\right)^{-1} \tilde{Z}_t^{\tilde{x},\tilde{\pi}}}{\tilde{Y}_t^{\tilde{x},\tilde{\pi}}} \in K$,
3. $\tilde{Y}_t^{\tilde{x},\tilde{\pi}} \delta_K(\tilde{\pi}_t) + \left(\tilde{Z}_t^{\tilde{x},\tilde{\pi}}\right)^\top \Sigma_t^{-1} \tilde{\pi}_t = 0$,

for all $t \in [0, T]$ almost surely.

Remark 6. The corresponding optimality condition for a primal control π with corresponding state process X^π and adjoint BSDE solutions (Y^π, Z^π) is

$$[\pi_t - a]^\top [X_t^\pi \Sigma_t (Y_t^\pi \theta_t + Z_t^\pi)] \geq 0$$

for all $t \in [0, T]$ and $a \in K$ almost surely. This is a much more complex condition, and the implementation would require some grid-based method to consider a sufficiently large number of $a \in K$, which would be computationally costly, especially in higher dimensions and unbounded (but still constrained) K. For this reason, we approach the dual conditions and then use the primal dual relations to derive the necessary primal processes.

The machine learning methodology to solve this problem, which we call in the sequel the deep controlled SMP (DCSMP) method, is as follows. First, we simulate the dual processes in the forward direction using the Euler-Maruyama scheme. We start the process $\tilde{Y}^{\tilde{x},\tilde{\pi}}$ at $\tilde{Y}_0^{\tilde{x},\tilde{\pi}} = -x_0$ to ensure the first condition is satisfied. This condition is why it appears that the forward simulation method is naturally suited to this problem. We use neural networks to define the processes $\tilde{\pi}$ and $\tilde{Z}^{\tilde{x},\tilde{\pi}}$, then update the values of $\tilde{X}^{\tilde{x},\tilde{\pi}}$ and $\tilde{Y}^{\tilde{x},\tilde{\pi}}$ accordingly.

Specifically, we set $\tilde{X}_0^0 = \tilde{x}^0$, $\tilde{Y}_0^0 = -x_0$ then for $i = 0, \dots, N-1$ set $\tilde{\pi}_i^0 = N_{\theta_i}\left(\tilde{X}_i^0\right)$ and

$$\begin{aligned}
\tilde{X}_{i+1}^0 &= \tilde{X}_i^0 - (t_{i+1} - t_i)\, \tilde{X}_i^0 \big(r_{t_i} + \delta_K\left(\tilde{\pi}_i^0\right)\big) - \tilde{X}_i^0 \left(\theta_{t_i} + \Sigma_{t_i}^{-1} \tilde{\pi}_i^0\right)^\top \Delta W_i, \\
\tilde{Y}_{i+1}^0 &= \tilde{Y}_i^0 + (t_{i+1} - t_i) \left[r_{t_i} \tilde{Y}_i^0 + \left(\tilde{Z}_i^0\right)^\top \theta_{t_i} \right] + \left(\tilde{Z}_i^0\right)^\top \Delta W_i, \\
\tilde{Z}_{i+1}^0 &= \tilde{Y}_i^0 \Sigma_{t_i}^\top N_{\lambda_i}\left(\tilde{X}_i^0\right),
\end{aligned} \tag{2.22}$$

where $\theta_i, \lambda_i \in \mathbb{R}^{\rho(1,m)}$ are parameters for neural networks, ΔW_i is a multivariate normal $\mathcal{N}_m\left(0, (t_{i+1} - t_i)\, \mathbb{1}_m\right)$ random variable. The two neural networks used here both output to m-dimensional space, unlike the general algorithm outlined in Section 2.1, where the second neural network is valued in $\mathbb{R}^{n \times n}$. We have the variables $y, \Theta := (\theta_i)_{i=0}^{N-1}$ and $\Lambda := (\lambda_i)_{i=0}^{N-1}$, which are optimised against the following loss functions:

$$\begin{aligned}
\mathcal{L}^{\tilde{x}}\left(\tilde{x}_0^0\right) &= \mathbb{E}\left[\tilde{U}\left(\tilde{X}_N^0\right)\right] + x_0 \tilde{x}_0^0, \\
\mathcal{L}^{\tilde{Z}}\left(\lambda_i^0, i\right) &= \mathbb{E}\left[\left|\tilde{U}'\left(\tilde{X}_N^0\right) - \tilde{Y}_N^0\right|^2 + \gamma \mathrm{dist}\left(N_{\lambda_i}\left(\tilde{X}_i^0\right), K\right)\right], \\
\mathcal{L}^{\tilde{\pi}}\left(\theta_i^0, i\right) &= \mathbb{E}\left[\left|\tilde{Y}_i^0 \delta_K\left(\tilde{\pi}_i^0\right) + \left(\tilde{Z}_i^0\right)^\top \Sigma_{t_i}^{-1} \tilde{\pi}_i^0\right|^2\right].
\end{aligned}$$

We take sample means to approximate the expected values. The hyperparameter $\gamma > 0$ is some large penalty constant, in our case $\gamma = 100$. These loss functions ensure the optimality conditions of Theorem 2.3.3 are satisfied.

In this algorithm we no longer have a variable representing the value at time 0, as we do not assume the Markovian structure that gives us $\tilde{V}$. We therefore need to perform Monte Carlo simulations, forming bounds of the value function. An upper bound can be found using the dual state process $\tilde{X}$ and a lower bound the primal state process $X^\pi = -\tilde{Y}^{\tilde{x}, \tilde{\pi}}$, as per (2.21). The estimates outputted are given by

$$v_{\text{low}} := \mathbb{E}\left[U\left(-\tilde{Y}_N^j\right)\right] \leq u\left(0, x_0\right) \leq \mathbb{E}\left[\tilde{U}\left(\tilde{X}_N^j\right)\right] + x_0 \tilde{x}_0^j =: v_{\text{high}} \tag{2.23}$$

for some sufficiently large j. The inequalities are valid up to discretisation and simulation errors, and as the number of time steps increases, the gap $v_{\text{high}} - v_{\text{low}}$ decreases. Assuming convergence of the control process and corresponding state and adjoint processes, the optimal value is eventually contained in this gap. Algorithm 5 gives the details of the DCSMP method.

Algorithm 5: Deep Controlled SMP for the Utility Maximisation problem

Input: $x_0 \in \mathbb{R}$

Result: A confidence interval for $v(0, x_0)$

Initialise $\tilde{x}_0^0 \sim \text{Unif}([x-0.1, x+0.1])$;

for $i = 0, 1, 2, \ldots, N-1$ **do**

  Initialise $\theta_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho(n,m)}\right)$, $\lambda_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{\rho\left(n,n^2\right)}\right)$;

end

Set $\Lambda^0 = \{\lambda_i^0\}_{i=0}^{N-1}$;

Set $\Theta^0 = \{\theta_i^0\}_{i=0}^{N-1}$;

Initialise $m_x^0 = 0$, $v_x^0 = 0$, $j = 0$ and $m_{\pi,i}^0 = v_{\pi,i}^0 = 0$, $m_{Z,i}^0 = v_{Z,i}^0 = 0$ for $i = 0, \ldots, N-1$;

while $\Lambda^j, \Theta^j, \tilde{x}_0^j$ *not converged* **do**

  Generate ΔW_i^k for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  Generate $\tilde{X}^{j,k}, \tilde{Y}^{j,k}, \tilde{Z}^{j,k}, \tilde{\pi}^{j,k}$ using (2.22) and $\left(\Lambda^j, \Theta^j, \tilde{x}_0^j\right)$ for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  for $i = 0, 1, 2, \ldots, N-1$ **do**

    Set $\mathcal{L}^{\tilde{Z}}\left(\lambda_i^j, i\right) = \frac{1}{B}\sum_{k=1}^{B}\left|\tilde{U}'\left(\tilde{X}_N^{j,k}\right) - \tilde{Y}_N^{j,k}\right|^2 + \gamma \text{dist}\left(N_{\lambda_i}\left(\tilde{X}_i^{j,k}\right), K\right)$;

    Set $\lambda_i^{j+1}, m_{Z,i}^{j+1}, v_{Z,i}^{j+1} = \text{ADAM}(\Lambda^j, \mathcal{L}^{\tilde{Z}}(\cdot, i), m_{Z,i}^j, v_{Z,i}^j, j)$;

  end

  Set $\Lambda^{j+1} = \{\lambda_i^{j+1}\}_{i=0}^{N-1}$;

  Regenerate $\tilde{X}^{j,k}, \tilde{Y}^{j,k}, \tilde{Z}^{j,k}, \tilde{\pi}^{j,k}$ using (2.22) and $\left(\Lambda^{j+1}, \Theta^j, \tilde{x}_0^j\right)$ for each $k = 1, \ldots B$, $i = 0, \ldots, N-1$;

  for $i = 0, 1, 2, \ldots, N-1$ **do**

    Set $\mathcal{L}^{\tilde{\pi}}\left(\theta_i^j, i\right) = -\frac{1}{B}\sum_{k=1}^{B}\left|\tilde{Y}_i^{j,k}\delta_K\left(\tilde{\pi}_i^{j,k}\right) + \left(\tilde{Z}_i^{j,k}\right)^\top \Sigma_{t_i}^{-1}\tilde{\pi}_i^{j,k}\right|^2$;

    Set $\theta_i^{j+1}, m_{\pi,i}^{j+1}, v_{\pi,i}^{j+1} = \text{ADAM}(\theta_i^j, \mathcal{L}^{\tilde{\pi}}(\cdot, i), m_{\pi,i}^j, v_{\pi,i}^j, j)$;

  end

  Set $\Theta^{j+1} = \{\theta_i^{j+1}\}_{i=0}^{N-1}$;

  Set $\mathcal{L}^{\tilde{x}}\left(\tilde{x}_0^j\right) = \frac{1}{B}\sum_{k=1}^{B}\tilde{U}\left(\tilde{X}_N^j\right) + x^\top \tilde{x}_0^j$;

  Set $\tilde{x}_0^{j+1}, m_x^{j+1}, v_x^{j+1} = \text{ADAM}(\tilde{x}_i^j, \mathcal{L}^{\tilde{x}}, m_x^j, v_x^j, j)$;

  Set $j = j+1$;

end

Output $v_{\text{low}}, v_{\text{high}}$ defined in (2.23);

2.4 Random Income

In this section we study the investment-consumption problem with randomly terminating income. This problem has an irregular structure due to an income stream that will terminate at some exponential time. To handle this, we reformulate the problem by converting the random income stream into a running cost for a transformed control problem. We split the time horizon into two regions: pre terminating, in which the income is perpetual over the entire time horizon; and post-terminating, in which there is no income term. We can solve both of these problems using our machine learning method, and recombine them to yield the solutions to the original problem. The problem with infinite time horizon and its duality theory is covered in Davey et al. (2021).

2.4.1 Primal Problem

Let S be the stock market (2.7). Consider an agent with wealth process $(X_t^{\pi,c})_{0\leq t\leq T}$ investing proportions $\pi_t \in K$ for some convex set $K \subset \mathbb{R}^m$ of their wealth in this stock market at any time $t \in [0, T]$, and the remainder in a risk free bond with uniformly bounded, deterministic interest rate $r_t > 0$. Furthermore, the agent may choose to consume some of their wealth before the time horizon. In addition to the wealth accrued from investing, the agent receives a continuous payment with rate $a \geq 0$, which terminates at some random exponential time $\tau \geq 0$ independent of S. The agent wishes to solve

$$\sup_{\pi,c} \mathbb{E}\left[\int_0^T e^{-\delta s} U\left(c_s\right) ds + e^{-\delta T} U_1\left(X_T^{\pi,c}\right)\right],$$

where $X^{\pi,c}$ is the wealth process satisfying a controlled stochastic differential equation:

$$dX_t^{\pi,c} = \left[X_t^{\pi,c}\left(r_t + \pi_t^\top\left(\mu_t - r_t\right)\right) - c_t + a_t\right] dt + X_t^{\pi,c}\pi_t^\top \Sigma_t dW_t, \quad 0 \leq t \leq T, \tag{2.24}$$

with the initial wealth $X_0^{\pi,c} = x_0$ where U and U_1 are utility functions satisfying the Inada conditions, $\delta > 0$ is a utility discount factor, and $a_t = a\mathbb{1}_{\tau>t}$ is the random income with exponential termination time $\tau \sim \mathrm{EXP}\left(\eta\right)$ for some $\eta \geq 0$. The proportional investment process π is constrained to the set K, and consumption process c is constrained to the non-negative real line $[0, \infty)$.

We assume that the problem is solved when there is no income, that is, we know how to compute the value function

$$v_0\left(t,x\right) := \sup_{\pi,c}\mathbb{E}\left[\int_t^T e^{-\delta(s-t)}U\left(c_s\right)ds + e^{-\delta(T-t)}U_1\left(X_T^{\pi,c,0}\right)\middle| X_t^{\pi,c,0}=x\right],$$

for $(t,x)\in[0,T]\times\mathbb{R}$, where

$$dX_s^{\pi,c,0} = \left[X_s^{\pi,c,0}\left(r_s+\pi_s^\top\Sigma_s\left(\mu_s-r_s\right)\right)-c_s\right]ds + X_s^{\pi,c,0}\pi_s^\top\Sigma_s dW_s, \quad 0\le s\le T,$$

with the initial condition $X_t^{\pi,c,0}=x$. This can be found either analytically, or numerically using our method. Note that $v_0\left(T,x\right)=U_1\left(x\right)$ for all $x\in\mathbb{R}$.

Define the value function

$$v\left(t,x\right) = \sup_{\pi,c}\mathbb{E}\left[\int_t^T e^{-\delta(s-t)}U\left(c_s\right)ds + e^{-\delta(T-t)}U_1\left(X_T^{\pi,c}\right)\right], \tag{2.25}$$

for $(t,x)\in[0,T]\times\mathbb{R}$ where $X^{\pi,c}$ satisfies (2.24) with $X_t^{\pi,c}=x$. Fix $t\in[0,T]$ and assume $\tau>t$, i.e, the income has not terminated yet at time t. We can relate v and v_0 as follows. Define

$\lambda := T \wedge \tau$ and $\alpha = \eta + \delta$. Then

$$v(t,x) \tag{2.26}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^\lambda e^{-\delta(s-t)} U(c_s)\,ds + \int_\lambda^T e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(T-t)} U_1\left(X_T^{\pi,c}\right) \middle| X_t^{\pi,c} = x\right] \tag{2.27}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^\lambda e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(\lambda-t)} \left(\int_\lambda^T e^{-\delta(s-\lambda)} U(c_s)\,ds + e^{-\delta(T-\lambda)} U_1\left(X_T^{\pi,c}\right)\right) \middle| X_t^{\pi,c} = x\right] \tag{2.28}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^\lambda e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(\lambda-t)} v_0\left(\lambda, X_\lambda^{\pi,c}\right) \middle| X_t^{\pi,c} = x\right] \tag{2.29}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^\infty \eta e^{-\eta(z-t)} \left(\int_t^{z\wedge T} e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(z\wedge T-t)} v_0\left(z\wedge T, X_{z\wedge T}^{\pi,c}\right)\right) dz \middle| X_t^{\pi,c} = x\right] \tag{2.30}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^T \eta e^{-\eta(z-t)} \left(\int_t^z e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(z-t)} v_0\left(z, X_z^{\pi,c}\right)\right) dz\right. \tag{2.31}$$

$$\left. + \int_T^\infty \eta e^{-\eta(z-t)} \left(\int_t^T e^{-\delta(s-t)} U(c_s)\,ds + e^{-\delta(T-t)} v_0\left(T, X_T^{\pi,c}\right)\right) dz \middle| X_t^{\pi,c} = x\right] \tag{2.32}$$

$$= \sup_{\pi,c} \mathbb{E}\left[\int_t^T e^{-\alpha(s-t)} f_s\left(X_s^{\pi,c,a}, \pi_s, c_s\right) ds + e^{-\alpha(T-t)} g\left(X_T^{\pi,c,a}\right) \middle| X_t^{\pi,c,a} = x\right], \tag{2.33}$$

where, for $(t,x,\pi,c) \in [0,T] \times (0,\infty) \times \mathbb{R}^m \times [0,\infty)$

$$dX_t^{\pi,c,a} = \left(X_t^{\pi,c,a}\left(r_t + (\pi_t)^\top (\mu_t - r_t)\right) - c_t + a\right) dt + X_t^{\pi,c,a} (\pi_t)^\top \Sigma_t dW_t, \tag{2.34}$$

$$f_t(x,\pi,c) = U(c) + \eta v_0(t,x), \tag{2.35}$$

$$g(x) = U_1(x). \tag{2.36}$$

The structure of the problem has changed, as the wealth process $X^{\pi,c,a}$ now has perpetual income, and the income termination time has been converted into a running cost function. This frames the control problem in a more standard way, which we may approach using our machine learning method. As we have discounting, the 2BSDE changes slightly, and we alter (2.4) to the following, for each $i = 1, \ldots, N-1$,

$$V_{i+1}^0 = V_i^0 + (t_{i+1} - t_i)\left(\alpha V_i^0 - f_{t_i}\left(X_i^0, u_i^0\right)\right) + \left(Y_i^0\right)^\top \sigma_{t_i}\left(X_i^0, u_i^0\right) \Delta W_i,$$
$$Y_{i+1}^0 = Y_i^0 + (t_{i+1} - t_i)\left(\alpha Y_i^0 - \partial_x H_{t_i}\left(X_i^0, u_i^0, Y_i^0, \Gamma_i^0 \sigma_{t_i}\left(X_i^0, u_i\right)\right)\right) + \Gamma_i^0 \sigma_{t_i}\left(X_i^0, u_i\right) \Delta W_i.$$

From the HJB equation (1.7) we can derive the optimal controls in terms of the value function and their derivatives, that is,

$$\begin{aligned} \pi_t^* &= \arg\max_{\pi\in K}\left\{x\partial_x v\left(t,x\right)\pi^\top\left(\mu-r\right)+\frac{1}{2}\partial_{xx}^2 v\left(t,x\right)x^2|\Sigma^\top\pi|^2\right\} \\ c_t^* &= \arg\max_{c\geq 0}\left\{U\left(c\right)-\partial_x v\left(t,x\right)c\right\}=I\left(\partial_x v\left(t,x\right)\right), \end{aligned} \tag{2.37}$$

where $I := \left(U'\right)^{-1}$.

2.4.2 Dual Problem

We can apply results of convex duality Bismut (1973) to the transformed problem (2.33). First, note that the constrained problem is equivalent to the unconstrained maximisation problem with

$$f_t\left(x,\pi,c\right)=U\left(c\right)+\eta v_0\left(t,x\right)-\Phi_{[0,\infty)}\left(c\right)-\Phi_K\left(\pi\right),$$

for $(t,x,\pi,c)\in[0,T]\times(0,\infty)\times\mathbb{R}\times[0,\infty)$ where $\Phi_A\left(z\right)$ is the infinite penalty function that equals 0 if $z\in A$ and ∞ otherwise. We can define the following dual functions, for $0\leq t\leq T$ and $\tilde{x}>0$,

$$\begin{aligned} \tilde{U}\left(\tilde{x}\right) &= \sup_{x>0}\{U\left(x\right)-xy\}, \\ \tilde{U}_1\left(\tilde{x}\right) &= \sup_{x>0}\{U_1\left(x\right)-xy\}, \\ \tilde{v}_0\left(t,\tilde{x}\right) &= \sup_{x>0}\{v_0\left(t,x\right)-xy\}. \end{aligned}$$

The dual running cost is not immediately obvious a priori. In particular, we will see that it is not just a sum of dual functions. Define the strictly positive dual process $\tilde{X}$ by

$$d\tilde{X}_t=\tilde{X}_t\left[\tilde{\alpha}_t dt+\tilde{\beta}_t^\top dW_t\right],\quad 0\leq t\leq T,$$

with the initial condition $\tilde{X}_0=\tilde{x}_0$ for some processes $\tilde{\alpha}\in\mathcal{H}^2\left(\mathbb{R}\right),\tilde{\beta}\in\mathcal{H}^2\left(\mathbb{R}^m\right)$ to be determined. By Itô's Lemma, the drift of the process $\left(e^{-\alpha t}X_t^{\pi,c,a}\tilde{X}_t\right)_{t\in[0,T]}$ is

$$e^{-\alpha t}X_t^{\pi,c,a}\tilde{X}_t\left[\tilde{\alpha}_t-\alpha+r_t+\pi_t^\top\left(\mu_t-r_t+\Sigma_t^\top\tilde{\beta}_t\right)\right]+e^{-\alpha t}\tilde{X}_t\left(a-c_t\right),\qquad 0\leq t\leq T.$$

Define a process L for $t \in [0,T]$ by

$$L_t = e^{-\alpha t} X_t^{\pi,c,a} \tilde{X}_t - \int_0^t e^{-\alpha s} \tilde{X}_s \left(a - c_s + X_s^{\pi,c,a} \left[\tilde{\alpha}_s - \alpha + r_s + \pi_s^\top \left(\mu_s - r_s + \Sigma_s^\top \tilde{\beta}_s \right) \right] \right) ds. \tag{2.38}$$

By construction L has zero drift so is a local martingale. Define $\tilde{\pi}_t = -\Sigma_t \left(\theta_t + \tilde{\beta}_t \right)$ and $\xi_t = \frac{\tilde{X}_t}{\eta} \left(\tilde{\alpha}_t + r_t - \alpha \right)$, where $\theta_t = \Sigma_t^{-1} \left(\mu_t - r_t \right)$, $t \in [0,T]$. Then the process $\tilde{X}$ is given by

$$d\tilde{X}_t = \left[\tilde{X}_t \left(\alpha - r_t \right) + \eta \xi_t \right] dt - \tilde{X}_t \left(\theta_t + \Sigma_t^{-1} \tilde{\pi}_t \right)^\top dW_t, \qquad 0 \le t \le T.$$

The diffusion of L is $X_t^{\pi,c,a} \tilde{X}_t \left(\Sigma_t^\top \pi_t - \theta_t + \left(\Sigma_t^{-1} \right)^\top \tilde{\pi}_t \right)$ for $t \in [0,T]$. One can show that L is a true martingale by using Doob's L^2 inequality to show that this diffusion process is square integrable. To do this we need to assume that the control processes are themselves square integrable Labbé and Heunis (2009). Therefore $\mathbb{E}[L_T] = L_0 = x_0 \tilde{x}_0$, and by the definition of the dual of U_1 we have, for any $(\pi, c, \tilde{\pi}, \xi) \in \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}^m \times \mathbb{R}$

$$\begin{aligned}
&\mathbb{E} \left[\int_0^T e^{-\alpha s} f\left(s, X_s^{\pi,c,a}, \pi_s, c_s\right) ds + e^{-\alpha T} U_1 \left(X_T\right) \right] && (2.39) \\
&\le \mathbb{E} \left[\int_0^T e^{-\alpha s} f\left(s, X_s^{\pi,c,a}, \pi_s, c_s\right) ds + e^{-\alpha T} \left(\tilde{U}_1 \left(\tilde{X}_T \right) + X_T^{\pi,c,a} \tilde{X}_T \right) \right] && (2.40) \\
&= \mathbb{E} \Bigg[\int_0^T e^{-\alpha s} \Big(f\left(s, X_s^{\pi,c,a}, \pi_s, c_s\right) - X_s^{\pi,c,a} \tilde{X}_s \pi_s^\top \tilde{\pi}_s && (2.41) \\
&\qquad + \eta X_s^{\pi,c,a} \xi_s - c_s \tilde{X}_s + a \tilde{X}_s \Big) ds + e^{-\alpha T} \tilde{U}_1 \left(\tilde{X}_T \right) \Bigg] + x_0 \tilde{x}_0. && (2.42)
\end{aligned}$$

Maximising over c, π yields

$$v\left(0, x_0\right) \le \mathbb{E} \left[\int_0^T e^{-\alpha t} \tilde{f}_t \left(\tilde{X}_t, \tilde{\pi}_t, \xi_t \right) ds + e^{-\alpha T} \tilde{U}_1 \left(\tilde{X}_T \right) \right] + x_0 \tilde{x}_0,$$

where

$$\begin{aligned}
\tilde{f}_t \left(\tilde{x}, \tilde{\pi}, \xi \right) &= \sup_{x,c,\pi} \{ U(c) - \Phi_{[0,\infty)}(c) - \Phi_K(\pi) - c\tilde{x} + \eta u_0(t,x) + \eta x \xi - x \tilde{x} \pi^\top \tilde{\pi} + a \tilde{x} \} \\
&= \tilde{U}(\tilde{x}) + a\tilde{x} + \Phi_{[0,\infty)}(\tilde{x}) + \sup_{x,\pi} \{ \eta u_0(t,x) + \eta x \xi - x \tilde{x} \pi^\top \tilde{\pi} - \Phi_K(\pi) \},
\end{aligned}$$

for $(t,\tilde{x},\tilde{\pi},\xi)\in[0,T]\times(0,\infty)\times\mathbb{R}^m\times\mathbb{R}$. The dual running cost then becomes

$$\begin{aligned}\tilde{f}_t(\tilde{x},\tilde{\pi},\xi)&=\tilde{U}(\tilde{x})+a\tilde{x}+\Phi_{[0,\infty)}(\tilde{x})+\sup_x\{\eta u_0(t,x)+\eta x\xi+x\tilde{x}\delta_K(\tilde{\pi})\}\\&=\tilde{U}(\tilde{x})+a\tilde{x}+\Phi_{[0,\infty)}(\tilde{x})+\eta v_0\left(t,-\xi-\frac{1}{\eta}\tilde{x}\delta_K(\tilde{\pi})\right).\end{aligned}$$

Define a new control $\gamma_t=-\xi_t-\frac{1}{\eta}\tilde{X}_t\delta_K(\tilde{\pi}_t)$, $t\in[0,T]$. In order for the running cost to be finite we must have $\gamma_t>0$. Minimising over $(\tilde{x}_0,\tilde{\pi},\gamma)\in(0,\infty)\times\mathcal{H}^2(\mathbb{R}^m)\times\mathcal{H}^2((0,\infty))$ gives us the following dual problem

$$\inf_{\tilde{x},\tilde{\pi},\gamma}\tilde{J}(\tilde{x},\tilde{\pi},\gamma):=\mathbb{E}\left[\int_0^T e^{-\alpha s}\tilde{f}\left(s,\tilde{X}_s^{\tilde{x},\tilde{\pi},\gamma},\tilde{\pi}_s,\gamma_s\right)ds+e^{-\alpha T}\tilde{g}\left(\tilde{X}_T^{\tilde{x},\tilde{\pi},\gamma}\right)\right]+x_0\tilde{x}, \tag{2.43}$$

where $\tilde{X}^{\tilde{x},\tilde{\pi},\gamma}$ is the dual process satisfying the SDE

$$d\tilde{X}_t^{\tilde{x},\tilde{\pi},\gamma}=\left(\tilde{X}_t^{\tilde{x},\tilde{\pi},\gamma}(\alpha-r_t-\delta_K(\tilde{\pi}_t))-\eta\gamma_t\right)dt-\tilde{X}_t^{\tilde{x},\tilde{\pi},\gamma}\left(\theta_t+\Sigma_t^{-1}\tilde{\pi}_t\right)^\top dW_t,\qquad 0\le t\le T, \tag{2.44}$$

with the initial condition $\tilde{X}_0^{\tilde{x},\tilde{\pi},\gamma}=\tilde{x}$ and

$$\begin{aligned}\tilde{f}_t(\tilde{x},\gamma)&=\tilde{U}(\tilde{x})+a\tilde{x}+\eta v_0(t,\gamma)\\\tilde{g}(\tilde{x})&=\tilde{U}_1(\tilde{x}),\end{aligned}$$

for $(\tilde{x},\gamma)\in(0,\infty)^2$. The dual value function is therefore given by

$$\tilde{v}(t,\tilde{x})=\inf_{\gamma>0,\tilde{\pi}\in\mathbb{R}^k}\mathbb{E}\left[\int_t^T e^{-\alpha(s-t)}\tilde{f}\left(s,\tilde{X}_s^{\tilde{x},\tilde{\pi},\gamma},\tilde{\pi}_s\gamma_s\right)ds+e^{-\alpha(T-t)}\tilde{g}\left(\tilde{X}_T^{\tilde{x},\tilde{\pi},\gamma}\right)\right], \tag{2.45}$$

for $(t,\tilde{x})\in[0,T]\times(0,\infty)$ with the initial condition $\tilde{X}_t^{\tilde{x},\tilde{\pi},\gamma}=\tilde{x}$. Note that the dual process $\tilde{X}^{\tilde{x},\tilde{\pi},\gamma}$ is only a dual with respect to the primal perpetual income process $X^{\pi,c,a}$, as opposed to the 'true' state process. The dual HJB equation is

$$\begin{aligned}&\frac{\partial\tilde{v}}{\partial t}(t,\tilde{x})-\alpha\tilde{v}(t,\tilde{x})+\tilde{U}(\tilde{x})+a\tilde{x}+(\alpha-r)\tilde{x}\partial_x\tilde{v}(t,\tilde{x})\\&+\inf_{\gamma>0}\{\eta v_0(t,\gamma)-\eta\gamma\partial_x\tilde{v}(t,\tilde{x})\}+\inf_{\tilde{x},\tilde{\pi}}\{-\delta_K(\tilde{\pi})\tilde{x}\tilde{v}_{\tilde{x}}+\frac{1}{2}|\theta+\Sigma^{-1}\tilde{\pi}|^2\tilde{x}^2\partial_{xx}^2\tilde{v}(t,\tilde{x})\}=0,\end{aligned} \tag{2.46}$$

for $(t,\tilde{x}) \in [0,T) \times (0,\infty)$, with the terminal condition $\tilde{v}(T,\tilde{x}) = \tilde{g}(\tilde{x})$. In particular, if K is a closed convex cone, then the dual HJB equation is reduced to a semilinear PDE

$$\begin{aligned}&\frac{\partial \tilde{v}}{\partial t}(t,\tilde{x}) - \alpha\tilde{v} + \tilde{U}(\tilde{x}) + a\tilde{x} + (\alpha - r)\tilde{x}\partial_x\tilde{v}(t,\tilde{x})\\&+ \inf_{\gamma>0}\{\eta v_0(t,\gamma) - \eta\gamma\partial_x\tilde{v}(t,\tilde{x})\} + \frac{1}{2}|\theta + \Sigma^{-1}\tilde{\pi}^*|^2\tilde{x}^2\partial^2_{xx}\tilde{v}(t,\tilde{x}) = 0\end{aligned}$$

for $(t,\tilde{x}) \in [0,T) \times (0,\infty)$, with the terminal condition $\tilde{v}(T,\tilde{x}) = \tilde{U}(\tilde{x})$, where $\tilde{\pi}^*$ is the minimum point of $|\theta + \Sigma^{-1}\tilde{\pi}|^2$ over $\tilde{\pi} \in \tilde{K} = \{\tilde{\pi} : \tilde{\pi}^\top\pi \geq 0, \forall\pi \in K\}$, the positive polar cone of K.

Remark 7. This HJB equation covers both the pre stopping and post stopping problem. Indeed, by taking $\eta = 0$ in (2.46) we recover the special case of perpetual income, and additionally taking $a = 0$ recovers the standard no income dual HJB. Note further that taking $\eta = 0$ removes dependence on γ in the dual HJB and dual state (2.44). This is to be expected, as γ does not appear in the deterministic income problems.

Remark 8. The dual formulation and problem also cover the infinite horizon $T = \infty$ case. In this setting, we need to assume that the stock and interest rate are stationary, so r, μ and Σ must be constants. In this case, the value function (1.3) loses its dependence on time. The time derivative term in the HJB equation (1.7) is then 0, and the terminal condition disappears. The derivation of the dual problem is very similar, except we must assume that

$$\lim_{t\to\infty}\mathbb{E}\left[e^{-\alpha t}X_t^{\pi,c,a}\tilde{X}_t^{\tilde{x},\tilde{\pi},\gamma}\right] = 0. \tag{2.47}$$

We no longer have the term $e^{-\alpha T}U_1(X_T^{\pi,c,a})$ in (2.39), so we add the left hand side of (2.47) instead. We can then use the definition (2.38) of L to recover the lost terms. See Davey et al. (2021) for more details.

The value function v is the conjugate function of the dual value function $\tilde{v}$, given by

$$u(t,x) := \sup_{\tilde{x}>0}\{\tilde{v}(t,\tilde{x}) + x\tilde{x}\}, \qquad (t,x) \in [0,T] \times (0,\infty).$$

The optimal wealth at time t is given by

$$X_t^* = -\partial_{\tilde{x}}\tilde{v}\left(t,\tilde{X}_t^*\right), \qquad 0 \leq t \leq T,$$

where $\tilde{X}^*$ is the optimal dual state process starting from $\tilde{X}_0^* = \tilde{x}^*$ with $\tilde{x}^*$ the solution of the equation $\partial_{\tilde{x}}\tilde{v}\,(0, \tilde{x}) + x_0 = 0$, and x_0 is the initial wealth at time 0.

Chapter 3

Convergence of the PPGM algorithm for the LQ Problem

In this chapter we analyse the convergence of the proximal policy gradient method (PPGM) for continuous time stochastic control problems. PPGM methods can be considered an extension of regular gradient based methods for problems with a 'difficult' running cost. In our setting the complexity occurs from an infinite penalty function, representing the control constraints of the problem. We consider the case where the diffusion function of the state process depends explicitly on the control process. Convergence results for this class of problems are not available in the literature, due to the complex dependence of the control process on the adjoint BSDE solutions for the control problem. Additional issues arise from the non-convexity of the cost function with respect to the control. We bridge the gap by deriving some sufficient conditions under which the PPGM algorithm converges for the LQ problem. We choose to restrict to this problem as it simplifies the analysis, while still sufficiently characterising the issues of PPGM convergence.

We study numerical methods for the Linear-Quadratic (LQ) problem. The LQ problem has been well studied in the literature, see Sun et al. (2016) for exposition. In Wu et al. (2020) it is shown that the problem reduces to solving two Ricatti ODEs to find the optimal control and value function, within a certain class of constraint spaces. However, convergence results of a PPGM algorithm for this problem in a general constraint space remain elusive, so it is still of mathematical interest to study this problem, and the Ricatti equation solving method allows us to determine benchmark solutions to numerically test our algorithm.

The PPGM approach to the LQ problem has been studied in various forms in the literature, see for example Carmona et al. (2019); Cassel and Koren (2021); Fazel et al. (2018); Hambly et al. (2021a,b); Wang et al. (2021). Typically the policy gradient method (PGM) is studied, where the control is unconstrained. In Fazel et al. (2018) converge results are obtained for an LQ problem in discrete time with no diffusion function for the state. These results are extended in Cassel and Koren (2021); Hambly et al. (2021b) to a diffusion function, that is still independent of state and control. These methods have also been applied to mean-field games, for example Hambly et al. (2021a); Wang et al. (2021)), the latter in continuous time. The method of PGM for LQ problems has also been studied in the context of machine learning, where problems in reinforcement learning can be approximated by LQ problems, see for example Wang et al. (2020).

The aim of this chapter is to extend the results of Reisinger et al. (2022) to a more general case. In Reisinger et al. (2022), convergence of a PPGM method is shown for a class of control problems with general drift, diffusion and cost functions. Our treatment of the LQ problem is not as general in its self, but is one example of a problem that does not satisfy the assumptions of Reisinger et al. (2022). To be specific, there is control in the diffusion of our state process, and the running and terminal cost functions do not have bounded derivatives. The key extension that this chapter provides is an analysis of regularity of both elements of the BSDE pair solving the adjoint equation.

The PPGM method has been studied for the LQ problem in a recent paper Giegrich et al. (2022). In this paper the authors consider the unconstrained problem in a relaxed control framework. Unlike our work, they introduce an entropy regulariser, for which the optimal control is a Guassian, as opposed to a deterministic function of the state. In this setting they give convergence results similar to ours, removing some of the assumptions in our unconstrained analysis by adapting the gradient step to be ‘geometry aware’. The key difference to our work is that all analysis occurs for the unconstrained case, and it is not immediate if this analysis extends to our more general constrained setting.

There are a number of iterative methods for continuous stochastic control problems in the literature where control is in the diffusion of the state process. For example, convergence the method of successive approximations has been established in this setting Kerimkulov et al.

(2021). In this method, the Hamiltonian is directly minimised at each iteration step. This contrasts from our method, where only a single step of gradient descent is taken. Numerically, our method is advantageous as less time is taken optimising a (potentially) incorrect Hamiltonian at early iteration steps. Convergence analysis is performed in Kerimkulov et al. (2021) but, in a similar vein to Reisinger et al. (2022), the running and cost functions are assumed to have bounded derivatives, not allowing the LQ problem. A recent paper Zhou and Lu (2023) considers the PGM method for control problems with control in the diffusion. They show sufficient conditions under which convergence of the cost function can be established for unconstrained minimisation of a more general state process that is constrained to a torus.

The main contributions of this chapter are as follows. We provide sufficient conditions under which the PPGM algorithm will converge linearly for the LQ problem in continuous time. We provide analysis for this problem as it is not currently treated in the literature, as the diffusion of the state process is control dependent. We relax the more technical assumptions for the unconstrained LQ problem, and give explicit convergence rates in terms of the coefficients of the problem. We perform numerical analysis to test how the theoretical convergence rates compare to those seen in application of the PPGM algorithm.

This chapter is organised as follows. Sections 3.1 and 3.2 describe the LQ problem and the algorithmic solving method respectively. Section 3.3 describes the main result, and Sections 3.4 and 3.5 derive sufficient conditions for convergence in the constrained and unconstrained setting respectively.

3.1 Problem Formulation

Let $W = (W_t)_{0\le t\le T}$ be a standard 1-dimensional Brownian motion on the natural filtered probability space $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0\le t\le T}, \mathbb{P}\right)$ augmented with all $\mathbb{P}$-null sets, and T a fixed finite horizon. We are considering this one dimensional case simply for notational brevity, all results in the sequel extend to the multi-dimensional Brownian motion case. We consider the following stochastic linear-quadratic problem

$$\inf_{u\in\mathcal{U}} \mathbb{E}\left[\int_0^T \frac{1}{2}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix}^\top \begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix} \begin{pmatrix} X_s^u \\ u_s \end{pmatrix} ds + \frac{1}{2}\left(X_T^u\right)^\top G X_T^u\right], \tag{3.1}$$

where the state control pair $(X^u, u) \in \mathcal{S}^2(\mathbb{R}^n) \times \mathcal{H}^2(\mathbb{R}^m)$ are determined by

$$dX_s^u = \left(A_s X_s^u + B_s u_s\right) ds + \left(C_s X_s^u + D_s u_s\right) dW_s, \qquad 0 \leq s \leq T \tag{3.2}$$

$$X_0^u = x_0, \tag{3.3}$$

for some given $x_0 \in \mathbb{R}^n$. The coefficients $A \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times n}\right)$, $B \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times m}\right)$, $C \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times n}\right)$, $D \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times m}\right)$, $Q \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times n}\right)$, $R \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m\times m}\right)$, $S \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m\times n}\right)$ and $G \in \mathbb{R}^{n\times n}$ are deterministic, with the matrices

$$\begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix}, \qquad 0 \leq s \leq T,$$

and G positive semi-definite. Without loss of generality, we assume R, Q and G are symmetric, so $R_s = R_s^\top$, $Q_s = Q_s^\top$ for all $s \in [0,T]$ and $G = G^\top$. The set of admissible controls is given by $\mathcal{U} = \mathcal{H}^2(U)$ for some closed convex set $U \subset \mathbb{R}^m$. With this integrability condition, and the continuity (hence uniform bound on the closed set $[0,T]$) of A, B, C and D, we have $X^u \in \mathcal{S}^2(\mathbb{R}^n)$ for all $u \in \mathcal{U}$ (Zhang, 2017, Theorem 3.2.2).

We assume here that the coefficients are deterministic and continuous in time, to allow us to describe Malliavin derivatives in the sequel. We choose the LQ problem here to simplify discussion, many of the results will extend to non linear models as long as the drift and diffusion are Lipschitz functions of the state and control. Furthermore, the running cost can also be changed to a differentiable function (in x and u) that is strongly convex in the control, with Lipschitz derivative in both x and u.

3.2 Proximal Policy Gradient Method

To deal with the control constraint define the cost functional J on the set $\mathcal{H}^2(\mathbb{R}^m)$ by

$$J(u) := \mathbb{E}\left[\int_0^T \left(\frac{1}{2}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix}^\top \begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix} + \Phi_U(u_s)\right) ds + \frac{1}{2}(X_T^u)^\top G X_T^u\right], \tag{3.4}$$

for $u \in \mathcal{H}^2(\mathbb{R}^m)$, recalling that $\Phi_U(z)$ is the infinite penalty function that equals 0 if $z \in U$ and ∞ otherwise. The function Φ_U may be non differentiable, so J may be non-smooth and non-convex. Therefore we use sub-differentials to describe stationary points of J.

Definition 3.2.1. Let $(X, ||\cdot||_X)$ be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_X \colon X^2 \to \mathbb{R}\cup\{\infty\}$. Let $F\colon X \to \mathbb{R}\cup\{\infty\}$. For $x^* \in \{x \in X | F(x) < \infty\}$ we define the Fréchet sub-differential of F at x^* by

$$\partial F(x^*) := \left\{\bar{x} \in X \middle| \liminf_{x \to x^*} \frac{F(x) - F(x^*) - \langle \bar{x}, x - x^*\rangle_X}{||x - x^*||_X} \geq 0\right\}.$$

We will characterise an optimal control as a process $u^* \in \mathcal{H}^2(\mathbb{R}^m)$ such that $J(u^*) < \infty$ and $0 \in \partial J(u^*)$. To find the optimal control, we will use the (generalised) Hamiltonian $H\colon [0,T] \times \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ which is given by the following.

$$H_t(x, u, y, z) := y^\top (A_t x + B_t u) + z^\top (C_t x + D_t u) + \frac{1}{2}u^\top R_t u + u^\top S_t x + \frac{1}{2}x^\top Q_t x, \tag{3.5}$$

for $t \in [0,T]$, $u \in \mathbb{R}^m$, and $x, y, z \in \mathbb{R}^n$. In our scheme we require the derivative of H with respect to the control and state, which are given by

$$\begin{aligned}\partial_u H_t(x, u, y, z) &= B_t^\top y + D_t^\top z + R_t u + S_t x, \\ \partial_x H_t(x, u, y, z) &= A_t^\top y + C_t^\top z + S_t^\top u + Q_t x,\end{aligned}$$

for $t \in [0,T]$, $u \in \mathbb{R}^m$, and $x, y, z \in \mathbb{R}^n$. The PGM method uses gradient descent with respect to the Hamiltonian. Usually (for example, in the HJB equation) the Hamiltonian would take the first and second derivatives of the value function - if it is indeed differentiable - as input. We do not know these a priori, and will instead input solutions to a BSDE, which act as estimates for these quantities. To deal with the constraint function, we introduce the proximal map

$\mathrm{prox}_U \colon \mathbb{R}^m \to \mathbb{R}^m$ as

$$\mathrm{prox}_U(u) := \arg\min_{z \in U} \left\{ \frac{1}{2}|z-u|^2 \right\}, \qquad u \in \mathbb{R}^m. \tag{3.6}$$

For this map to be well defined, we naturally assume that U is non empty. The algorithm is focused on finding feedback controls $\phi \colon [0,T] \times \mathbb{R}^n \to U$. Let $\mathcal{B}([0,T] \times \mathbb{R}^n; U)$ be the space of such measurable functions. Define, for $\phi \in \mathcal{B}([0,T] \times \mathbb{R}^n; U)$,

$$\begin{aligned}
||\phi||_\infty &:= \sup_{(t,x)\in[0,T]\times\mathbb{R}^n} |\phi_t(x)|, \\
|\phi|_0 &:= \sup_{(t,x)\in[0,T]\times\mathbb{R}^n} \frac{|\phi_t(x)|}{1+|x|}, \\
[\phi]_1 &:= \sup_{(t,x,y)\in[0,T]\times\mathbb{R}^n\times\mathbb{R}^n,\, x\neq y} \frac{|\phi_t(x)-\phi_t(y)|}{|x-y|}.
\end{aligned}$$

Then define

$$\mathcal{V} := \{\phi \in \mathcal{B}([0,T]\times\mathbb{R}^n; U) : |\phi|_0 + [\phi]_1 < \infty\}.$$

For $\phi \in \mathcal{V}$ define the associated control $u^\phi \in \mathcal{H}^2(U)$ by $u_t^\phi = \phi_t\left(X_t^\phi\right)$ where,

$$\begin{aligned}
dX_s^\phi &= \left(A_s X_s + B_s \phi_s\left(X_s^\phi\right)\right) ds + \left(C_s X_s + D_s \phi_s\left(X_s^\phi\right)\right) dW_s, \qquad 0 \le s \le T, \\
X_0^\phi &= x_0.
\end{aligned} \tag{3.7}$$

The motivation for our iterative algorithm is the Stochastic Maximum Principle, Theorem 1.5.1. Since H defined in (3.5) is a quadratic function of u, if we assume R_t is invertible for all $t \in [0,T]$ and our control space is unconstrained then we can solve (1.12) explicitly to yield

$$u_t^* = -R_t^{-1}\left(B_t^\top Y_t^{u^*} + D_t^\top Z_t^{u^*} + S_t X_t^{u^*}\right), \qquad 0 \le t \le T. \tag{3.8}$$

Substituting (3.8) into (1.10) and (3.2) allows us to solve the problem numerically by solving the FBSDE. However, this is a fully coupled FBSDE and the method for solving is computationally complex, particularly for high dimensional problems. The PPGM alleviates this complexity by instead solving a series of decoupled FBSDEs, and updating the control based on a gradient descent algorithm for the Hamiltonian. We give the details now.

For some arbitrary initial guess $\phi_0 \in \mathcal{V}$, we can define an iterative scheme as follows, for all

$k \in \mathbb{N}_0$ and $(t,x) \in [0,T] \times \mathbb{R}^n$.

$$\phi_t^{k+1}(x) := \operatorname{prox}_U \left(\phi_t^k(x) - \tau \left(B_t^\top Y_t^{t,x,\phi^k} + D_t^\top Z_t^{t,x,\phi^k} + R_t \phi_t^k(x) + S_t x \right) \right), \tag{3.9}$$

where for given $\phi \in \mathcal{V}$ the processes $\left(X^{t,x,\phi},\, Y^{t,x,\phi},\, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{H}^2(t,T;\mathbb{R}^n)$ are determined by the following FBSDE, for $s \in [t,T]$

$$\begin{aligned}
dX_s^{t,x,\phi} &= \left(A_s X_s^{t,x,\phi} + B_s \phi_s\left(X_s^{t,x,\phi}\right)\right) ds + \left(C_s X_s^{t,x,\phi} + D_s \phi_s\left(X_s^{t,x,\phi}\right)\right) dW_s,\\
X_t^{t,x,\phi} &= x,\\
dY_s^{t,x,\phi} &= -\left(A_s^\top Y_s^{t,x,\phi} + C_s^\top Z_s^{t,x,\phi} + S_s^\top \phi_s\left(X_s^{t,x,\phi}\right) + Q_s X_s^{t,x,\phi}\right) ds + Z_s^{t,x,\phi} dW_s,\\
Y_T^{t,x,\phi} &= G X_T^{t,x,\phi}.
\end{aligned} \tag{3.10}$$

Definition 3.2.2 (Proximal PGM Scheme)**.** Define the updating function $\mathcal{L}\colon \mathcal{V} \to \mathcal{B}([0,T] \times \mathbb{R}^n; U)$ by

$$\mathcal{L}(\phi)_t(x) := \operatorname{prox}_U \left(\phi_t(x) - \tau \left(B_t^\top Y_t^{t,x,\phi} + D_t^\top Z_t^{t,x,\phi} + R_t \phi_t(x) + S_t x \right) \right), \tag{3.11}$$

for $\phi \in \mathcal{V}$, $t \in [0,T]$ and $x \in \mathbb{R}^n$, where $\left(X^{t,x,\phi},\, Y^{t,x,\phi},\, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{H}^2(t,T;\mathbb{R}^n)$ are solutions to (3.10). Fix $\phi^0 \in \mathcal{V}$, then for $k \in \mathbb{N}_0$ define our PPGM scheme $\left(\phi^k\right)_{k \in \mathbb{N}_0}$ by

$$\phi_t^{k+1}(x) = \mathcal{L}\left(\phi^k\right)_t(x), \tag{3.12}$$

for $t \in [0,T]$ and $x \in \mathbb{R}^n$.

3.3 Main Result

The main result we wish to show is that the scheme in (3.12) is a contraction mapping in $\mathcal{V}$, so the sequence $\left(\phi^k\right)$ converges to some element $\phi^* \in \mathcal{V}$, with respect to the $|\cdot|_0$ norm. We then wish to show that the corresponding control u^{ϕ^*} is in fact an optimal control. Throughout this section and the proceeding section, many of the Theorems we state are similar to those presented in Reisinger et al. (2022). Our proofs are adapted from these, to consider a more general case with control in the diffusion of the state process.

Theorem 3.3.1. *Let $\mathcal{V}_L$ satisfy Assumptions 2, 3, and 4. Let $\phi^0 \in \mathcal{V}_L$ satisfy (3.24) and (3.44). Let $\tau > 0$ satisfy Assumption 1. Then there exists $\phi^* \in \mathcal{V}$ with associated control $u^{\phi^*} \in \mathcal{H}^2(U)$, and constants $\hat{c} \in [0,1)$ and $\hat{C} \geq 0$ such that*

1. *For all $k \in \mathbb{N}_0$, $|\phi^k - \phi^*|_0 \leq \hat{c}^k |\phi^* - \phi^0|_0$.*
2. *For all $k \in \mathbb{N}_0$, $\|u^{\phi^k} - u^{\phi^*}\|_{\mathcal{H}^2} \leq \hat{C}\hat{c}^k |\phi^* - \phi^0|_0$.*
3. *u^{ϕ^*} is a stationary point of J defined in (3.4).*

We will use convex analysis to characterise a stationary point of J as a fixed point of (3.12). To establish the existence of ϕ^*, we will use Banach's fixed point theorem, applied to the space $(\mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m), |\cdot|_0)$. The following Lemma allows us to do this.

Lemma 3.3.2. *The space $(\mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m), |\cdot|_0)$ is complete.*

Proof. For any $f \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$, define the functions $\tilde{f}, \bar{f} \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$ by, for $(t,x) \in [0,T] \times \mathbb{R}^n$,

$$\tilde{f}_t(x) = \frac{f_t(x)}{1 + |x|},$$
$$\bar{f}_t(x) = f_t(x)(1 + |x|).$$

Suppose that a sequence $(f^k)_{k \in \mathbb{N}_0}$ in $\mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$ satisfies $\lim_{k,j \to \infty} |f^k - f^j|_0 = 0$. We need to show that there exists $f^\infty \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$ such that $\lim_{k \to \infty} |f^\infty - f^k|_0 = 0$. By Definition of $|\cdot|_0$, we have

$$0 = \lim_{k,j \to \infty} \sup_{(t,x) \in [0,T] \times \mathbb{R}^n} \frac{|f_t^k(x) - f_t^j(x)|}{1 + |x|} = \lim_{k,j \to \infty} \sup_{(t,x) \in [0,T] \times \mathbb{R}^n} |\tilde{f}_t^k(x) - \tilde{f}_t^j(x)|$$

Hence $\lim_{k,j \to \infty} \|\tilde{f}^k - \tilde{f}^j\|_\infty = 0$. As $(\mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m), \|\cdot\|_\infty)$ is complete, there exists $f^* \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$ such that $\lim_{k \to \infty} \|f^* - \tilde{f}^k\|_\infty = 0$, that is,

$$0 = \lim_{k \to \infty} \sup_{(t,x) \in [0,T] \times \mathbb{R}^n} |f_t^*(x) - \tilde{f}_t^k(x)| = \lim_{k \to \infty} \sup_{(t,x) \in [0,T] \times \mathbb{R}^j} \frac{|\bar{f}_t^*(x) - f_t^k(x)|}{1 + |x|}.$$

Hence $\lim_{k \to \infty} |\bar{f}^* - f^k|_0 = 0$, as required. $\square$

To use Banach's fixed point theorem to show existence of a limiting control $\phi^* \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$, we must show two things: that the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ is contained within $\mathcal{V}$, and that the mapping $\phi^k \mapsto \phi^{k+1}$ defined in (3.12) is a contraction. To be specific, we wish to show that for all $k \in \mathbb{N}$

$$|\phi^k|_0 + [\phi^k]_1 < \infty, \tag{3.13}$$

$$|\phi^{k+1} - \phi^k|_0 \leq \hat{c}|\phi^k - \phi^{k-1}|_0. \tag{3.14}$$

To prove these statements, we have to perform some Lipschitz analysis, to show the function $\phi \mapsto B_t^\top Y_t^{t,x,\phi} + D_t^\top Z_t^{t,x,\phi}$ is Lipschitz with a sufficiently small Lipschitz coefficient with respect to both x and ϕ, uniformly in $t \in [0,T]$. We will use the following Theorem.

Assumption 1. There exists $\mu > 0$ such that, for all $t \in [0,T]$

$$R_t \succ \mu \mathbb{1}_{m,m}, \qquad \tau < \min\left(\frac{2}{\mu + \|R\|_\infty}, \frac{\mu + \|R\|_\infty}{2\mu\|R\|_\infty}\right),$$

where $\mathbb{1}_{m,m}$ is the identity matrix of size m.

Remark 9. Our PPGM framework exists in continuous time, and this forms the basis of our assumption on the learning rate. However, as is seen in Giegrich et al. (2022) the learning rate may need to be adapted to the mesh size of the discretisation that is used in the implementation of this algorithm.

Theorem 3.3.3. *Define a constant*

$$c_0 := \frac{1}{2}\frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}.$$

Then for all $t \in [0,T]$, $\phi_1, \phi_2 \in \mathcal{V}$ and $x_1, x_2 \in \mathbb{R}^n$

$$\begin{aligned}
|\mathcal{L}\left(\phi^1\right)_t(x_1) - \mathcal{L}\left(\phi^2\right)_t(x_2)| \leq{}& (1 - 2\tau c_0)\left|\phi_t^1(x_1) - \phi_t^2(x_2)\right| \\
&+ \tau\|S\|_\infty |x_1 - x_2| \\
&+ \tau\|B\|_\infty \left|Y_t^{t,x_1,\phi^1} - Y_t^{t,x_2,\phi^2}\right| \\
&+ \tau\|D\|_\infty \left|Z_t^{t,x_1,\phi^1} - Z_t^{t,x_2,\phi^2}\right|.
\end{aligned}$$

Proof. Fix $t \in [0,T]$, $\phi_1, \phi_2 \in \mathcal{V}_L$ and $x_1, x_2 \in \mathbb{R}^n$. The projection prox_U is 1-Lipschitz. Hence we have by the definition (3.11) of $\mathcal{L}$,

$$\begin{aligned}
&|\mathcal{L}\left(\phi^1\right)_t(x_1) - \mathcal{L}\left(\phi^2\right)_t(x_2)| \\
&\leq \Big| \left(\phi^1_t(x_1) - \tau\left(B_t^\top Y_t^{t,x_1,\phi^1} + D_t^\top Z_t^{t,x_1,\phi^1} + R_t\phi^1_t(x_1) + S_t x_1\right)\right) \qquad (3.15)\\
&-\left(\phi^2_t(x_2) - \tau\left(B_t^\top Y_t^{t,x_2,\phi^2} + D_t^\top Z_t^{t,x_2,\phi^2} + R_t\phi^2_t(x_2) + S_t x_2\right)\right)\Big| \\
&\leq \left|\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right) - \tau R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)\right| \qquad (3.16)\\
&+\tau\|B\|_\infty|Y_t^{t,x_1,\phi^1} - Y_t^{t,x_2,\phi^2}| + \tau\|D\|_\infty|Z_t^{t,x_1,\phi^1} - Z_t^{t,x_2,\phi^2}| + \tau\|S\|_\infty|x_1 - x_2|.
\end{aligned}$$

Define $f\colon \mathbb{R}^m \to \mathbb{R}$ by $f(u) = \frac{1}{2}u^\top R_t u$. Then $\nabla f(u) = R_t u$. By Assumption 1, f is μ-strongly convex in u and ∇f is $\|R\|_\infty$-Lipschitz in u. Hence by Lemma A.1.11 we have, for any $u_1, u_2 \in \mathbb{R}^m$

$$\begin{aligned}
&(u_1 - u_2)^\top\left(\nabla f(u_1) - \nabla f(u_2)\right) \\
&\geq \frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}|u_1 - u_2|^2 + \frac{1}{\mu + \|R\|_\infty}|\nabla f(u_1) - \nabla f(u_2)|^2.
\end{aligned} \tag{3.17}$$

Now, substituting in the definition of f into (3.17), we have

$$(u_1 - u_2)^\top R_t(u_1 - u_2) \geq \frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}|u_1 - u_2|^2 + \frac{1}{\mu + \|R\|_\infty}|R_t u_1 - R_t u_2|^2.$$

Therefore we have

$$\begin{aligned}
&\left|\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right) - \tau R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)\right|^2 \\
&= \left|\phi^1_t(x_1) - \phi^2_t(x_2)\right|^2 - 2\tau\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)^\top R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right) + \tau^2\left|R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)\right|^2 \\
&\leq \left(1 - \frac{2\tau\mu\|R\|_\infty}{\mu + \|R\|_\infty}\right)\left|\phi^1_t(x_1) - \phi^2_t(x_2)\right|^2 + \left(\tau^2 - \frac{2\tau}{\mu + \|R\|_\infty}\right)\left|R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)\right|^2 \\
&= \left(1 - \frac{2\tau\mu\|R\|_\infty}{\mu + \|R\|_\infty}\right)\left|\phi^1_t(x_1) - \phi^2_t(x_2)\right|^2 + \tau\left(\tau - \frac{2}{\mu + \|R\|_\infty}\right)\left|R_t\left(\phi^1_t(x_1) - \phi^2_t(x_2)\right)\right|^2 \\
&\leq \left(1 - \frac{2\tau\mu\|R\|_\infty}{\mu + \|R\|_\infty}\right)\left|\phi^1_t(x_1) - \phi^2_t(x_2)\right|^2,
\end{aligned}$$

where we have used that, by Assumption 1,

$$\tau - \frac{2}{\mu + \|R\|_\infty} < 0.$$

Taking the square root, and using $\sqrt{1-a} \leq 1 - \frac{a}{2}$ for $a = \tau \frac{2\mu\|R\|_\infty}{\mu+\|R\|_\infty}$, where $a \in [0,1]$ by Assumption 1, yields

$$\left|\left(\phi_t^1\left(x_1\right) - \phi_t^2\left(x_2\right)\right) - \tau R\left(\phi_t^1\left(x_1\right) - \phi_t^2\left(x_2\right)\right)\right| \leq \left(1 - \tau\frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}\right)\left|\phi_t^1\left(x_1\right) - \phi_t^2\left(x_2\right)\right|. \tag{3.18}$$

Substituting (3.18) into (3.16) yields

$$\begin{aligned}
&\left|\mathcal{L}\left(\phi^1\right)_t\left(x_1\right) - \mathcal{L}\left(\phi^2\right)_t\left(x_2\right)\right| \\
&\leq \left(1 - \tau\frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}\right)\left|\phi_t^1\left(x_1\right) - \phi_t^2\left(x_2\right)\right| \\
&+ \tau\|B\|_\infty|Y_t^{t,x_1,\phi^1} - Y_t^{t,x_2,\phi^2}| + \tau\|D\|_\infty|Z_t^{t,x_1,\phi^1} - Z_t^{t,x_2,\phi^2}| + \tau\|S\|_\infty|x_1 - x_2|,
\end{aligned}$$

as required. □

Given this Theorem, we have to control the quantities $|Y_t^{t,x_1,\phi_1} - Y_t^{t,x_2,\phi_2}|$ and $|Z_t^{t,x_1,\phi_1} - Z_t^{t,x_2,\phi_2}|$ in terms of $|x_1 - x_2|$ and $|\phi_1 - \phi_2|_0$ for any $x_1, x_2 \in \mathbb{R}^n$ and $\phi_1, \phi_2 \in \mathcal{V}$, for any $t \in [0,T]$. To do this we will need to use our BSDE and SDE estimate Theorems. In order to study $|Z_t^{t,x_1,\phi_1} - Z_t^{t,x_2,\phi_2}|$, we will describe Z using Malliavin Calculus.

3.4 Sufficient Conditions for the Constrained Case

3.4.1 Showing $[\phi^k]_1$ is Uniformly Bounded

We now want to find a recurrence relation for the sequence $\left([\phi^k]_1\right)_{k\in\mathbb{N}_0}$. To this end we study the Lipschitz coefficients of the Y and Z processes. We begin with Y. The following result is adapted from (Reisinger et al., 2022, Proposition 3.7). The core concept of the proof is the same, but the function L_Y is altered to account for the control in the diffusion of X.

Theorem 3.4.1. *For $\phi \in \mathcal{V}$, $t \in [0,T]$, $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{H}^2\left(t,T;\mathbb{R}^n\right)$ be solutions to (3.10). Then for all $\phi \in \mathcal{V}$, there exists $L_Y =$*

$L_Y([\phi]_1)$ *such that for all* $t \in [0,T]$, $x_1, x_2 \in \mathbb{R}^n$

$$|Y_t^{t,x_1,\phi} - Y_t^{t,x_2,\phi}| \leq L_Y([\phi]_1)\,|x_1 - x_2|.$$

Proof. Fix $\phi \in \mathcal{V}$, $t \in [0,T]$, $x_1, x_2 \in \mathbb{R}^n$. Define, for $s \in [t,T]$

$$\begin{aligned}
\Delta X_s &= X_s^{t,x_1,\phi} - X_s^{t,x_2,\phi},\\
\Delta Y_s &= Y_s^{t,x_1,\phi} - Y_s^{t,x_2,\phi},\\
\Delta Z_s &= Z_s^{t,x_1,\phi} - Z_s^{t,x_2,\phi},\\
\Delta \phi_s &= \phi_s\left(X_s^{t,x_1,\phi}\right) - \phi_s\left(X_s^{t,x_2,\phi}\right).
\end{aligned}$$

Then, using the definition (3.10) of the individual FBSDEs, we see that $(\Delta X, \Delta Y, \Delta Z)$ solve the following FBSDE, for $t \leq s \leq T$,

$$\begin{aligned}
d\Delta X_s &= \left(A_s \Delta X_s + B_s \Delta\phi_s\right) ds + \left(C_s \Delta X_s + D_s \Delta \phi_s\right) dW_s,\\
\Delta X_t &= x_1 - x_2,\\
d\Delta Y_s &= -\left(A_s^\top \Delta Y_s + C_s^\top \Delta Z_s + S_t^\top \Delta\phi_s + Q_s \Delta X_s\right) ds + \Delta Z_s^\top dW_s,\\
\Delta Y_T &= G \Delta X_T.
\end{aligned}$$

Define

$$\tilde{c}_1 = C_{\mathrm{BSDE}}\left(\|A\|_\infty, \|C\|_\infty\right).$$

By Theorem A.1.10, we have

$$\mathbb{E}\left[|\Delta Y_t|^2\right] \leq \tilde{c}_1 \mathbb{E}\left[|G|^2 |\Delta X_T|^2 + \left(\int_t^T |S_s^\top \Delta\phi_s + Q_s \Delta X_s| ds\right)^2\right).$$

Since ϕ is Lipschitz, we have $|\Delta\phi_s| \leq [\phi]_1 |\Delta X_s|$ for all $s \in [t,T]$. Therefore, by Holder

$$\begin{aligned}
\mathbb{E}\left[|\Delta Y_t|^2\right] &\leq \tilde{c}_1 \mathbb{E}\left[|G|^2 |\Delta X_T|^2 + \left(\int_t^T \left(\|S\|_\infty [\phi]_1 + \|Q\|_\infty\right) |\Delta X_s| ds\right)^2\right] && (3.19)\\
&\leq \tilde{c}_1 \mathbb{E}\left[|G|^2 |\Delta X_T|^2 + \left(\|S\|_\infty [\phi]_1 + \|Q\|_\infty\right)^2 T \int_t^T |\Delta X_s|^2 ds\right]. && (3.20)
\end{aligned}$$

We now study $\mathbb{E}\left[|\Delta X_s|^2\right]$ for fixed $s \in [t,T]$. Define

$$\tilde{c}_2 = C_{\mathrm{SDE}}\left(\|A\|_\infty, \|C\|_\infty\right),$$
$$\tilde{c}_3 = \frac{1}{2}T\left(\|B\|_\infty^2 T + \|D\|_\infty^2\right).$$

Applying Theorem A.1.9 to $(\Delta X_r)_{r\in[t,s]}$, we then have,

$$\begin{aligned}\mathbb{E}\left[|\Delta X_s|^2\right] &\leq \tilde{c}_2\mathbb{E}\left[|x_1 - x_2|^2 + \left(\int_t^s |B_r\Delta\phi_r| dr\right)^2 + \int_t^s |D_r\Delta\phi_r|^2 dr\right] \\ &\leq \tilde{c}_2\mathbb{E}\left[|x_1 - x_2|^2 + [\phi]_1^2\left(\|B\|_\infty^2 T + \|D\|_\infty^2\right)\int_t^s |\Delta X_r|^2 ds\right].\end{aligned}$$

Hence by gronwall we have

$$\mathbb{E}\left[|\Delta X_s|^2\right] \leq \tilde{c}_2 e^{2\tilde{c}_3[\phi]_1^2}|x_1 - x_2|^2. \tag{3.21}$$

Substituting (3.21) into (3.20) yields

$$\begin{aligned}\mathbb{E}\left[|\Delta Y_t|^2\right] &\leq \tilde{c}_1|G|^2\tilde{c}_2 e^{2\tilde{c}_3[\phi]_1^2}|x_1 - x_2|^2 \\ &\quad + \tilde{c}_1\tilde{c}_2\left(\|S\|_\infty[\phi]_1 + \|Q\|_\infty\right)^2 T\int_t^T e^{2\tilde{c}_3T[\phi]_1^2} ds|x_1 - x_2|^2 \\ &\leq \tilde{c}_1\tilde{c}_2\left(|G|^2 + \left(\|S\|_\infty[\phi]_1 + \|Q\|_\infty\right)^2 T^2\right)e^{2\tilde{c}_3T[\phi]_1^2}|x_1 - x_2|^2.\end{aligned}$$

Note that by Theorem A.1.4, there exists a measurable mapping $v_s^\phi(x) = Y_s^{t,x,\phi}$. In particular for each $i = 1, 2$, the variable $Y_t^{t,x_i,\phi} = v_t^\phi(x_i)$ is deterministic by Blumenthal's 0-1 Law. Hence we may write, using the fact that $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \geq 0$,

$$|\Delta Y_t| = \mathbb{E}\left[|\Delta Y_t|^2\right]^{\frac{1}{2}} \leq L_Y\left([\phi]_1\right)|x_1 - x_2|,$$

where

$$
\begin{aligned}
L_Y\left([\phi]_1\right) &= \sqrt{\tilde{c}_1\tilde{c}_2}\left(\tilde{c}_4+\tilde{c}_5T[\phi]_1\right)e^{\tilde{c}_3T[\phi]_1^2},\\
\tilde{c}_1 &= C_{\mathrm{BSDE}}\left(\|A\|_\infty,\|C\|_\infty\right),\\
\tilde{c}_2 &= C_{\mathrm{SDE}}\left(\|A\|_\infty,\|C\|_\infty\right),\\
\tilde{c}_3 &= \frac{1}{2}\left(\|B\|_\infty^2T+\|D\|_\infty^2\right),\\
\tilde{c}_4 &= |G|+\|Q\|_\infty T,\\
\tilde{c}_5 &= \|S\|_\infty,
\end{aligned}
$$

where C_{SDE} and C_{BSDE} are defined in Theorems A.1.9 and A.1.10 respectively. □

We now want to claim the same result holds for the Z process. However given the definition of Z in (3.10), it is not clear how to prove any estimates for $\mathbb{E}\left[\sup_{t\in[0,T]}|Z_t|^2\right]$, only integrals of the process $|Z|^2$ (e.g., using Theorem A.1.10 for (Y,Z)). Instead, we apply Malliavin calculus to show that Z can be written as a measurable function of the X process and its Malliavin derivative, and bound the Lipschitz dependence of this function on the initial data.

Lemma 3.4.2. *Fix $t\in[0,T]$, $x\in\mathbb{R}^n$, and $\phi\in\mathcal{V}$. Then $X_s^{t,x,\phi}\in\mathbb{D}^{1,2}\left(\mathbb{R}^n\right)$ for all $s\in[t,T]$.*

Proof. See appendix. □

Lemma 3.4.3. *Fix $t\in[0,T]$, $x\in\mathbb{R}^n$, and $\phi\in\mathcal{V}$. Then $Y_s^{t,x,\phi},Z_s^{t,x,\phi}\in\mathbb{D}^{1,2}\left(\mathbb{R}^n\right)$ for all $s\in[t,T]$.*

Proof. See appendix. □

We use Lemmas 3.4.2 and 3.4.3 to find Lipschitz estimates for the differences of the process $Z^{t,x,\phi}$ for different $x\in\mathbb{R}^n$.

Assumption 2. We assume the following.

1. There exists non empty $\mathcal{V}_L\subset\mathcal{V}$ and an increasing non negative function $C_L\colon[0,\infty)\to[0,\infty)$ such that for all $\phi\in\mathcal{V}_L$, $0\le t\le s\le T$

$$
\left|\delta v_t^\phi\left(x_1\right)-\delta v_t^\phi\left(x_2\right)\right|\left|C_tx_2+D_t\phi_t\left(x_2\right)\right|\le C_L\left([\phi]_1\right)|x_1-x_2|.
$$

2. There exists $\phi \in \mathcal{V}_L$ such that the sequence $\left(\phi^k\right)_{k \in \mathbb{N}_0}$ generated by (3.12) with $\phi^0 = \phi$ is contained in $\mathcal{V}_L$.

Remark 10. Assumption 2 holds in the unconstrained case, with $\mathcal{V}_L$ the space of linear functions. Indeed, let $\phi_s(x) = \alpha_s x + \beta_s$, $s \in [0,T]$, for some $\alpha \in \mathcal{C}^0\left([0,T]; \mathbb{R}^{m \times n}\right)$ and $\beta \in \mathcal{C}^0\left([0,T]; \mathbb{R}^m\right)$. Then v^ϕ is of the form $v_s^\phi(x) = a_s x + b_s$, $s \in [t,T]$, for some $a \in C^0\left([t,T]; \mathbb{R}^{n \times n}\right)$ and $b \in C^0\left([t,T]; \mathbb{R}^n\right)$. In particular, we have for any $x_1, x_2 \in \mathbb{R}^n$,

$$\delta v_t^\phi(x_1) - \delta v_t^\phi(x_2) = a_t - a_t = 0.$$

The assumption therefore holds with $C_L([\phi]_1) = 0$. In the constrained case, this assumption inequality would hold if, for example, $C = 0$, U is a bounded space, and $\mathcal{V}_L$ was the set

$$\mathcal{V}_L = \{\phi \in \mathcal{V} \colon v_t^\phi \in \mathcal{C}_b^2\left(\mathbb{R}^n; \mathbb{R}^n\right) \text{ for all } t \in [0,T]\}.$$

However, it is not clear in this case if Assumption 2 is satisfied.

Theorem 3.4.4. *Let $\phi \in \mathcal{V}_L$, $t \in [0,T]$, and $x_1, x_2 \in \mathbb{R}^n$. Then there exists $L_Z([\phi]_1) \geq 0$ such that*

$$|Z_t^{t,x_1,\phi} - Z_t^{t,x_2,\phi}| \leq L_Z([\phi]_1)\,|x_1 - x_2|.$$

Proof. Let $t \in [0,T]$ and $\phi \in \mathcal{V}$. By Theorem A.1.4, there exists $v^\phi \colon \Omega \times [t,T] \times \mathbb{R}^n \to \mathbb{R}^n$ such that, for $s \in [t,T]$ and $x \in \mathbb{R}^n$

$$v_s^\phi\left(X_s^{t,x,\phi}\right) = Y_s^{t,x,\phi}.$$

By Theorem 3.4.6, the function v_t^ϕ is deterministic and Lipschitz in x with coefficient $L_Y([\phi]_1)$. Furthermore, by Theorem 1.6.4, (A.2) and the chain rule we have, for any $s \in [t,T]$ and $x \in \mathbb{R}^n$

$$Z_s^{t,x,\phi} = D_t Y_s^{t,x,\phi} = D_t v_t^\phi\left(X_s^{t,x,\phi}\right) = \delta v_t^\phi\left(X_s^{t,x,\phi}\right)\mathcal{D}_s X_s^{t,x,\phi}.$$

In particular, at $s = t$ by (A.2) we have

$$Z_t^{t,x,\phi} = \delta v_t^\phi(x)\,\mathcal{D}_t X_t^{t,x,\phi} = \delta v_t^\phi(x)\left(C_t x + D_t \phi_t(x)\right). \tag{3.22}$$

Note by 3.4.1 that the function δv_t^ϕ is uniformly bounded by $L_Y([\phi]_1)$ Therefore, for any

$x_1, x_2 \in \mathbb{R}^n$ we have

$$\begin{aligned}
\left| Z_t^{t,x_1,\phi} - Z_t^{t,x_2,\phi} \right| &= \left| \delta v_t^{\phi}(x_1)\left(C_t x_1 + D_t \phi_t(x_1)\right) - \delta v_t^{\phi}(x_2)\left(C_t x_2 + D_t \phi_t(x_2)\right) \right| \\
&\leq \left| \delta v_t^{\phi}(x_1) \right| \left| C_t (x_1 - x_2) + D_t\left(\phi_t(x_1) - \phi_t(x_2)\right) \right| \\
&+ \left| \delta v_t^{\phi}(x_1) - \delta v_t^{\phi}(x_2) \right| \left| C_t x_2 + D_t \phi_t(x_2) \right| \\
&\leq L_Y([\phi]_1)\left(\|C\|_\infty + \|D\|_\infty [\phi]_1\right) |x_1 + x_2| \\
&+ \left| \delta v_t^{\phi}(x_1) - \delta v_t^{\phi}(x_2) \right| \left| C_t x_2 + D_t \phi_t(x_2) \right|.
\end{aligned}$$

Under Assumption 2, for $\phi \in \mathcal{V}_L$ we have

$$\begin{aligned}
\left| Z_t^{t,x_1,\phi} - Z_t^{t,x_2,\phi} \right| &\leq L_Y([\phi]_1)\left(\|C\|_\infty + \|D\|_\infty [\phi]_1\right) |x_1 + x_2| + C_L([\phi]_1)|x_1 - x_2| \\
&= L_Z([\phi]_1)|x_1 - x_2|,
\end{aligned}$$

where

$$L_Z([\phi]_1) = L_Y([\phi]_1)\left(\|C\|_\infty + \|D\|_\infty [\phi]_1\right) + C_L([\phi]_1).$$

□

In order to show the sequence of Lipschitz constants are a contraction, we first show they are uniformly bounded. We have the following

$$\begin{aligned}
L_Y([\phi]_1) &= \sqrt{\tilde{c}_1 \tilde{c}_2}\left(\tilde{c}_4 + \tilde{c}_5 T [\phi]_1\right) e^{\tilde{c}_3 T [\phi]_1^2}, \\
L_Z([\phi]_1) &= \sqrt{\tilde{c}_1 \tilde{c}_2}\left(\tilde{c}_4 + \tilde{c}_5 T [\phi]_1\right) e^{\tilde{c}_3 T [\phi]_1^2}\left(\|C\|_\infty + \|D\|_\infty [\phi]_1\right) + C_L([\phi]_1).
\end{aligned}$$

Define constants

$$\begin{aligned}
\tilde{c}_6 &= \sqrt{\tilde{c}_1 \tilde{c}_2}\,\tilde{c}_4, \\
\tilde{c}_7 &= \sqrt{\tilde{c}_1 \tilde{c}_2}\,\tilde{c}_5 T, \\
\tilde{c}_8 &= \tilde{c}_6 \|C\|_\infty + \frac{1}{2}\left(\tilde{c}_6 \|C\|_\infty + \tilde{c}_7 \|D\|_\infty\right)^2, \\
\tilde{c}_9 &= \tilde{c}_7 \|D\|_\infty + \frac{1}{2}\left(\tilde{c}_6 \|C\|_\infty + \tilde{c}_7 \|D\|_\infty\right)^2.
\end{aligned}$$

We then have

$$\begin{aligned} L_Y\left([\phi]_1\right) &= \left(\tilde{c}_6 + \tilde{c}_7[\phi]_1\right) e^{\tilde{c}_3 T[\phi]_1^2}, \\ L_Z\left([\phi]_1\right) &= \left(\tilde{c}_8 + \tilde{c}_9[\phi]_1^2\right) e^{\tilde{c}_3 T[\phi]_1^2} + C_L\left([\phi]_1\right). \end{aligned} \tag{3.23}$$

With these regularity results, we can bound the Lipschitz coefficient of the control iterates, in the same style as (Reisinger et al., 2022, Proposition 3.8).

Theorem 3.4.5. *Let $\phi^0 \in \mathcal{V}_L$. Define*

$$\begin{aligned} \tilde{c}_{10} &= \max\left(T\tilde{c}_3[\phi^0]_1, \frac{T\tilde{c}_3\|S\|_\infty + 3c_0}{c_0}\right), \\ \tilde{c}_{11} &= \|S\|_\infty + \left(\|B\|_\infty \tilde{c}_6 e^{\tilde{c}_{10}^2} + \|D\|_\infty \tilde{c}_7\right) e^{\tilde{c}_{10}^2} + \|D\|_\infty C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right) \end{aligned}$$

Assume the following

$$\begin{aligned} \|B\|_\infty T\tilde{c}_3\tilde{c}_6 e^{\tilde{c}_{10}^2} &\le c_0, \\ \|D\|_\infty T\tilde{c}_3\tilde{c}_8 e^{\tilde{c}_{10}^2} &\le c_0, \\ \|B\|_\infty \tilde{c}_7 e^{\tilde{c}_{10}^2} &\le \frac{1}{2}c_0, \\ \|D\|_\infty \frac{\tilde{c}_9}{T\tilde{c}_3}\tilde{c}_{10} e^{\tilde{c}_{10}^2} &\le \frac{1}{2}c_0, \\ \|D\|_\infty T\tilde{c}_3 C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right) &\le c_0, \end{aligned} \tag{3.24}$$

Then the sequences $\left(\phi^k\right)_{m\in\mathbb{N}_0}$ generated by (3.12) are Lipschitz, and the sequence of Lipschitz coefficients $\left([\phi^k]_1\right)_{m\in\mathbb{N}_0}$ satisfies, for all $m \in \mathbb{N}_0$

$$[\phi^k]_1 \le M\left(\phi^0\right) := [\phi^0]_1 + \frac{\tilde{c}_{11}}{c_0}.$$

Remark 11. The conditions (3.24) would be satisfied for sufficiently small T (assuming that C_L has sub-linear growth), as one can show that the bounding coefficients C_{SDE} and C_{BSDE} tend to 0 as T tends to 0 linearly. It will also be satisfied for sufficiently large c_0, that is, if we take R to be sufficiently large. Note that there is dependence on c_0 on the left of the consequentialities via $\tilde{c}_{10}$, but this term remains bounded as we take $c_0 \to \infty$.

Proof. Fix $\phi^0 \in \mathcal{V}$. By Theorem 3.3.3 with $\phi^1 = \phi^2 = \phi^k$ we have, for any $t \in [0,T]$ and

$x_1, x_2 \in \mathbb{R}^n$,

$$\begin{aligned}\left|\phi_t^{k+1}(x_1) - \phi_t^{k+1}(x_2)\right| \leq{}& (1-2\tau c_0)\left|\phi_t^k(x_1) - \phi_t^k(x_2)\right| \\ &+ \tau\|S\|_\infty |x_1 - x_2| \\ &+ \tau\|B\|_\infty \left|Y_t^{t,x_1,\phi^k} - Y_t^{t,x_2,\phi^k}\right| \\ &+ \tau\|D\|_\infty \left|Z_t^{t,x_1,\phi^k} - Z_t^{t,x_2,\phi^k}\right|.\end{aligned}$$

Since $\phi^k \in \mathcal{V}$, it is $[\phi^k]_1$-Lipschitz, and by Theorems 3.4.1 and 3.4.4 we have

$$\begin{aligned}\left|Y_t^{t,x_1,\phi^k} - Y_t^{t,x_2,\phi^k}\right| &\leq L_Y\left([\phi^k]_1\right)|x_1 - x_2|, \\ \left|Z_t^{t,x_1,\phi^k} - Z_t^{t,x_2,\phi^k}\right| &\leq L_Z\left([\phi^k]_1\right)|x_1 - x_2|.\end{aligned}$$

Hence we may write

$$\begin{aligned}\left|\phi_t^{k+1}(x_1) - \phi_t^{k+1}(x_2)\right| \leq{}& (1-2\tau c_0)\left|\phi_t^k(x_1) - \phi_t^k(x_2)\right| \\ &+ \tau\|S\|_\infty |x_1 - x_2| \\ &+ \tau\|B\|_\infty L_Y\left([\phi^k]_1\right)|x_1 - x_2| \\ &+ \tau\|D\|_\infty L_Z\left([\phi^k]_1\right)|x_1 - x_2|.\end{aligned}$$

Since $x_1, x_2 \in \mathbb{R}^n$ were arbitrary, we have, for all $k \in \mathbb{N}_0$

$$[\phi^{k+1}]_1 \leq (1-2\tau c_0)[\phi^k]_1 + \tau\|S\|_\infty + \tau\|B\|_\infty L_Y\left([\phi^k]_1\right) + \tau\|D\|_\infty L_Z\left([\phi^k]_1\right). \tag{3.25}$$

First, we will prove by induction that $\sup_{k\in\mathbb{N}_0} T\tilde{c}_3[\phi^k]_1 \leq \tilde{c}_{10}$. When $k=0$ we have $T\tilde{c}_3[\phi^0]_1 \leq$

$\tilde{c}_{10}$ by definition. So suppose that $T\tilde{c}_3[\phi^k]_1 \leq \tilde{c}_{10}$. Then by (3.25) and (3.23) we have

$$
\begin{aligned}
&T\tilde{c}_3[\phi^{k+1}]_1 && (3.26)\\
&\leq (1-2\tau c_0)\, T\tilde{c}_3[\phi^k]_1 + \tau T\tilde{c}_3\|S\|_\infty + \tau T\tilde{c}_3\|B\|_\infty L_Y\left([\phi^k]_1\right) + \tau T\tilde{c}_3\|D\|_\infty L_Z\left([\phi^k]_1\right) && (3.27)\\
&= (1-2\tau c_0)\, T\tilde{c}_3[\phi^k]_1 + \tau T\tilde{c}_3\|S\|_\infty + \tau\ T\tilde{c}_3\|B\|_\infty \left(\tilde{c}_6 + \tilde{c}_7[\phi^k]_1\right) e^{\tilde{c}_3 T[\phi^k]_1^2} \\
&\quad + \tau T\tilde{c}_3\|D\|_\infty \left(\left(\tilde{c}_8 + \tilde{c}_9[\phi^k]_1^2\right) e^{\tilde{c}_3 T[\phi^k]_1^2} + C_L\left([\phi^k]_1\right)\right) && (3.28)\\
&\leq (1-2\tau c_0)\, \tilde{c}_{10} + \tau T\tilde{c}_3\|S\|_\infty + \tau\|B\|_\infty \left(T\tilde{c}_3\tilde{c}_6 + \tilde{c}_7\tilde{c}_{10}\right) e^{\tilde{c}_{10}^2} \\
&\quad + \tau\|D\|_\infty \left(\left(T\tilde{c}_3\tilde{c}_8 + \frac{\tilde{c}_9}{T\tilde{c}_3}\tilde{c}_{10}^2\right) e^{\tilde{c}_{10}^2} + T\tilde{c}_3 C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right)\right) && (3.29)\\
&\leq (1-2\tau c_0)\, \tilde{c}_{10} + \tau\left(T\tilde{c}_3\|S\|_\infty + 3c_0\right) + \tau c_0\tilde{c}_{10} && (3.30)\\
&\leq (1-2\tau c_0)\, \tilde{c}_{10} + 2\tau c_0\tilde{c}_{10} && (3.31)\\
&= \tilde{c}_{10}, && (3.32)
\end{aligned}
$$

as required. Substituting this bound into (3.28) yields

$$
\begin{aligned}
&[\phi^{k+1}]_1 \\
&\leq (1-2\tau c_0)\,[\phi^k]_1 + \tau\left(\|S\|_\infty + \left(\|B\|_\infty\tilde{c}_6 e^{\tilde{c}_{10}^2} + \|D\|_\infty\tilde{c}_7\right) e^{\tilde{c}_{10}^2} + \|D\|_\infty C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right)\right) \\
&\quad + \tau[\phi^k]_1\left(\tilde{c}_7 e^{\tilde{c}_{10}^2} + \|D\|_\infty\tilde{c}_9\frac{\tilde{c}_{10}}{T\tilde{c}_3}e^{\tilde{c}_{10}^2}\right) \\
&\leq (1-\tau c_0)\,[\phi^k]_1 + \tau\left(\|S\|_\infty + \left(\|B\|_\infty\tilde{c}_6 e^{\tilde{c}_{10}^2} + \|D\|_\infty\tilde{c}_7\right) e^{\tilde{c}_{10}^2} + \|D\|_\infty C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right)\right).
\end{aligned}
$$

This is a linear difference inequality, meaning that the sequence $\left([\phi^k]_1\right)_{k\in\mathbb{N}_0}$ is bounded by the sequence $(a_k)_{k\in\mathbb{N}_0}$ defined by

$$
a_{k+1} = c a_k + (1-c)\, M\left(\phi^0\right), \qquad k \in \mathbb{N}_0,
$$

with $c := 1 - \tau c_0$ satisfying $|c| \leq 1$ by Assumption 1. This equation has solution

$$
a_k = c^k a_0 + \left(1 - c^k\right) M\left(\phi^0\right), \qquad k \in \mathbb{N}_0.
$$

Therefore we have, for $k \in \mathbb{N}_0$,

$$[\phi^{k+1}]_1 \leq (1-\tau c_0)^{k+1} [\phi^0]_1 + M\left(\phi^0\right)\left(1-(1-\tau c_0)^{k+1}\right) \leq [\phi^0]_1 + M\left(\phi^0\right),$$

as required. $\square$

3.4.2 Convergence in $|.|_0$ norm

We wish to study the sequence $\left(|\phi^{k+1}-\phi^k|_0\right)_{k\in\mathbb{N}_0}$. To do this, we need to study the linear growth rate (in terms of x) of the update in (3.12), for two different control functions. To this end we must study the linear growth of differences of $Y_t^{t,x,\phi}$, $Z_t^{t,x,\phi}$ when we fix $x \in \mathbb{R}^n$, $t \in [0,T]$ and change $\phi \in \mathcal{V}$. The following result is adapted from (Reisinger et al., 2022, Proposition 3.9). The core concept of the proof is the same, but the function C_Y is altered to account for the control in the diffusion of X.

Theorem 3.4.6. *For $\phi \in \mathcal{V}$, $t \in [0,T]$, $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{H}^2\left(t,T;\mathbb{R}^n\right)$ be solutions to (3.10). Let ϕ^1,ϕ^2, then there exists*

$$C_Y\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) \geq 0,$$

such that for all $t \in [0,T]$, $x \in \mathbb{R}^n$

$$|Y_t^{t,x,\phi^1} - Y_t^{t,x,\phi^2}| \leq C_Y\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)(1+|x|)\,|\phi^1-\phi^2|_0$$

Proof. Fix $t \in [0,T]$, $x \in \mathbb{R}^n$ and $\phi^1,\phi^2 \in \mathcal{V}$. Define the differences processes, for $s \in [t,T]$,

$$\begin{aligned}
\Delta X_s &= X_s^{t,x,\phi^1} - X_s^{t,x,\phi^2},\\
\Delta Y_s &= Y_s^{t,x,\phi^1} - Y_s^{t,x,\phi^2},\\
\Delta Z_s &= Z_s^{t,x,\phi^1} - Z_s^{t,x,\phi^2},\\
\Delta \phi_s &= \phi_s^1\left(X_s^{t,x,\phi^1}\right) - \phi_s^2\left(X_s^{t,x,\phi^2}\right).
\end{aligned}$$

Using the definition (3.10) of the individual FBSDEs, the difference processes $(\Delta X, \Delta Y, \Delta Z)$

solve the FBSDE, for $s \in [t,T]$,

$$
\begin{aligned}
\Delta X_s &= \int_t^s (A_r \Delta X_r + B_r \Delta\phi_r)\, dr + \int_t^s (C_r \Delta X_r + D_r \Delta\phi_r)\, dW_r, \\
\Delta Y_s &= G\Delta X_T + \int_t^s \left(A_r^\top \Delta Y_r + C_r^\top \Delta Z_r + S_r^\top \Delta\phi_r + Q_r \Delta X_r\right) dr - \int_t^s \Delta Z_r dW_r.
\end{aligned}
$$

Let us apply Theorem A.1.10 to $(\Delta Y, \Delta Z)$. We then have

$$
\mathbb{E}\left[|\Delta Y_t|^2\right] \leq \tilde{c}_1 \mathbb{E}\left[|G\Delta X_T|^2 + \left(\int_t^T |S_r^\top \Delta\phi_r + Q_r \Delta X_r| dr\right)^2\right]. \tag{3.33}
$$

Note that, for $r \in [t,T]$, we have

$$
\begin{aligned}
|\Delta\phi_r| &= \left|\phi_r^1\left(X_r^{t,x,\phi^1}\right) - \phi_r^2\left(X_r^{t,x,\phi^2}\right)\right| \\
&\leq \left|\phi_r^1\left(X_r^{t,x,\phi^1}\right) - \phi_r^2\left(X_r^{t,x,\phi^1}\right)\right| + \left|\phi_r^2\left(X_r^{t,x,\phi^1}\right) - \phi_r^2\left(X_r^{t,x,\phi^2}\right)\right| \\
&\leq |\phi^1 - \phi^2|_0 \left|X_r^{t,x,\phi^1}\right| + [\phi^2]_1 \left|\Delta X_r\right|.
\end{aligned}
$$

Using this, and Holder's inequality, we may bound (3.20) by the following.

$$
\begin{aligned}
&\mathbb{E}\left[|\Delta Y_t|^2\right] \\
&\leq \tilde{c}_1 |G|^2 \mathbb{E}\left[|\Delta X_T^2|\right] \\
&+ \tilde{c}_1 \mathbb{E}\left[\left(\int_t^T \left(\left(\|S\|_\infty [\phi^2]_1 + \|Q\|_\infty\right)|\Delta X_r| + \|S\|_\infty |\phi^1 - \phi^2|_0 |X_r^{t,x,\phi^1}|\right) dr\right)^2\right] &\quad (3.34) \\
&\leq \tilde{c}_1 \mathbb{E}\left[|G|^2 |\Delta X_T^2|\right] \\
&+ 3\tilde{c}_1 T \mathbb{E}\left[\int_t^T \left(\left(\|S\|_\infty^2 [\phi^2]_1^2 + \|Q\|_\infty^2\right)|\Delta X_r|^2 + \|S\|_\infty^2 |\phi^1 - \phi^2|_0^2 |X_r^{t,x,\phi^1}|^2\right) dr\right] &\quad (3.35) \\
&= \tilde{c}_1 |G|^2 \mathbb{E}\left[|\Delta X_T^2|\right] + 3\tilde{c}_1 T \left(\|S\|_\infty^2 [\phi^2]_1^2 + \|Q\|_\infty^2\right) \int_t^T \mathbb{E}\left[|\Delta X_r|^2\right] dr \\
&+ 3\tilde{c}_1 T \|S\|_\infty^2 |\phi^1 - \phi^2|_0^2 \int_t^T \mathbb{E}\left[|X_r^{t,x,\phi^1}|^2\right] dr &\quad (3.36) \\
&\leq 3\tilde{c}_1 T \left(\tilde{c}_4 + \tilde{c}_5^2 T [\phi^2]_1^2\right) \sup_{r\in[t,T]} \mathbb{E}\left[|\Delta X_r|^2\right] \\
&+ 3\tilde{c}_1 \tilde{c}_5^2 T^2 |\phi^1 - \phi^2|_0^2 \sup_{r\in[t,T]} \mathbb{E}\left[|X_r^{t,x,\phi^1}|^2\right]. &\quad (3.37)
\end{aligned}
$$

We proceed to bound these lines individually. First, applying Theorem A.1.9 to X^{t,x,ϕ^1} gives a constant

$$C_{Y,1} = C_{\mathrm{SDE}}\left(\|A\|_\infty + \|B\|_\infty[\phi^1]_0, \|C\|_\infty + \|D\|_\infty[\phi^1]_0\right),$$

such that

$$\begin{aligned}
&\mathbb{E}\left[\sup_{r\in[t,T]}\left|X_r^{t,x,\phi^1}\right|^2\right] \\
&\leq C_{Y,1}\mathbb{E}\left[|x|^2 + \left(\int_t^T \|B\|_\infty|\phi_r^1(0)|dr\right)^2 + \int_t^T \|D\|_\infty^2|\phi_r^1(0)|^2dr\right] && (3.38)\\
&\leq C_{Y,1}\left(|x|^2 + 2\tilde{c}_3\int_t^T|\phi_r^1(0)|^2dr\right) && (3.39)\\
&= C_{Y,1}\left(|x|^2 + 2\tilde{c}_3T|\phi^1|_0^2\right) && (3.40)\\
&\leq C_{Y,2}\left(1+|x|^2\right), && (3.41)
\end{aligned}$$

where

$$C_{Y,2} = \max\left(1, 2\tilde{c}_3T|\phi^1|_0^2\right)C_{Y,1}.$$

Now for fixed $s \in [t,T]$, applying Theorem A.1.9 to $(\Delta X_r)_{r\in[t,s]}$, and using Holder's inequality and (3.41), yields

$$\begin{aligned}
\mathbb{E}\left[|\Delta X_s|^2\right] &\leq \tilde{c}_2\mathbb{E}\left[\left(\int_t^s\|B\|_\infty|\Delta\phi_r|dr\right)^2 + \int_s^t\|D\|_\infty^2|\Delta\phi_r|^2dr\right]\\
&\leq \tilde{c}_2\tilde{c}_4\mathbb{E}\left[\int_s^t|\Delta\phi_r|^2dr\right]\\
&\leq \tilde{c}_2\tilde{c}_4\mathbb{E}\left[\int_s^t\left(|\phi^1-\phi^2|_0|X_r^{t,x,\phi^1}| + [\phi^2]_1|\Delta X_r|\right)^2dr\right]\\
&\leq \tilde{c}_2\tilde{c}_4\mathbb{E}\left[2C_{Y,2}|\phi^1-\phi^2|_0^2\left(1+|x|^2\right) + \int_t^s 2[\phi^2]_1^2|\Delta X_r|^2dr\right].
\end{aligned}$$

By Gronwall's inequality, we therefore have

$$\mathbb{E}\left[|\Delta X_s|^2\right] \leq 4C_{Y,2}\tilde{c}_2\tilde{c}_3e^{4\tilde{c}_2\tilde{c}_3[\phi^2]_1^2T}|\phi^1-\phi^2|_0^2\left(1+|x|^2\right).$$

Since s was arbitrary, we have

$$\sup_{s\in[t,T]}\mathbb{E}\left[|\Delta X_s|^2\right] \leq C_{Y,3}|\phi^1-\phi^2|_0^2\left(1+|x|^2\right), \tag{3.42}$$

where

$$C_{Y,3} = 4\tilde{c}_2\tilde{c}_3 e^{4\tilde{c}_2\tilde{c}_3[\phi^2]_1^2 T} C_{Y,2}.$$

Substituting (3.41) and (3.42) into (3.37) yields

$$\begin{aligned}\mathbb{E}\left[|\Delta Y_t|^2\right] \leq\; & 3\tilde{c}_1 T\left(\tilde{c}_4 + \tilde{c}_5^2 T[\phi^2]_1^2\right) C_{Y,3}|\phi^1 - \phi^2|_0^2\left(1 + |x|^2\right) \\ & + 3\tilde{c}_1\tilde{c}_5^2 T^2|\phi^1 - \phi^2|_0^2 C_{Y,2}\left(1 + |x|^2\right).\end{aligned}$$

Therefore

$$|\Delta Y_t| \leq C_Y\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)|\phi^1 - \phi^2|_0\left(1 + |x|\right),$$

where, using that $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \geq 0$, the term $C_Y\left([\phi^1]_1, [\phi^2]_1\right) \geq 0$ is given by

$$\begin{aligned} C_Y\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) &= \sqrt{3}T^{\frac{1}{2}}\tilde{c}_1^{\frac{1}{2}}\left(\tilde{c}_4^{\frac{1}{2}} + T^{\frac{1}{2}}\tilde{c}_5[\phi^2]_1\right)C_{Y,3}^{\frac{1}{2}} + \sqrt{3}\tilde{c}_1^{\frac{1}{2}}\tilde{c}_5 T C_{Y,2}^{\frac{1}{2}}, \\ \tilde{c}_1 &= C_{\mathrm{BSDE}}\left(\|A\|_\infty, \|C\|_\infty\right), \\ \tilde{c}_2 &= C_{\mathrm{SDE}}\left(\|A\|_\infty, \|C\|_\infty\right), \\ \tilde{c}_3 &= \frac{1}{2}\left(\|B\|_\infty^2 T + \|D\|_\infty^2\right), \\ \tilde{c}_4 &= |G| + \|Q\|_\infty T, \\ \tilde{c}_5 &= \|S\|_\infty, \\ C_{Y,1} &= C_{\mathrm{SDE}}\left(\|A\|_\infty + \|B\|_\infty[\phi^1]_0, \|C\|_\infty + \|D\|_\infty[\phi^1]_0\right), \\ C_{Y,2} &= \max\left(1, 2\tilde{c}_3 T|\phi^1|_0^2\right)C_{Y,1}, \\ C_{Y,3} &= 4\tilde{c}_2\tilde{c}_3 e^{4\tilde{c}_2\tilde{c}_3[\phi^2]_1^2 T} C_{Y,2}, \end{aligned}$$

as required. □

Assumption 3. We assume the following.

1. There exist $\mathcal{V}_L \subset \mathcal{V}$ and a non-negative increasing (in all inputs) function $C_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) \geq 0$ such that for any $\phi^1, \phi^2 \in \mathcal{V}_L$, $0 \leq t \leq T$

$$\left|\delta v_t^{\phi^1}(x) - \delta v_t^{\phi^2}(x)\right| \leq C_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)|\phi^1 - \phi^2|_0. \tag{3.43}$$

2. There exists $\phi \in \mathcal{V}_L$ such that the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ generated by (3.12) with $\phi^0 = \phi$ is contained in $\mathcal{V}_L$.

Remark 12. Again, in the unconstrained case this assumption is satisfied with $\mathcal{V}_L$ the space of linear functions. Let $\phi_t^i(x) = \alpha_t^i x + \beta_t^i$, for some $\alpha^i \in \mathcal{C}^0([0,T];\mathbb{R}^{m\times n})$ and $\beta^i \in \mathcal{C}^0([0,T];\mathbb{R}^m)$, $i = 1,2$. Then v^{ϕ^i} is of the form $v_s^{\phi^i}(x) = a_s^i x + b_s^i$, $s \in [t,T]$, for some $a^i \in C^0([t,T];\mathbb{R}^{n\times n})$ and $b^i \in C^0([t,T];\mathbb{R}^n)$, for $i = 1,2$. In this case we have

$$\left|\delta v_t^{\phi^1}(x) - \delta v_t^{\phi^2}(x)\right| = |a_t^1 - a_t^2|.$$

Now, by (3.83) and Theorem 3.5.2 we have

$$\begin{aligned}
\|a^1 - a^2\|_\infty &\le e^{\left(c_1 + c_2\|\alpha^1\|_\infty\right)T}\left(c_4 + c_2 T C_a\left(\|\alpha^2\|_\infty\right)\right)\|\alpha^1 - \alpha^2\|_\infty \\
&\le e^{\left(c_1 + c_2\|\alpha^1\|_\infty\right)T}\left(c_4 + c_2 T C_a\left(\|\alpha^2\|_\infty\right)\right)\max\left(\|\alpha^1 - \alpha^2\|_\infty, \|\beta^1 - \beta^2\|_\infty\right) \\
&= e^{\left(c_1 + c_2[\phi^1]_1\right)T}\left(c_4 + c_2 T C_a\left([\phi^2]_1\right)\right)|\phi^1 - \phi^2|_0.
\end{aligned}$$

Therefore the assumption is satisfied with

$$C_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) = e^{\left(c_1 + c_2 C_a\left([\phi^2]_1\right)\right)T}\left(c_4 + c_2 T[\phi^2]_1\right).$$

In the constrained case, we have

$$\left|\delta v_t^{\phi^1}(x) - \delta v_t^{\phi^2}(x)\right| = \left|\delta\left(v_t^{\phi^1} - v_t^{\phi^2}\right)(x)\right| \le [v^{\phi^1} - v^{\phi^2}]_1.$$

Note that by Theorem 3.4.6 we have

$$|v^{\phi^1} - v^{\phi^2}|_0 \le C_Y\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)|\phi^1 - \phi^2|_0.$$

Therefore the assumption would be satisfied if for any pair $\phi^1, \phi^2 \in \mathcal{V}_L$ we had

$$[v^{\phi^1} - v^{\phi^2}]_1 \le \tilde{C}_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)|v^{\phi^1} - v^{\phi^2}|_0,$$

for some non negative increasing function $\tilde{C}_v$. Again, it is not clear in this case if Assumption 3 is satisfied.

Theorem 3.4.7. *For $\phi \in \mathcal{V}$, $t \in [0,T]$, $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{S}^2(t,T;\mathbb{R}^n) \times \mathcal{H}^2(t,T;\mathbb{R}^n)$ be solutions to (3.10). Let $\phi^1, \phi^2 \in \mathcal{V}_L$, $t \in [0,T]$, and $x \in \mathbb{R}^n$.*

Then there exists a constant

$$C_Z\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) \geq 0,$$

such that

$$\left|Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2}\right| \leq C_Z\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) |\phi^1 - \phi^2|_0 \left(1 + |x|\right).$$

Proof. For any $t \in [0,T]$, $x \in \mathbb{R}^n$, $\phi \in \mathcal{V}_L$ recall the representation (3.22)

$$Z_t^{t,x,\phi} = \delta v_t^{\phi}\left(x\right) \mathcal{D}_t X_t^{t,x,\phi} = \delta v_t^{\phi}\left(x\right) \left(C_t x + D_t \phi_t\left(x\right)\right),$$

where v_t^{ϕ} is a measurable, Lipschitz function, with Lipschitz coefficient $L_Y\left([\phi]_1\right)$. Now, fix $t \in [0,T]$, $\phi^1, \phi^2 \in \mathcal{V}_L$, $x \in \mathbb{R}^n$. We have

$$\begin{aligned}
\left|Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2}\right| &= \left|\delta v_t^{\phi^1}\left(x\right)\left(C_t x + D_t \phi_t^1\left(x\right)\right) - \delta v_t^{\phi^2}\left(x\right)\left(C_t x + D_t \phi_t^2\left(x\right)\right)\right| \\
&\leq \left|\delta v_t^{\phi^1}\left(x\right) - \delta v_t^{\phi^2}\left(x\right)\right| \left|C_t x + D_t \phi_t^1\left(x\right)\right| \\
&+ \left|\delta v_t^{\phi^2}\left(x\right)\right| \left|D_t\left(\phi_t^1\left(x\right) - \phi_t^2\left(x\right)\right)\right| \\
&\leq \left|\delta v_t^{\phi^1}\left(x\right) - \delta v_t^{\phi^2}\left(x\right)\right| \left(\|C\|_\infty + \|D\|_\infty |\phi^1|_0\right)\left(1 + |x|\right) \\
&+ L_Y\left([\phi^2]_1\right) \|D\|_\infty |\phi^1 - \phi^2|_0 \left(1 + |x|\right).
\end{aligned}$$

Under Assumption 3, we have

$$\begin{aligned}
\left|Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2}\right| &\leq C_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) |\phi^1 - \phi^2|_0 \left(\|C\|_\infty + \|D\|_\infty |\phi^1|_0\right)\left(1 + |x|\right) \\
&+ L_Y\left([\phi^2]_1\right) \|D\|_\infty |\phi^1 - \phi^2|_0 \left(1 + |x|\right).
\end{aligned}$$

The result follows with

$$C_Z\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right) := C_v\left(|\phi^1|_0, [\phi^1]_1, [\phi^2]_1\right)\left(\|C\|_\infty + \|D\|_\infty |\phi^1|_0\right) + L_Y\left([\phi^2]_1\right) \|D\|_\infty.$$

□

With these sensitivity estimates of $Y_t^{t,x,\phi}$ and $Z_t^{t,x,\phi}$, we can use Theorem 3.3.3 to derive a difference inequality for the sequence $\left(|\phi^{k+1} - \phi^k|_0\right)_{k \in \mathbb{N}_0}$.

Assumption 4. We assume the following.

1. There exists non empty $\mathcal{V}_L \subset \mathcal{V}$ and $C_K \geq 0$ such that for all $\phi \in \mathcal{V}_L$,

$$|\phi|_0 \leq C_K.$$

2. There exists $\phi \in \mathcal{V}_L$ such that the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ generated by (3.12) with $\phi^0 = \phi$ is contained in $\mathcal{V}_L$.

Remark 13. It is shown in Theorem 3.5.6 that in the unconstrained case, the sequence $\left(|\phi^k|_0\right)_{k\in\mathbb{N}_0}$ is uniformly bounded when $\mathcal{V}_L$ is the space of linear functions. In the constrained case, this would be satisfied for any $\phi \in \mathcal{V}$ if the constraint space was bounded. In a more general setting, we may establish this bound if our problem has running and terminal cost functions with bounded derivatives.

Theorem 3.4.8. *Let $\mathcal{V}_L$ satisfy Assumptions 2, 3 and 4. Let τ satisfy Assumption 1. Let $\phi^0 \in \mathcal{V}_L$ satisfy (3.24) and additionally satisfy*

$$\|B\|_\infty C_Y\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right) + \|D\|_\infty C_Z\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right) \leq 2c_0, \tag{3.44}$$

where

$$\begin{aligned}
M\left(\phi^0\right) &= [\phi^0]_1 + \frac{\tilde{c}_{11}}{c_0},\\
\tilde{c}_{11} &= \|S\|_\infty + \left(\|B\|_\infty \tilde{c}_6 e^{\tilde{c}_{10}^2} + \|D\|_\infty \tilde{c}_7\right) e^{\tilde{c}_{10}^2} + \|D\|_\infty C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right),\\
c_0 &= \frac{1}{2}\frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty}.
\end{aligned}$$

Then there exists $\hat{c} \in (0,1)$ such that the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ defined by (3.12) satisfies, for all $k \in \mathbb{N}_0$

$$|\phi^{k+1} - \phi^k|_0 \leq \hat{c}|\phi^k - \phi^{k-1}|_0.$$

Proof. Proof is adapted from (Reisinger et al., 2022, Theorem 3.11). Fix $x \in \mathbb{R}^n$, $t \in [0,T]$ and

$k \in \mathbb{N}_0,$. By Theorem 3.3.3 with $\phi^1 = \phi^k$, $\phi^2 = \phi^{k-1}$, and $x_1 = x_2 = x$, we have

$$\begin{aligned} |\phi_t^{k+1}(x) - \phi_t^k(x)| \leq{}& (1 - 2\tau c_0) |\phi_t^k(x) - \phi_t^{k-1}(x)| \\ &+ \tau \|B\|_\infty \left| Y_t^{t,x,\phi^k} - Y_t^{t,x,\phi^{k-1}} \right| \\ &+ \tau \|D\|_\infty \left| Z_t^{t,x,\phi^k} - Z_t^{t,x,\phi^{k-1}} \right|. \end{aligned} \tag{3.45}$$

By definition, $\phi^k, \phi^{k-1} \in \mathcal{V}_L$. Hence, using Theorems 3.4.6 and 3.4.7 we have

$$\begin{aligned} \left|\phi_t^k(x) - \phi_t^{k-1}(x)\right| &\leq \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|), \\ \left| Y_t^{t,x,\phi^k} - Y_t^{t,x,\phi^{k-1}} \right| &\leq C_Y\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|), \\ \left| Z_t^{t,x,\phi^k} - Z_t^{t,x,\phi^{k-1}} \right| &\leq C_Z\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|). \end{aligned}$$

Substituting these into (3.45) gives

$$\begin{aligned} |\phi_t^{k+1}(x) - \phi_t^k(x)| \leq{}& (1 - 2\tau c_0) \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|), \\ &+ \tau \|B\|_\infty C_Y\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|), \\ &+ \tau \|D\|_\infty C_Z\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0 (1 + |x|). \end{aligned}$$

Since $x \in \mathbb{R}^n$ and $t \in [0, T]$ were arbitrary, we have, for $k \in \mathbb{N}$,

$$\begin{aligned} |\phi^{k+1} - \phi^k|_0 \leq{}& (1 - 2\tau c_0) \left|\phi^k - \phi^{k-1}\right|_0 \\ &+ \tau \|B\|_\infty C_Y\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0 \\ &+ \tau \|D\|_\infty C_Z\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) \left|\phi^k - \phi^{k-1}\right|_0. \end{aligned}$$

By Theorem 3.4.5, we have $\sup_{k \in \mathbb{N}_0} [\phi^k]_1 \leq M(\phi^0)$. Using this, Assumption 4, and that the functions C_Y and C_Z are increasing in all their inputs, we have

$$\begin{aligned} C_Y\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) &\leq C_Y\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right), \\ C_Z\left(|\phi^k|_0, [\phi^k]_1, [\phi^{k-1}]_1\right) &\leq C_Z\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right). \end{aligned}$$

We may therefore write, for $k \in \mathbb{N}$,

$$|\phi^{k+1} - \phi^k|_0 \leq \hat{c}|\phi^k - \phi^{k-1}|_0,$$

where

$$\hat{c} = (1 - 2\tau c_0) + \tau \left(\|B\|_\infty C_Y \left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right) + \tau\|D\|_\infty C_Z \left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right)\right).$$

We require $\hat{c} < 1$, which is equivalent to (3.44), as required. □

3.4.3 Proof of Main Result

Theorem 3.4.9. *Let $\mathcal{V}_L$ satisfy Assumptions 2, 3, and 4. Let $\phi^0 \in \mathcal{V}_L$ satisfy (3.24) and (3.44). Let $\tau > 0$ satisfy Assumption 1. Then there exists $\phi^* \in \mathcal{V}$ with associated control $u^{\phi^*} \in \mathcal{H}^2(U)$, and constants $\hat{c} \in [0,1)$ and $\hat{C} \geq 0$ such that*

1. *For all $k \in \mathbb{N}_0$, $|\phi^k - \phi^*|_0 \leq \hat{c}^k|\phi^k - \phi^*|_0$.*
2. *For all $k \in \mathbb{N}_0$, $\|u^{\phi^k} - u^{\phi^*}\|_{\mathcal{H}^2} \leq \hat{C}\hat{c}^k|\phi^* - \phi^0|_0$.*
3. *u^{ϕ^*} is a stationary point of J defined in (3.4).*

Proof. Proof is adapted from (Reisinger et al., 2022, Theorem 3.13). Let $\phi^0 \in \mathcal{V}$ satisfy the assumptions. By Theorem 3.4.8, the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ is a Cauchy sequence in the Banach space $(\mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m), |\cdot|_0)$. Therefore by Banach's fixed point Theorem, there exists $\phi^* \in \mathcal{B}([0,T] \times \mathbb{R}^n; \mathbb{R}^m)$ such that $\lim_{k\to\infty} |\phi^k - \phi^*|_0 = 0$. Since, by Theorem 3.4.5, we have $[\phi^k]_1 \leq M(\phi^0)$ for all k, this limiting function satisfies $[\phi^*]_1 \leq M(\phi^0)$. To show $\phi^* \in \mathcal{V}$, it remains to show that ϕ^* takes values in U.

By Theorems 3.4.6 and 3.4.7 there exists $C \geq 0$ such that for all $t \in [0,T]$, $x \in \mathbb{R}^n$ and $k_1, k_2 \in \mathbb{N}_0$,

$$\max\left(\left|Y_t^{t,x,\phi^{k_1}} - Y_t^{t,x,\phi^{k_2}}\right|, \left|Z_t^{t,x,\phi^{k_1}} - Z_t^{t,x,\phi^{k_2}}\right|\right) \leq C\left(1 + |x|\right)|\phi^{k_1} - \phi^{k_2}|_0.$$

Therefore the sequences $\left(Y_t^{t,x,\phi^k}\right)$ and $\left(Z_t^{t,x,\phi^k}\right)$ are Cauchy in $(\mathbb{R}^n, |\cdot|)$. Hence there exists $\mathcal{Y}\colon [0,T]\times\mathbb{R}^n \to \mathbb{R}^n$ and $\mathcal{Z}\colon [0,T]\times\mathbb{R}^n \to \mathbb{R}^n$ such that, for $t\in[0,T]$ and $x\in\mathbb{R}^n$,

$$(\mathcal{Y}_t(x), \mathcal{Z}_t(x)) = \lim_{k\to\infty}\left(Y_t^{t,x,\phi^k}, Z_t^{t,x,\phi^k}\right).$$

By continuity of the function prox_U we may write, for $t\in[0,T]$, $x\in\mathbb{R}^n$

$$\begin{aligned}
&\phi_t^*(x)\\
&= \lim_{k\to\infty}\mathcal{L}\left(\phi^k\right)_t(x)\\
&= \lim_{k\to\infty}\mathrm{prox}_U\left(\phi_t^k(x) - \tau\left(B_t^\top Y_t^{t,x,\phi^k} + D_t^\top Z_t^{t,x,\phi^k} + R_t\phi_t^k(x) + S_t x\right)\right)\\
&= \mathrm{prox}_U\left(\lim_{k\to\infty}\phi_t^k(x) - \tau\left(\lim_{k\to\infty}B_t^\top Y_t^{t,x,\phi^k} + \lim_{k\to\infty}D_t^\top Z_t^{t,x,\phi^k} + \lim_{k\to\infty}R_t\phi_t^k(x) + S_t x\right)\right)\\
&= \mathrm{prox}_U\left(\phi_t^*(x) - \tau\left(B_t^\top\mathcal{Y}_t(x) + D_t^\top\mathcal{Z}_t(x) + R_t\phi_t^*(x) + S_t x\right)\right).
\end{aligned}$$

Hence $\phi_t^*(x)\in U$ for all $t\in[0,T]$ and $x\in\mathbb{R}^n$, so $\phi^*\in\mathcal{V}$. By Theorems 3.4.6 and 3.4.7 we have $(\mathcal{Y}_t(x), \mathcal{Z}_t(x)) = \left(Y_t^{t,x,\phi^*}, Z_t^{t,x,\phi^*}\right)$ for all $t\in[0,T]$ and $x\in\mathbb{R}^n$. Hence we may write, for $(t,x)\in[0,T]\times\mathbb{R}^n$,

$$\phi_t^*(x) = \mathrm{prox}_U\left(\phi_t^*(x) - \tau\left(B_t^\top Y_t^{t,x,\phi^*} + D_t^\top Z_t^{t,x,\phi^*} + R_t\phi_t^*(x) + S_t x\right)\right). \tag{3.46}$$

We now proceed to prove the items in the Theorem. For item 1, following the same ideas as Theorem 3.4.8 yields

$$|\phi^{k+1} - \phi^*|_0 \le \hat{c}|\phi^k - \phi^*|_0,$$

as required. For item 2, note that

$$\begin{aligned}
\|u^{\phi^{k+1}} - u^{\phi^*}\|_{\mathcal{H}^2} &= \|\phi^{k+1}\left(X^{0,x_0,\phi^{k+1}}\right) - \phi^*\left(X^{0,x_0,\phi^*}\right)\|_{\mathcal{H}^2}\\
&\le \|\phi^{k+1}\left(X^{0,x_0,\phi^{k+1}}\right) - \phi^{k+1}\left(X^{0,x_0,\phi^*}\right)\|_{\mathcal{H}^2} + \|\phi^{k+1}\left(X^{0,x_0,\phi^*}\right) - \phi^*\left(X^{0,x_0,\phi^*}\right)\|_{\mathcal{H}^2}\\
&\le [\phi^{k+1}]_1\|X^{0,x_0,\phi^{k+1}} - X^{0,x_0,\phi^*}\|_{\mathcal{H}^2} + |\phi^{k+1} - \phi^*|_0\left(1 + \|X^{0,x_0,\phi^*}\|_{\mathcal{H}}^2\right).
\end{aligned}$$

Using Theorem A.1.9, one can show that there exists $C\ge 0$ such that $||X^{0,x_0,\phi^*}||_{\mathcal{H}}^2 \le C$ and $||X^{0,x_0,\phi^{k+1}} - X^{0,x_0,\phi^*}||_{\mathcal{H}^2} \le C|\phi^{k+1} - \phi^*|_0$, yielding the result.

We now prove item 3. Consider the FBSDE, for $s \in [0,T]$,

$$\begin{aligned}
dX_s^{\phi^*} &= \left(A_s X_s^{\phi^*} + B_s \phi_s^*\left(X_s^{\phi^*}\right)\right) ds + \left(C_s X_s^{\phi^*} + D_s \phi_s^*\left(X_s^{\phi^*}\right)\right) dW_s, \\
X_0^{\phi^*} &= x_0, \\
dY_s^{\phi^*} &= -\left(A_s^\top Y_s^{\phi^*} + C_s^\top Z_s^{\phi^*} + S_s^\top \phi_s^*\left(X_s^{\phi^*}\right) + Q_s X_s^{\phi^*}\right) ds + Z_s^{\phi^*} dW_s, \\
Y_T^{\phi^*} &= G X_T^{\phi^*}.
\end{aligned} \tag{3.47}$$

Fix $t \in [0,T]$. By the Markov property, we have $\left(Y_t^{\phi^*}, Z_t^{\phi^*}\right) = \left(Y_t^{t,X_t^{\phi^*},\phi^*}, Z_t^{t,X_t^{\phi^*},\phi^*}\right)$. Hence we can use (3.46) to write

$$\begin{aligned}
\phi_t^*\left(X_t^{\phi^*}\right) &= \operatorname{prox}_U\left(\phi_t^*\left(X_t^{\phi^*}\right) - \tau\left(B_t^\top Y_t^{t,X_t^{\phi^*},\phi^*} + D_t^\top Z_t^{t,X_t^{\phi^*},\phi^*} + R_t \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}\right)\right) \\
&= \operatorname{prox}_U\left(\phi_t^*\left(X_t^{\phi^*}\right) - \tau\left(B_t^\top Y_t^{\phi^*} + D_t^\top Z_t^{\phi^*} + R_t \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}\right)\right).
\end{aligned}$$

We wish to use Theorem A.1.6. We therefore set $J(u) = F(u) + \tilde{F}(u)$ where, for $u \in \mathcal{H}^2(\mathbb{R}^m)$,

$$\begin{aligned}
F(u) &= \mathbb{E}\left[\int_0^T \frac{1}{2}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix}^\top \begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix} ds + \frac{1}{2}\left(X_T^u\right)^\top G X_T^u\right], \\
\tilde{F}(u) &= \mathbb{E}\left[\int_0^T \Phi_U(u_t)\, dt\right],
\end{aligned}$$

where X^u satisfies (3.7). Then $\tilde{F}$ is proper, lower semi continuous and convex with $\operatorname{prox}_{\tau\tilde{F}} = \operatorname{prox}_U$ and F is differentiable with

$$\nabla F\left(u^{\phi^*}\right)_t = B_t^\top Y_t^{\phi^*} + D_t^\top Z_t^{\phi^*} + R_t \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}.$$

The result then follows from Theorem A.1.6 and (3.46). □

3.5 Convergence of PGM for Unconstrained Case

Let us now consider the case $U = \mathbb{R}^m$. The update formula (3.11) then becomes

$$\mathcal{L}(\phi)_t(x) = \phi_t(x) - \tau\left(B_t^\top Y_t^{t,x,\phi} + D_t^\top Z_t^{t,x,\phi} + R_t \phi_t(x) + S_t x\right), \tag{3.48}$$

for $\phi \in \mathcal{V}$, $t \in [0,T]$ and $x \in \mathbb{R}^n$, where $\left(X^{t,x,\phi},\, Y^{t,x,\phi},\, Z^{t,x,\phi}\right) \in \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{H}^2\left(t,T;\mathbb{R}^n\right)$ are solutions to, for $s \in [t,T]$,

$$\begin{aligned}
dX_s^{t,x,\phi} &= \left(A_s X_s^{t,x,\phi} + B_s \phi_s\left(X_s^{t,x,\phi}\right)\right) ds + \left(C_s X_s^{t,x,\phi} + D_s \phi_s\left(X_s^{t,x,\phi}\right)\right) dW_s, \\
X_t^{t,x,\phi} &= x, \\
dY_s^{t,x,\phi} &= -\left(A_s^\top Y_s^{t,x,\phi} + C_s^\top Z_s^{t,x,\phi} + S_s^\top \phi_s\left(X_s^{t,x,\phi}\right) + Q_s X_s^{t,x,\phi}\right) ds + \left(Z_s^{t,x,\phi}\right)^\top dW_s, \\
Y_T^{t,x,\phi} &= G X_T^{t,x,\phi}.
\end{aligned} \tag{3.49}$$

Fix $\phi^0 \in \mathcal{V}$, then for $k \in \mathbb{N}_0$ define our PGM scheme $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ by

$$\phi_t^{k+1}(x) = \mathcal{L}\left(\phi^k\right)_t(x), \tag{3.50}$$

for $t \in [0,T]$ and $x \in \mathbb{R}^n$.

Theorem 3.5.1. *Define a function $\phi \in \mathcal{V}$ by*

$$\phi_t(x) := \alpha_t x + \beta_t, \qquad 0 \le t \le T,\, x \in \mathbb{R}^n,$$

for $\alpha \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m\times n}\right)$ and $\beta \in \mathcal{C}^0\left([0,T];\mathbb{R}^m\right)$. Then there exists $\tilde{\alpha} \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m\times n}\right)$ and $\tilde{\beta} \in \mathcal{C}^0\left([0,T];\mathbb{R}^m\right)$ such that

$$\mathcal{L}(\phi)_t(x) = \tilde{\alpha}_t x + \tilde{\beta}_t,$$

for all $t \in [0,T]$ and $x \in \mathbb{R}^n$.

Proof. Fix $t \in [0,T]$ and $x \in \mathbb{R}^n$. Since the coefficients B, D, R, and S in (3.48) are continuous in t, it suffices to show that $Y_t^{t,x,\phi}, Z_t^{t,x,\phi}$ are linear in x, and the linear coefficients are continuous in t. Substituting the definition of ϕ into (3.49) gives us, for $s \in [t,T]$,

$$\begin{aligned}
dX_s^{t,x,\phi} &= \left(\left(A_s + B_s\alpha_s\right) X_s^{t,x,\phi} + B_s\beta_s\right) ds + \left(\left(C_s + D_s\alpha_s\right) X_s^{t,x,\phi} + D_s\beta_s\right) dW_s, \\
X_t^{t,x,\phi} &= x, \\
dY_s^{t,x,\phi} &= -\left(A_s^\top Y_s^{t,x,\phi} + C_s^\top Z_s^{t,x,\phi} + S_s^\top \beta_s + \left(Q_s + S_s^\top \alpha_s\right) X_s^{t,x,\phi}\right) ds + \left(Z_s^{t,x,\phi}\right)^\top dW_s, \\
Y_T^{t,x,\phi} &= G X_T^{t,x,\phi}.
\end{aligned}$$

By Feynmann-Kac Theorem, if there exists a function $v^\phi \in \mathcal{C}^{1,2}\left([t,T] \times \mathbb{R}^n;\mathbb{R}^n\right)$ solving the

PDE, for $s \in [t,T]$ and $\tilde{x} \in \mathbb{R}^n$

$$\begin{aligned}
&\partial_s v_s^\phi(\tilde{x}) + \partial_x v_s^\phi(\tilde{x})\left((A_s + B_s\alpha_s)\tilde{x} + B_s\beta_s\right) \\
&+ \frac{1}{2}\left((C_s + D_s\alpha_s)\tilde{x} + D_s\beta_s\right)^\top \partial_{xx}^2 v_s^\phi(\tilde{x})\left((C_s + D_s\alpha_s)\tilde{x} + D_s\beta_s\right) \\
&+ A_s^\top v_s^\phi(\tilde{x}) + C_s^\top \partial_x v_s^\phi(\tilde{x})\left((C_s + D_s\alpha_s)\tilde{x} + D_s\beta_s\right) + S^\top\beta_s + \left(Q_s + S_s^\top\alpha_s\right)\tilde{x} = 0,
\end{aligned} \tag{3.51}$$

with $v_T^\phi(\tilde{x}) = G\tilde{x}$ for $\tilde{x} \in \mathbb{R}^n$, then we have

$$Y_s^{t,x,\phi} = v_s^\phi\left(X_s^{t,x,\phi}\right) \qquad Z_s^{t,x,\phi} = \partial_x v_s^\phi\left(X_s^{t,x,\phi}\right)\left((C_s + D_s\alpha_s)X_s^{t,x,\phi} + D_s\beta_s\right) \tag{3.52}$$

for $s \in [t,T]$. We propose $v_s^\phi(\tilde{x}) = a_s\tilde{x} + b_s$ for $s \in [t,T]$ and $\tilde{x} \in \mathbb{R}^n$, for some $a \in \mathcal{C}^1([t,T];\mathbb{R}^{n\times n})$ and $b \in \mathcal{C}^1([t,T];\mathbb{R}^n)$. We then have

$$\begin{aligned}
\partial_s v_s^\phi(\tilde{x}) &= \dot{a}_s\tilde{x} + \dot{b}_s, \\
\partial_x v_s^\phi(\tilde{x}) &= a_s, \\
\partial_{xx}^2 v_s^\phi(x) &= 0,
\end{aligned}$$

where $\dot{a}$ denotes the time derivative of a. Substituting these into (3.51), for any $\tilde{x} \in \mathbb{R}^n$ and $s \in [t,T]$ we have

$$\begin{aligned}
&\dot{a}_s\tilde{x} + \dot{b}_s + a_s\left((A_s + B_s\alpha_s)\tilde{x} + B_s\beta_s\right) \\
&+ A_s^\top(a_s\tilde{x} + b_s) + C_s^\top a_s\left((C_s + D_s\alpha_s)\tilde{x} + D_s\beta_s\right) + S_s^\top\beta_s + \left(Q_s + S_s^\top\alpha_s\right)\tilde{x} = 0.
\end{aligned}$$

As $\tilde{x}$ was arbitrary, we have ODEs

$$\begin{aligned}
\dot{a}_s + a_s(A_s + B_s\alpha_s) + A_s^\top a_s + C_s^\top a_s(C_s + D_s\alpha_s) + \left(Q_s + S_s^\top\alpha_s\right) &= 0, \\
\dot{b}_s + A_s^\top b_s + \left(a_s B_s + C_s^\top a_s D_s + S_s^\top\right)\beta_s &= 0,
\end{aligned} \tag{3.53}$$

for $s \in [t,T]$ with $a_T = G$ and $b_T = 0$. Let us solve these equations. As these ODEs are linear, and α, β are continuous, the solutions are continuously differentiable. For arbitrary $\xi \in \mathcal{C}^0([0,T];\mathbb{R}^{m\times n})$, define a linear operator L_ξ on $\mathcal{C}^1([t,T];\mathbb{R}^{n\times n})$ by

$$(L_\xi f)_s := -\int_s^T f_r(A_r + B_r\xi_r) + A_r^\top f_r + C_r^\top f_r(C_r + D_r\xi_r)\,dr, \qquad t \le s \le T.$$

The first ODE in (3.53) can then be written as

$$a_s - (L_\alpha a)_s = G - \int_s^T (Q_r + S_r\alpha_r)\,dr, \qquad t \le s \le T.$$

To find an explicit representation of a we need to find the inverse of the linear operator $1 - L_\alpha$, where 1 is the identity operator. If we can show that the Neumann partial sums $L^{(N)} := \sum_{k=0}^{N} L_\alpha^k$ converge with respect to the operator norm, defined for any $L\colon \mathcal{C}^1([t,T],\mathbb{R}^{n\times n}) \to \mathcal{C}^1([t,T],\mathbb{R}^{n\times n})$ as

$$\|L\|_{\text{op}} = \sup_{\|f\|_\infty = 1} \|Lf\|_\infty, \qquad \|f\|_\infty = \sup_{s\in[t,T]} |f_s|, \qquad f \in \mathcal{C}^1\left([t,T];\mathbb{R}^{n\times n}\right),$$

then we have $(1 - L_\alpha)^{-1} = \sum_{k=0}^{\infty} L_\alpha^k$, where we use the convention $L_\alpha^0 = 1$. To see this, note that the condition $\lim_{N\to\infty} \|L^{(N)}\|_{\text{op}} < \infty$ implies $\lim_{k\to\infty} L_\alpha^k = 0$. We then have, for any $N \in \mathbb{N}_0$ and $f \in \mathcal{C}^1([t,T],\mathbb{R}^{n\times n})$

$$(1 - L_\alpha)\, L^{(N)} f = \sum_{k=0}^{N} \left(L_\alpha^k f - L_\alpha^{k+1} f\right) = L_\alpha^0 f - L_\alpha^N f = f - L_\alpha^N f.$$

Taking the limit as $N \to \infty$, and using the continuity of L_α yields

$$(1 - L_\alpha)\left(\sum_{k=0}^{\infty} L_\alpha^k\right) f = f,$$

for all $f \in \mathcal{C}^1([t,T];\mathbb{R}^{n\times n})$, hence $\lim_{N\to\infty}(1 - L_\alpha)\, L_N = 1$. Similarly we have $\lim_{N\to\infty} L_N (1 - L_\alpha) = 1$, so the inverse is well defined. We proceed to show that $\lim_{N\to\infty} \|L_N\|_{\text{op}} < \infty$. Define constants

$$\begin{aligned} c_1 &:= 2\|A\|_\infty + \|C\|_\infty^2, \\ c_2 &:= \|B\|_\infty + \|C\|_\infty \|D\|_\infty. \end{aligned}$$

Then we claim that for any $k \in \mathbb{N}_0$ we have

$$\|L_\alpha^k\|_{\text{op}} \le \frac{T^k \left(c_1 + c_2\|\alpha\|_\infty\right)^k}{k!}. \tag{3.54}$$

When $k = 0$ both sides are equal to one, so we consider $k \in \mathbb{N}$. Let $f \in C^1([t,T];\mathbb{R}^{n\times n})$ have

$\|f\|_\infty = 1$. Then we have

$$
\begin{aligned}
&\|L_\alpha^k f\|_\infty \\
&= \sup_{s\in[t,T]} \left|\left(L_\alpha L_\alpha^{k-1} f\right)_s\right| \\
&= \sup_{s\in[t,T]} \left|\int_s^T L_\alpha^{k-1} f_{r_1} \left(A_{r_1} + B_{r_1}\alpha_{r_1}\right) + A_{r_1}^\top L_\alpha^{k-1} f_{r_1} + C_{r_1}^\top L_\alpha^{k-1} f_{r_1} \left(C_{r_1} + D_{r_1}\alpha_{r_1}\right) dr_1\right| \\
&\le \sup_{s\in[t,T]} \left|\int_s^T \left(2\|A\|_\infty + \|C\|_\infty^2 + \left(\|B\|_\infty + \|C\|_\infty\|D\|_\infty\right)\|\alpha\|_\infty\right) |L_\alpha^{k-1} f_{r_1}| dr_1\right| \\
&= (c_1 + c_2\|\alpha\|_\infty) \int_t^T |L_\alpha^{k-1} f_{r_1}| dr_1 \\
&= (c_1 + c_2\|\alpha\|_\infty) \int_t^T \left|\int_{r_1}^T L_\alpha^{k-2} f_{r_2} \left(A_{r_2} + B_{r_2}\alpha_{r_2}\right) + A_{r_2}^\top L_\alpha^{k-2} f_{r_2} + C_{r_2}^\top L_\alpha^{k-2} f_{r_2} \left(C_{r_2} + D_{r_2}\alpha_{r_2}\right) dr_2\right| dr_1 \\
&\le (c_1 + c_2\|\alpha\|_\infty) \int_t^T \left|\int_{r_1}^T \left(2\|A\|_\infty + \|C\|_\infty^2 + \left(\|B\|_\infty + \|C\|_\infty\|D\|_\infty\right)\|\alpha\|_\infty\right) |L_\alpha^{k-2} f_{r_2}| dr_2\right| dr_1 \\
&= (c_1 + c_2\|\alpha\|_\infty)^2 \int_t^T \int_{r_1}^T |L_\alpha^{k-2} f_{r_2}| dr_2 dr_1 \\
&\dots \\
&= (c_1 + c_2\|\alpha\|_\infty)^k \int_t^T \int_{r_1}^T \dots \int_{r_{k-1}}^T |f_{r_k}| dr_k \dots dr_1 \\
&\le (c_1 + c_2\|\alpha\|_\infty)^k \int_t^T \int_{r_1}^T \dots \int_{r_{k-1}}^T \|f\|_\infty dr_k \dots dr_1 \\
&= (c_1 + c_2\|\alpha\|_\infty)^k \Psi_k(t)
\end{aligned}
$$

where we define a function $\Psi_k \colon [t,T] \to R$ by

$$
\Psi_k(s) := \int_s^T \int_{r_1}^T \dots \int_{r_{k-1}}^T 1 dr_k \dots dr_k, \qquad t \le s \le T,
$$

We show by induction that $\Psi_k(s) = \frac{(T-s)^k}{k!}$ for all $s \in [t,T]$. When $k=1$ both sides are equal to $(T-s)$. If we assume that the statement holds at k, we then have

$$
\Psi_{k+1}(s) = \int_s^T \Phi_k(r)\, dr = \int_s^T \frac{(T-r)^k}{k!} dr = \frac{(T-s)^{k+1}}{(k+1)!},
$$

as required. Hence $\Psi_k(t) = \frac{(T-t)^k}{k!} \le \frac{T^k}{k!}$ for all $k \in \mathbb{N}$. As f was arbitrary, the result (3.54) follows.

By (3.54), we have

$$\left\|\sum_{k=0}^{N} L_{\alpha}^{k}\right\|_{\mathrm{op}} \leq \sum_{k=0}^{N} \frac{T^{k}\left(c_{1}+c_{2}\|\alpha\|_{\infty}\right)^{k}}{k!} \rightarrow e^{T\left(c_{1}+c_{2}\|\alpha\|_{\infty}\right)}<\infty, \qquad N \rightarrow \infty. \tag{3.55}$$

Therefore the inverse operator $\left(1-L_{\alpha}\right)^{-1}=\sum_{j=0}^{\infty} L_{\alpha}^{j}$ is well defined, and we have the explicit solution

$$a_{s}=\left(1-L_{\alpha}\right)^{-1}\left(G-\int_{\cdot}^{T}\left(Q_{r}+S_{r}^{\top} \alpha_{r}\right) d r\right)_{s}, \qquad t \leq s \leq T. \tag{3.56}$$

A similar analysis yields

$$b_{s}=\left(1-L_{A}\right)^{-1}\left(-\int_{\cdot}^{T}\left(a_{r} B_{r}+C_{r}^{\top} a_{r} D_{r}+S_{r}^{\top}\right) \beta_{r} d r\right)_{s}, \qquad t \leq s \leq T,$$

where we define the operator L_{A} on $\mathcal{C}^{1}\left([0, T] ; \mathbb{R}^{n}\right)$ by

$$\left(L_{A} g\right)_{s}:=-\int_{s}^{T} A_{r}^{\top} g_{r} d r, \qquad t \leq s \leq T.$$

for $g \in \mathcal{C}^{1}\left([0, T] ; \mathbb{R}^{n}\right)$. Given this solution for v^{ϕ}, we can substitute into (3.52) to give the BSDE solutions

$$Y_{t}^{t, x, \phi}=a_{t} x+b_{t}, \qquad Z_{t}^{t, x, \phi}=a_{t}\left(\left(C_{t}+D_{t} \alpha_{t}\right) x+D_{t} \beta_{t}\right), \tag{3.57}$$

which are linear functions of x with coefficients that are continuous in t. Note that the ODEs governing a and b are independent of the choice of t, so can be extended to solutions on the entire horizon $[0, T]$. □

We have shown that our update maps $\mathcal{V}_{L}:=\{\phi \in \mathcal{V} \colon \phi \text{ linear}\}$ to its self. We can write the updates in (3.50) as the following, for $t \in[0, T]$, $x \in \mathbb{R}^{n}$ and $k \in \mathbb{N}_{0}$,

$$\begin{aligned}
\phi_{t}^{k}(x) &= \alpha_{t}^{k} x+\beta_{t}^{k}, \\
Y_{t}^{t, x, \phi^{k}} &= a_{t}^{k} x+b_{t}^{k}, \\
Z_{t}^{t, x, \phi^{k}} &= a_{t}^{k}\left(\left(C_{t}+D_{t} \alpha_{t}^{k}\right) x+D_{t} \beta_{t}^{k}\right).
\end{aligned} \tag{3.58}$$

For some $\alpha^{k} \in \mathcal{C}^{0}\left([0, T] ; \mathbb{R}^{m \times n}\right)$, $\beta^{k} \in \mathcal{C}^{0}\left([0, T] ; \mathbb{R}^{m}\right)$, $a^{k} \in \mathcal{C}^{1}\left([0, T] ; \mathbb{R}^{n \times n}\right)$ and $b^{k} \in \mathcal{C}^{1}\left([0, T] ; \mathbb{R}^{n}\right)$

to be determined. Substituting (3.58) into (3.50) yields, for $t \in [0,T]$ and $x \in \mathbb{R}^n$,

$$\begin{aligned}
\phi_t^{k+1}(x) &= \phi_t^k(x) - \tau\left(B_t^\top Y_t^{t,x,\phi^k} + D_t^\top Z_t^{t,x,\phi^k} + R_t\phi_t^k(x) + S_t x\right) \\
&= \alpha_t^k x + \beta_t^k - \tau\left(B_t^\top\left(a_t^k x + b_t^k\right) + D_t^\top a_t^k\left(\left(C_t + D_t\alpha_t^k\right)x + D_t\beta_t^k\right) + R_t\left(\alpha_t^k x + \beta_t^k\right) + S_t x\right) \\
&= \left(\alpha_t^k - \tau\left(B_t^\top a_t^k + D_t^\top a_t^k\left(C_t + D_t\alpha_t^k\right) + R_t\alpha_t^k + S_t\right)\right)x \\
&+ \beta_t^k - \tau\left(B_t^\top b_t^k + \left(D_t^\top a_t^k D_t + R_t\right)\beta_t^k\right),
\end{aligned}$$

giving us the update formula for $k \in \mathbb{N}_0$ and $t \in [0,T]$,

$$\begin{aligned}
\alpha_t^{k+1} &= \alpha_t^k - \tau\left(B_t^\top a_t^k + D_t^\top a_t^k\left(C_t + D_t\alpha_t^k\right) + R_t\alpha_t^k + S_t\right), \\
\beta_t^{k+1} &= \beta_t^k - \tau\left(B_t^\top b_t^k + \left(D_t^\top a_t^k D_t + R_t\right)\beta_t^k\right),
\end{aligned} \tag{3.59}$$

where, for $s \in [0,T]$,

$$\begin{aligned}
a_s^k &= \left(1 - L_{\alpha^k}\right)^{-1}\left(G - \int_\cdot^T \left(Q_r + S_r^\top\alpha_r\right)dr\right)_s, \\
b_s^k &= \left(1 - L_A\right)^{-1}\left(-\int_\cdot^T \left(a_r B_r + C_r^\top a_r D_r + S_r^\top\right)\beta_r dr\right)_s,
\end{aligned} \tag{3.60}$$

and

$$\begin{aligned}
\left(L_{\alpha^k} f\right)_s &= -\int_s^T f_r\left(A_r + B_r\alpha_r^k\right) + A_r^\top f_r + C_r^\top f_r\left(C_r + D_r\alpha_r^k\right)dr, \\
\left(L_A g\right)_s &= -\int_s^T A_r^\top g_r dr.
\end{aligned}$$

for $f \in \mathcal{C}^1\left([0,T];\mathbb{R}^{n\times n}\right)$, $g \in \mathcal{C}^1\left([0,T];\mathbb{R}^n\right)$. Given these updates, we can bound the linear coefficients of consecutive BSDE solutions, which will lead to a bound on the linear coefficients of consecutive control policies. Define constants

$$\begin{aligned}
c_3 &:= |G| + T\|Q\|_\infty, \\
c_4 &:= T\|S\|_\infty.
\end{aligned}$$

Using the definition of the operator norm and (3.55) we have, for any $k \in \mathbb{N}_0$

$$\begin{aligned}
\|a^k\|_\infty &\leq \| \left(1 - L_{\alpha^k}\right)^{-1} \|_{\text{op}} \sup_{t\in[0,T]} \left| G - \int_t^T \left(Q_s + S_s^\top \alpha_s^k\right) ds \right| \\
&\leq \sum_{j=0}^{\infty} \|L_{\alpha^k}^j\|_{\text{op}} \left(|G| + T\left(\|Q\|_\infty + \|S\|_\infty \|\alpha^k\|_\infty\right)\right) \\
&= \sum_{j=0}^{\infty} \|L_{\alpha^k}^j\|_{\text{op}} \left(c_3 + c_4\|\alpha^k\|_\infty\right) \\
&\leq \sum_{j=0}^{\infty} \frac{T^j \left(c_1 + c_2\|\alpha^k\|_\infty\right)^j}{j!} \left(c_3 + c_4\|\alpha^k\|_\infty\right) \\
&= e^{T\left(c_1 + c_2\|\alpha^k\|_\infty\right)} \left(c_3 + c_4\|\alpha^k\|_\infty\right).
\end{aligned}$$

Similarly we may bound the b^k function, yielding the estimates for all $k \in \mathbb{N}_0$

$$\begin{aligned}
\|a^k\|_\infty &\leq e^{T\left(c_1 + c_2\|\alpha^k\|_\infty\right)} \left(c_3 + c_4\|\alpha^k\|_\infty\right), \\
\|b^k\|_\infty &\leq e^{T\|A\|_\infty} \left(c_2 T\|a^k\|_\infty + c_4\right) \|\beta^k\|_\infty
\end{aligned} \tag{3.61}$$

Finally, we give the relation between both the weighted norm $|\cdot|_0$ and the uniform Lipschitz coefficient $[\cdot]_1$ and linear coefficients of $\phi \in \mathcal{V}_L$.

Lemma 3.5.2. *Let $\phi \in \mathcal{V}_L$. Write $\phi_t(x) = \alpha_t x + \beta_t$ for $t \in [0, T]$ and $x \in \mathbb{R}^m$ for some $\alpha \in \mathcal{C}^0([0, T], \mathbb{R}^{m\times n})$ and $\beta \in \mathcal{C}^0([0, T], \mathbb{R}^m)$. Then*

$$\begin{aligned}
[\phi]_1 &= \|\alpha\|_\infty, \\
|\phi|_0 &= \max\left(\|\alpha\|_\infty, \|\beta\|_\infty\right).
\end{aligned}$$

Proof. For the first equality, we note that ϕ is differentiable with respect to x, hence the Lipschitz coefficient can be related to the bound on the first derivative as follows.

$$[\phi]_1 = \sup_{(x,t)\in\mathbb{R}^n\times[0,T]} |\partial_x \phi_t(x)| = \sup_{t\in[0,T]} |\alpha_t| = \|\alpha\|_\infty.$$

For the second equality note that, for any $x \in \mathbb{R}^n$ and $t \in [0, T]$

$$|\phi_t(x)| \leq |\phi_t(0)| + [\phi]_1|x| = |\beta_t| + \|\alpha\|_\infty |x| \leq \max\left(\|\beta\|_\infty, \|\alpha\|_\infty\right)(1 + |x|).$$

Which implies $|\phi|_0 \leq \max\left(\|\alpha\|_\infty, \|\beta\|_\infty\right)$. Since α and β are continuous on the closed interval

$[0, T]$, we may define $t_\alpha = \arg\max_{s\in[0,T]} |\alpha_s|$ and $t_\beta = \arg\max_{s\in[0,T]} |\beta_s|$. We then have

$$\begin{aligned} \|\alpha\|_\infty &= \lim_{|x|\to\infty} \frac{|\phi_{t_a}(x)|}{1+|x|} \le |\phi|_0, \\ \|\beta\|_\infty &= \frac{|\phi_{t_\beta}(0)|}{1+|0|} \le |\phi|_0. \end{aligned}$$

Hence $\max(\|\alpha\|_\infty, \|\beta\|_\infty) \le |\phi|_0$, as required. $\square$

3.5.1 Showing $[\phi^k]_1$ is Uniformly Bounded

Theorem 3.5.3. *For $\phi \in \mathcal{V}_L$, $t \in [0, T]$, $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{H}^2(t, T; \mathbb{R}^n)$ be solutions to (3.10). Then for all $\phi \in \mathcal{V}_L$, there exists $L_Y = L_Y([\phi]_1) \ge 0$ such that for all $t \in [0, T]$, $x_1, x_2 \in \mathbb{R}^n$*

$$|Y_t^{t,x_1,\phi} - Y_t^{t,x_2,\phi}| \le L_Y([\phi]_1)|x_1 - x_2|$$

Proof. Let $\phi_t(x) = \alpha_t x + \beta_t$ for some $\alpha \in \mathcal{C}^0([0, T]; \mathbb{R}^{m\times n})$ and $\beta \in \mathcal{C}^0([0, T]; \mathbb{R}^m)$. Since $Y_t^{t,\cdot,\phi}$ is linear in x, this follows automatically. Let $Y_t^{t,x,\phi} = a_t x + b_t$ for $a \in \mathcal{C}^1([0, T]; \mathbb{R}^{n\times n})$ and $b \in \mathcal{C}^1([0, T]; \mathbb{R}^n)$ solving (3.53). Then

$$L_Y([\phi]_1) = L_Y(\|\alpha\|_\infty) = \|a\|_\infty.$$

In particular, following the same logic as 3.61) we can write out a bound on $\|a\|_\infty$ explicitly to yield

$$L_Y(\|\alpha\|_\infty) = e^{T(c_1+c_2\|\alpha\|_\infty)}(c_3 + c_4\|\alpha\|_\infty). \tag{3.62}$$

$\square$

Theorem 3.5.4. *For $\phi \in \mathcal{V}_L$, $t \in [0, T]$, $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{H}^2(t, T; \mathbb{R}^n)$ be solutions to (3.10). Then for all $\phi \in \mathcal{V}_L$, there exists $L_Z = L_Z([\phi]_1) \ge 0$ such that for all $t \in [0, T]$, $x_1, x_2 \in \mathbb{R}^n$*

$$|Z_t^{t,x_1,\phi} - Z_t^{t,x_2,\phi}| \le L_Z([\phi]_1)|x_1 - x_2|$$

Proof. Let $\phi_t(x) = \alpha_t x + \beta_t$ for some $\alpha \in \mathcal{C}^0([0,T];\mathbb{R}^{m\times n})$ and $\beta \in \mathcal{C}^0([0,T];\mathbb{R}^m)$. Again, Since $Z_t^{t,\cdot,\phi}$ is linear in x, this follows automatically. Let $Y_t^{t,x,\phi} = a_t x + b_t$ for $a \in \mathcal{C}^1([0,T];\mathbb{R}^{n\times n})$ and $b \in \mathcal{C}^1([0,T];\mathbb{R}^n)$ solving (3.53). Then using the representation of $Z^{t,x,\phi}$ in (3.57), we have

$$L_Z([\phi]_1) = L_Z(\|\alpha\|_\infty) = \|a\|_\infty (\|C\|_\infty + \|D\|_\infty \|\alpha\|_\infty).$$

In particular, following the same logic as 3.61) we can write out a bound on $\|a\|_\infty$ explicitly to yield

$$L_Z(\|\alpha\|_\infty) = e^{T(c_1 + c_2\|\alpha\|_\infty)} (c_3 + c_4\|\alpha\|_\infty) (\|C\|_\infty + \|D\|_\infty \|\alpha\|_\infty). \tag{3.63}$$

$\square$

Using the fact that, for $a, b, c, d, x \geq 0$

$$(a + bx)(c + dx) \leq ac + \frac{1}{2}(ad + bc)^2 + \left(bd + \frac{1}{2}(ad + bc)^2\right) x^2,$$

we have

$$L_Y\left([\phi^k]_1\right) = L_Y\left(\|\alpha^k\|_\infty\right) = e^{c_1 T}\left(c_3 + c_4\|\alpha^k\|_\infty\right) e^{c_2 T\|\alpha^k\|_\infty}, \tag{3.64}$$

$$L_Z\left([\phi^k]_1\right) = L_Z\left(\|\alpha^k\|_\infty\right) = e^{c_1 T}\left(c_5 + c_6\|\alpha^k\|_\infty^2\right) e^{c_2 T\|\alpha^k\|_\infty}. \tag{3.65}$$

where we define the constants

$$c_5 := c_3\|C\|_\infty + \frac{1}{2}(c_3\|C\|_\infty + c_4\|D\|_\infty)^2,$$
$$c_6 := c_4\|D\|_\infty + \frac{1}{2}(c_3\|C\|_\infty + c_4\|D\|_\infty)^2.$$

Theorem 3.5.5. *Let $\phi^0 \in \mathcal{V}_L$. Define constants*

$$c_7 := \max\left(c_2 T[\phi^0]_1, \frac{c_2 T\|S\|_\infty + c_0}{c_0}\right),$$
$$c_8 := \|S\|_\infty + \left(\|B\|_\infty c_3 + \|D\|_\infty c_5 e^{c_1 T + c_7}\right).$$

Assume the following

$$\begin{aligned}
2\|B\|_\infty c_2 c_3 T e^{c_1 T + c_7} &\leq c_0, \\
2\|D\|_\infty c_2 c_5 T e^{c_1 T + c_7} &\leq c_0, \\
2\|B\|_\infty c_4 e^{c_1 T + c_7} &\leq c_0, \\
2\|D\|_\infty \frac{c_6 c_7}{c_2 T} e^{c_1 T + c_7} &\leq c_0.
\end{aligned} \tag{3.66}$$

Then the sequences $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ *generated by (3.12) are Lipschitz, and the sequence of Lipschitz coefficients* $\left([\phi^k]_1\right)_{k\in\mathbb{N}_0}$ *satisfies*

$$\sup_{k\in\mathbb{N}_0}[\phi^k]_1 = \sup_{k\in\mathbb{N}_0}\|\alpha^k\|_\infty \leq c_\alpha := [\phi^0]_1 + \frac{c_8}{c_0}.$$

Remark 14. The assumptions given in (3.66) is satisfied for large c_0. Recalling that

$$c_0 = \frac{1}{2}\frac{\mu\|R\|_\infty}{\mu + \|R\|_\infty},$$

we will have c_0 large for large values of μ and $\|R\|_\infty$, as described by Assumption 1. The left hand side of all 4 assumptions only depend on μ and $\|R\|_\infty$ through c_7, and we have

$$\lim_{c_0\to\infty}\frac{c_2 T\|S\|_\infty + c_0}{c_0} = 1 < \infty.$$

Therefore, the left hand sides are bounded while the right hand sides are unbounded. This logic can be applied to later assumptions, c.f. (3.73) and (3.94).

Proof. Using Theorem 3.3.3 with $\phi^1 = \phi^2 = \phi^k$, and maximising over $x_1 \neq x_2$ and $t \in [0,T]$ we have, for all $k \in \mathbb{N}_0$

$$[\phi^{k+1}]_1 \leq (1 - 2\tau c_0)\,[\phi^k]_1 + \tau\|S\|_\infty + \tau\|B\|_\infty L_Y\left([\phi^k]_1\right) + \tau\|D\|_\infty L_Z\left([\phi^k]_1\right). \tag{3.67}$$

By Lemma 3.5.2, and plugging in (3.64) and (3.65), this can be re-written as

$$\begin{aligned}
\|\alpha^{k+1}\|_\infty \leq{}& (1 - 2\tau c_0)\,\|\alpha^k\|_\infty + \tau\|S\|_\infty \\
&+ \tau\|B\|_\infty\left(c_3 + c_4\|\alpha^k\|_\infty\right)e^{\left(c_1 + c_2\|\alpha^k\|_\infty\right)T} \\
&+ \tau\|D\|_\infty\left(c_5 + c_6\|\alpha^k\|_\infty^2\right)e^{\left(c_1 + c_2\|\alpha^k\|_\infty\right)T}.
\end{aligned} \tag{3.68}$$

First, we will prove by induction that $\sup_{k\in\mathbb{N}_0} c_2T\|\alpha^k\|_\infty \leq c_7$. When $k=0$ we have $c_2T\|\alpha^0\|_\infty \leq c_7$ by definition of c_7. So suppose that $c_2T\|\alpha^k\|_\infty \leq c_7$ for some fixed $k \in \mathbb{N}_0$. Then by (3.68) we have

$$\begin{aligned} c_2T\|\alpha^{k+1}\|_\infty &\leq (1-2\tau c_0)\, c_2T\|\alpha^k\|_\infty + \tau c_2T\|S\|_\infty \\ &\quad + \tau\|B\|_\infty c_2 \left(c_3 + c_4\|\alpha^k\|_\infty\right) Te^{\left(c_1+c_2\|\alpha^k\|_\infty\right)T} \\ &\quad + \tau\|D\|_\infty c_2 \left(c_5 + c_6\|\alpha^k\|_\infty^2\right) Te^{\left(c_1+c_2\|\alpha^k\|_\infty\right)T}. \end{aligned} \tag{3.69}$$

Using the induction assumption, (3.66) and the definition of c_7, we have

$$\begin{aligned} c_2T\|\alpha^{k+1}\|_\infty &\leq (1-2\tau c_0)\, c_7 + \tau c_2T\|S\|_\infty \\ &\quad + \tau\|B\|_\infty \left(c_2c_3T + c_4c_7\right) e^{c_1T+c_7} \\ &\quad + \tau\|D\|_\infty \left(c_2c_5T + \frac{c_6c_7^2}{c_2T}\right) e^{c_1T+c_7} \\ &\leq (1-2\tau c_0)\, c_7 + \tau\left(c_2T\|S\|_\infty + c_0\right) + \tau c_0c_7 \\ &\leq (1-2\tau c_0)\, c_7 + 2\tau c_0c_7 \\ &= c_7, \end{aligned}$$

as required. Substituting this bound into (3.69) divided by c_2T, and using the first two assumptions in (3.66) yields, for all $k \in \mathbb{N}_0$

$$\begin{aligned} \|\alpha^{k+1}\|_\infty &\leq (1-2\tau c_0)\,\|\alpha^k\|_\infty + \tau\|S\|_\infty \\ &\quad + \tau\|B\|_\infty c_3e^{c_1T+c_7} + \tau\|D\|_\infty c_5e^{c_1T+c_7} \\ &\quad + \tau\|\alpha^k\|_\infty \left(\|B\|_\infty c_4e^{c_1T+c_7} + \|D\|_\infty \frac{c_6c_7}{c_2T}e^{c_1T+c_7}\right) \\ &\leq (1-\tau c_0)\,\|\alpha^k\|_\infty + \tau\left(\|S\|_\infty + \|B\|_\infty c_3e^{c_1T+c_7} + \|D\|_\infty c_5e^{c_1T+c_7}\right). \\ &= (1-\tau c_0)\,\|\alpha^k\|_\infty + \tau c_8. \end{aligned}$$

This is a linear difference inequality. A sequence satisfying this iterative inequality is bounded by the solution of the corresponding linear difference equation. To be specific, we have, for all $k \in \mathbb{N}_0$,

$$\|\alpha^k\|_\infty \leq x_k, \tag{3.70}$$

where x_k satisfies the difference equation, for $k \in \mathbb{N}_0$,

$$x_0 = \|\alpha^0\|_\infty,$$
$$x_{k+1} = (1 - \tau c_0)\, x_k + \tau c_8.$$

Since $|1 - \tau c_0| \leq 1$ by Assumption 1, this difference equation has solution

$$x_k = (1 - \tau c_0)^k \|\alpha^0\|_\infty + \frac{1 - (1 - \tau c_0)^k}{\tau c_0} \tau c_8, \qquad k \in \mathbb{N}_0.$$

Therefore the inequality (3.70) becomes

$$\begin{aligned} [\phi^k]_1 &\leq (1 - \tau c_0)^k [\phi^0]_1 + \frac{\left(1 - (1 - \tau c_0)^k\right)}{\tau c_0} \tau c_8 \\ &\leq [\phi^0]_1 + \frac{c_8}{c_0}, \end{aligned}$$

for all $k \in \mathbb{N}_0$, as required. □

3.5.2 Showing $|\phi^k|_0$ is Uniformly Bounded

We have shown that the sequence of Lipschitz coefficients are uniformly bounded. Now we will do the same for the sequence of weighted norms $\left(|\phi^k|_0\right)_{k \in \mathbb{N}_0}$.

Theorem 3.5.6. *Define a non negative, non decreasing function* $C_a \colon [0, \infty) \to [0, \infty)$ *by*

$$C_a(x) := e^{(c_1 + c_2 x)T} (c_3 + c_4 x), \qquad 0 \leq x < \infty,$$

and define constants

$$c_9 := T e^{\|A\|_\infty T} \left(c_2 C_a(c_\alpha) + \|S\|_\infty\right), \tag{3.71}$$

$$c_{10} := (1 - 2\tau c_0) + \tau \left(c_9 + \|D\|_\infty^2 C_a(c_\alpha)\right). \tag{3.72}$$

Then we have

$$\|\beta^{k+1}\|_\infty \leq c_{10} \|\beta^k\|_\infty, \qquad k \in \mathbb{N}_0.$$

Furthermore, suppose that $\phi^0 \in \mathcal{V}_L$ satisfies

$$c_9 + \|D\|_\infty^2 C_a(c_\alpha) < 2c_0. \tag{3.73}$$

Then we have $c_{10} < 1$ and $\sup_{k\in\mathbb{N}_0} \|\beta^k\|_\infty \le c_\beta := \|\beta^0\|_\infty$. In particular, each element of the sequence $(\phi^k)_{k\in\mathbb{N}_0}$ generated by (3.12) has finite $|\cdot|_0$ norm, and the sequence $(|\phi^k|_0)_{k\in\mathbb{N}_0}$ satisfies, for all $k \in \mathbb{N}_0$

$$\sup_{k\in\mathbb{N}_0} |\phi^k|_0 = \max\left(\sup_{k\in\mathbb{N}_0} \|\alpha^k\|_\infty, \sup_{k\in\mathbb{N}_0} \|\beta^k\|_\infty\right) \le \max(c_\alpha, c_\beta) < \infty.$$

Proof. Applying Theorem 3.3.3 with $\phi^1 = \phi^k$, $\phi^2 = 0$, $x_1 = x_2 = 0$, and using (3.58) yields, for any $t \in [0,T]$ and $k \in \mathbb{N}_0$,

$$\begin{aligned} |\beta_t^{k+1}| &\le (1 - 2\tau c_0)\left|\beta_t^k\right| + \tau\left(\|B\|_\infty \left|Y_t^{t,0,\phi^k}\right| + \|D\|_\infty \left|Z_t^{t,0,\phi^k}\right|\right) \\ &= (1 - 2\tau c_0)\left|\beta_t^k\right| + \tau\left(\|B\|_\infty \left|b_t^k\right| + \|D\|_\infty \left|a_t^k D\beta_t^k\right|\right). \end{aligned}$$

Taking the supremum over t yields, for all $k \in N_0$,

$$\|\beta^{k+1}\|_\infty \le (1 - 2\tau c_0)\|\beta^k\|_\infty + \tau\left(\|B\|_\infty \|b^k\|_\infty + \|D\|_\infty^2 \|a^k\|_\infty \|\beta^k\|_\infty\right). \tag{3.74}$$

Recall that by (3.61), we have the following estimates for $k \in \mathbb{N}_0$.

$$\|a^k\|_\infty \le C_a\left(\|\alpha^k\|_\infty\right), \tag{3.75}$$

$$\|b^k\|_\infty \le e^{T\|A\|_\infty}\left(c_2 T\|a^k\|_\infty + c_4\right)\|\beta^k\|_\infty. \tag{3.76}$$

Furthermore, by Theorem 3.5.5, and the fact that C_a increasing, we have

$$\sup_{k\in N_0} \|a^k\|_\infty \le C_a\left(\sup_{k\in N_0} \|\alpha^k\|_\infty\right) \le C_a(c_\alpha). \tag{3.77}$$

Substituting (3.77) into (3.76) gives, for all $k \in \mathbb{N}_0$

$$\|b^k\|_\infty \le e^{T\|A\|_\infty}(c_2 T C_a(c_\alpha) + c_4)\|\beta^k\|_\infty. \tag{3.78}$$

Further substituting (3.78) into (3.74) yields, for all $k \in \mathbb{N}_0$,

$$\|\beta^{k+1}\|_\infty \leq c_{10}\|\beta^k\|_\infty. \tag{3.79}$$

The assumption (3.73) directly implies $c_{10} < 1$, and in this case we have the following uniform bound

$$\sup_{k\in\mathbb{N}_0} \|\beta^k\|_\infty \leq \sup_{k\in\mathbb{N}_0} c_{10}^k\|\beta^0\|_\infty = \|\beta^0\|_\infty,$$

The final statement follows directly from this, Lemma 3.5.2 and Theorem 3.5.5. □

3.5.3 Showing $\mathcal{L}$ is a Contraction in $|\cdot|_0$

Theorem 3.5.7. *For $\phi \in \mathcal{V}_L$, $t \in [0,T]$ and $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi},\ Y^{t,x,\phi},\ Z^{t,x,\phi}\right) \in \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{S}^2\left(t,T;\mathbb{R}^n\right) \times \mathcal{H}^2\left(t,T;\mathbb{R}^n\right)$ be solutions to (3.10). Let $\phi^1, \phi^2 \in \mathcal{V}_L$, $t \in [0,T]$ and $x \in \mathbb{R}^n$. Denote $\phi^i_t\left(x\right) = \alpha^i_t x + \beta^i_t$ and $Y^{t,x,\phi^i}_t = a^i_t x + b^i_t$ for $\alpha^i \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m\times n}\right)$, $\beta^i \in \mathcal{C}^0\left([0,T];\mathbb{R}^m\right)$, $a^i \in \mathcal{C}^1\left([0,T];\mathbb{R}^{n\times n}\right)$ and $b^i \in \mathcal{C}^1\left([0,T];\mathbb{R}^n\right)$. Define a non negative constant $C_Y\left(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty\right) \geq 0$ by*

$$\begin{aligned}
C_Y\left(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty\right) &:= \max\left(C_{Y,1}, C_{Y,2}, C_{Y,3}\right), \\
C_{Y,1} &:= e^{\left(c_1 + c_2\|\alpha^1\|_\infty\right)T}\left(c_4 + c_2 T C_a\left(\|\alpha^2\|_\infty\right)\right), \\
C_{Y,2} &:= e^{\|A\|_\infty T} c_2 T C_{Y,1} \\
C_{Y,3} &:= e^{\|A\|_\infty T}\left(c_4 + c_2 T C_a\left(\|\alpha^2\|_\infty\right)\right).
\end{aligned}$$

Then

$$|Y^{t,x,\phi^1}_t - Y^{t,x,\phi^2}_t| \leq C_Y\left(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty\right)\left(1 + |x|\right)|\phi^1 - \phi^2|_0.$$

Note that for notational brevity we have suppressed the dependence of $C_{Y,1}$, $C_{Y,2}$, and $C_{Y,3}$ on $\|\beta^1\|_\infty$, $\|\alpha^1\|_\infty$ and $\|\alpha^2\|_\infty$.

Proof. Fix $\phi^1, \phi^2 \in \mathcal{V}_L$, $t \in [0,T]$ and $x \in \mathbb{R}^n$. Recall that by (3.53), the functions a^i, b^i satisfy the following ODEs

$$
\begin{aligned}
\dot{a}_s^i + a_s^i \left(A_s + B_s\alpha_s^i\right) + A_s^\top a_s^i + C_s^\top a_s^i \left(C_s + D_s\alpha_s^i\right) + \left(Q_s + S_s^\top \alpha_s^i\right) &= 0,\\
\dot{b}_s^i + A_s^\top b_s^i + \left(a_s^i B_s + C_s^\top a_s^i D_s + S_s^\top\right)\beta_s^i &= 0,
\end{aligned}
$$

with $a_T^i = G, b_T^i = 0$. Therefore, the functions

$$
\begin{aligned}
\delta a_s &:= a_s^1 - a_s^2,\\
\delta b_s &:= b_s^1 - b_s^2,\\
\delta \alpha_s &:= \alpha_s^1 - \alpha_s^2,\\
\delta \beta_s &:= \beta_s^1 - \beta_s^2,
\end{aligned}
$$

for $s \in [0, T]$ solve the ODEs

$$
\begin{aligned}
\dot{\delta a}_s + \delta a_s \left(A_s + B_s\alpha_s^1\right) + A_s^\top \delta a_s + C_s^\top \delta a_s \left(C_s + D_s\alpha_s^1\right) + \left(a_s^2 B_s + C_s^\top a_s^2 D_s + S_s^\top\right)\delta\alpha_s &= 0,\\
\dot{\delta b}_s + A_s^\top \delta b_s + \left(\delta a_s B_s + C_s^\top \delta a_s D_s\right)\beta_s^1 + \left(a_s^2 B_s + C_s^\top a_s^2 D_s + S_s^\top\right)\delta\beta_s &= 0,
\end{aligned}
$$

with $\delta a_T = \delta b_T = 0$. We can solve these using the same solving method as (3.56) to yield

$$
\begin{aligned}
\delta a_t &= \left(1 - L_{\alpha^1}\right)^{-1}\left(-\int_\cdot^T \left(a_s^2 B_s + C_s^\top a_s^2 D_s + S_s^\top\right)\delta\alpha_s ds\right)_t,\\
\delta b_t &= \left(1 - L_A\right)^{-1}\left(-\int_\cdot^T \left(\delta a_s B_s + C_s^\top \delta a_s D_s\right)\beta_s^1 + \left(a_s^2 B_s + C_s^\top a_s^2 D_s + S_s^\top\right)\delta\beta_s ds\right)_t,
\end{aligned}
$$

where

$$
\begin{aligned}
\left(L_{\alpha^1} f\right)_s &= -\int_s^T f_r\left(A_r + B_r\alpha_r^1\right) + A_r^\top f_r + C_r^\top f_r\left(C_r + D_r\alpha_r^1\right) dr,\\
\left(L_A g\right)_s &= -\int_s^T A_r^\top g_r dr,
\end{aligned}
$$

for $f \in \mathcal{C}^1\left([0,T];\mathbb{R}^{n\times n}\right)$, $g \in \mathcal{C}^1\left([0,T];\mathbb{R}^n\right)$ and $s \in [0,T]$. Following the same argument as

(3.61), the supremum norm of the solutions can be bounded by the following.

$$\|\delta a\|_\infty \leq e^{\left(c_1+c_2\|\alpha^1\|_\infty\right)T}\left(c_4+c_2T\|a^2\|_\infty\right)\|\delta\alpha\|_\infty, \tag{3.80}$$

$$\|\delta b\|_\infty \leq e^{\|A\|_\infty T}\left(c_2T\|\delta a\|_\infty\|\beta^1\|_\infty+\left(c_4+c_2T\|a^2\|_\infty\right)\|\delta\beta\|_\infty\right). \tag{3.81}$$

By (3.61) we have

$$\|a^2\|_\infty \leq C_a\left(\|\alpha^2\|_\infty\right). \tag{3.82}$$

Substituting (3.82) into (3.80) yields

$$\|\delta a\|_\infty \leq e^{\left(c_1+c_2\|\alpha^1\|_\infty\right)T}\left(c_4+c_2TC_a\left(\|\alpha^2\|_\infty\right)\right)\|\delta\alpha\|_\infty = C_{Y,1}\|\delta\alpha\|_\infty. \tag{3.83}$$

Substituting (3.83) and (3.82) into (3.81) yields

$$\|\delta b\|_\infty \leq e^{\|A\|_\infty T}\left(c_2TC_{Y,1}\|\delta\alpha\|_\infty\|\beta^1\|_\infty+\left(c_4+c_2TC_a\left(\|\alpha^2\|_\infty\right)\right)\|\delta\beta\|_\infty\right) \tag{3.84}$$

$$= C_{Y,2}\|\delta\alpha\|_\infty + C_{Y,3}\|\delta\beta\|_\infty. \tag{3.85}$$

Note that

$$\left|Y_t^{t,x,\phi^1}-Y_t^{t,x,\phi^2}\right| = |\delta a_t x+\delta b_t| \leq \max\left(\|\delta a\|_\infty,\|\delta b\|_\infty\right)(1+|x|), \tag{3.86}$$

and by (3.83) and (3.85) we have

$$\max\left(\|\delta a\|_\infty,\|\delta b\|_\infty\right) \leq \max\left(C_{Y,1},C_{Y,2}\right)\|\delta\alpha\|_\infty + C_{Y,3}\|\delta\beta\|_\infty \tag{3.87}$$

$$\leq C_Y\left(\|\beta^1\|_\infty,\|\alpha^1\|_\infty,\|\alpha^2\|_\infty\right)\max\left(\|\delta\alpha\|_\infty,\|\delta\beta\|_\infty\right), \tag{3.88}$$

where

$$\begin{aligned}
C_Y\left(\|\beta^1\|_\infty,\|\alpha^1\|_\infty,\|\alpha^2\|_\infty\right) &= \max\left(C_{Y,1},C_{Y,2},C_{Y,3}\right),\\
C_{Y,1} &= e^{\left(c_1+c_2\|\alpha^1\|_\infty\right)T}\left(c_4+c_2TC_a\left(\|\alpha^2\|_\infty\right)\right),\\
C_{Y,2} &= e^{\|A\|_\infty T}c_2TC_{Y,1}\\
C_{Y,3} &= e^{\|A\|_\infty T}\left(c_4+c_2TC_a\left(\|\alpha^2\|_\infty\right)\right).
\end{aligned}$$

Since the function $\phi^1 - \phi^2$ is of the form

$$\phi_s^1(\tilde{x}) - \phi_s^2(\tilde{x}) = \delta\alpha_s \tilde{x} + \delta\beta_s, \qquad (s, \tilde{x}) \in [0, T] \times \mathbb{R}^n,$$

we can apply Lemma 3.5.2 to yield

$$|\phi^1 - \phi^2|_0 = \max\left(\|\delta\alpha\|_\infty, \|\delta\beta\|_\infty\right). \tag{3.89}$$

Combining (3.86), (3.88) and (3.89) yields the result. □

Theorem 3.5.8. *For $\phi \in \mathcal{V}_L$, $t \in [0, T]$ and $x \in \mathbb{R}^n$ let $\left(X^{t,x,\phi},\, Y^{t,x,\phi},\, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{H}^2(t, T; \mathbb{R}^n)$ be solutions to (3.10). Let $\phi^1, \phi^2 \in \mathcal{V}_L$, $t \in [0, T]$ and $x \in \mathbb{R}^n$. Denote $\phi_t^i(x) = \alpha_t^i x + \beta^i$ and $Y_t^{t,x,\phi^i} = a_t^i x + b_t^i$ for $\alpha^i \in \mathcal{C}^0([0, T]; \mathbb{R}^{m\times n})$, $\beta^i \in \mathcal{C}^0([0, T]; \mathbb{R}^m)$, $a^i \in \mathcal{C}^1([0, T]; \mathbb{R}^{n\times n})$ and $b^i \in \mathcal{C}^1([0, T]; \mathbb{R}^n)$ for $i = 1, 2$. Define a constant $C_Z(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty) \geq 0$ by*

$$\begin{aligned}
C_Z\left(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty\right) &:= \max\left(C_{Z,1}, C_{Z,2}, C_{Z,3}\right), \\
C_{Z,1} &:= C_{Y,1}\left(\|C\|_\infty + \|D\|_\infty \|\alpha^1\|_\infty\right), \\
C_{Z,2} &:= C_{Y,1}\|D\|_\infty \|\beta^1\|_\infty, \\
C_{Z,3} &:= C_a\left(\|\alpha^2\|_\infty\right)\|D\|_\infty \\
C_{Y,1} &= e^{\left(c_1 + c_2\|\alpha^1\|_\infty\right)T}\left(c_4 + c_2 T C_a\left(\|\alpha^2\|_\infty\right)\right).
\end{aligned}$$

Then

$$|Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2}| \leq C_Z\left(\|\beta^1\|_\infty, \|\alpha^1\|_\infty, \|\alpha^2\|_\infty\right)(1 + |x|)\,|\phi^1 - \phi^2|_0$$

Again, for notational brevity we have suppressed the dependence of $C_{Z,1}$, $C_{Z,2}$, and $C_{Z,3}$ on $\|\beta^1\|_\infty$, $\|\alpha^1\|_\infty$ and $\|\alpha^2\|_\infty$.

Proof. Fix $\phi^1, \phi^2 \in \mathcal{V}_L$, $t \in [0, T]$ and $x \in \mathbb{R}^n$. We use the same definitions of the difference processes as the previous proof, given as follows, for $s \in [0, T]$

$$\begin{aligned}
\delta a_s &:= a_s^1 - a_s^2, \\
\delta b_s &:= b_s^1 - b_s^2, \\
\delta \alpha_s &:= \alpha_s^1 - \alpha_s^2, \\
\delta \beta_s &:= \beta_s^1 - \beta_s^2.
\end{aligned}$$

Recall that by (3.57) we have, for $i = 1, 2$,

$$Z_t^{t,x,\phi^i} = a_t^i \left(\left(C_t + D_t \alpha_t^i \right) x + D_t \beta_t^i \right). \tag{3.90}$$

We then have

$$Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2} = \left(\delta a_t \left(C_t + D_t \alpha_t^1 \right) + a_t^2 D_t \delta \alpha_t \right) x + \left(\delta a_t D_t \beta_t^1 + a_t^2 D_t \delta \beta_t \right).$$

In particular, we have

$$|Z_t^{t,x,\phi^1} - Z_t^{t,x,\phi^2}| \le \max \left(\left\| \delta a \left(C + D \alpha^1 \right) + a^2 D \delta \alpha \right\|_\infty, \left\| \delta a D \beta^1 + a^2 D \delta \beta \right\|_\infty \right) (1 + |x|). \tag{3.91}$$

We can bound the individual terms in the maximum by the following, using (3.82) and (3.83). For the first term we have

$$\begin{aligned}
&\left\| \delta a \left(C + D \alpha^1 \right) + a^2 D \delta \alpha \right\|_\infty \\
&\le \|\delta a\|_\infty \left(\|C\|_\infty + \|D\|_\infty \|\alpha^1\|_\infty \right) + \|a^2\|_\infty \|D\|_\infty \|\delta \alpha\|_\infty \\
&\le C_{Y,1} \|\delta \alpha\|_\infty \left(\|C\|_\infty + \|D\|_\infty \|\alpha^1\|_\infty \right) + C_a \left(\|\alpha^2\|_\infty \right) \|D\|_\infty \|\delta \beta\|_\infty.
\end{aligned}$$

For the second term we have

$$\begin{aligned}
&\left\| \delta a D \beta^1 + a^2 D \delta \beta \right\|_\infty \\
&\le \|D\|_\infty \|\beta^1\|_\infty \|\delta a\|_\infty + \|a^2\|_\infty \|D\|_\infty \|\delta \beta\|_\infty \\
&\le C_{Y,1} \|D\|_\infty \|\beta^1\|_\infty \|\delta \alpha\|_\infty + C_a \left(\|\alpha^2\|_\infty \right) \|D\|_\infty \|\delta \beta\|_\infty.
\end{aligned}$$

Combining these yields

$$\begin{aligned}&\max\left(\left\|\delta a\left(C+D\alpha^1\right)+a^2 D\delta\alpha\right\|_\infty,\left\|\delta a D\beta^1+a^2 D\delta\beta\right\|_\infty\right)\\&\leq C_Z\left(\|\beta^1\|_\infty,\|\alpha^1\|_\infty,\|\alpha^2\|_\infty\right)\max\left(\|\delta\alpha\|_\infty,\|\delta\beta\|_\infty\right),\end{aligned}\tag{3.92}$$

where $C_Z\left(\|\beta^1\|_\infty,\|\alpha^1\|_\infty,\|\alpha^2\|_\infty\right)=\max\left(C_{Z,1},C_{Z,2},C_{Z,3}\right)$, with

$$\begin{aligned}C_{Z,1}&=C_{Y,1}\left(\|C\|_\infty+\|D\|_\infty\|\alpha^1\|_\infty\right),\\C_{Z,2}&=C_{Y,1}\|D\|_\infty\|\beta^1\|_\infty,\\C_{Z,3}&=C_a\left(\|\alpha^2\|_\infty\right)\|D\|_\infty\\C_{Y,1}&=e^{\left(c_1+c_2\|\alpha^1\|_\infty\right)T}\left(c_4+c_2TC_a\left(\|\alpha^2\|_\infty\right)\right).\end{aligned}$$

Following the same logic of the proof of the previous Theorem, we note that

$$|\phi^1-\phi^2|_0=\max\left(\|\delta\alpha\|_\infty,\|\delta\beta\|_\infty\right).\tag{3.93}$$

The result follows from combining (3.93) with (3.91) and (3.92). □

Remark 15. Note that both functions C_Y and C_Z are combinations of linear and exponential terms with non-negative coefficients. Therefore both functions are increasing in all their respective inputs.

Given the estimates in Theorems 3.5.7 and 3.5.8, and the update formula bound in Theorem 3.3.3, we derive a difference equation for the sequence $\left(|\phi^{k+1}-\phi^k|_0\right)_{k\in\mathbb{N}_0}$. We show the conditions under which this difference equation is a contraction.

Theorem 3.5.9. *Let $\phi^0\in\mathcal{V}$ satisfy (3.66), (3.73) and additionally satisfy*

$$\|B\|_\infty C_Y\left(c_\beta,c_\alpha,c_\alpha\right)+\|D\|_\infty C_Z\left(c_\beta,c_\alpha,c_\alpha\right)\leq 2c_0.\tag{3.94}$$

Then there exists $\hat{c}\in(0,1)$, such that the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ defined by (3.12) satisfies, for all $k\in\mathbb{N}_0$

$$|\phi^{k+1}-\phi^k|_0\leq\hat{c}|\phi^k-\phi^{k-1}|_0.$$

Proof. Fix $x\in\mathbb{R}^n$, $t\in[0,T]$ and $k\in\mathbb{N}$. By Theorem 3.3.3 with $\phi^1=\phi^k$, $\phi^2=\phi^{k-1}$, and

$x_1 = x_2 = x$, we have

$$\begin{aligned} |\phi_t^{k+1}(x) - \phi_t^k(x)| \leq{}& (1-2\tau c_0)\,|\phi_t^k(x) - \phi_t^{k-1}(x)| \\ &+ \tau\|B\|_\infty |Y_t^{t,x,\phi^k} - Y_t^{t,x,\phi^{k-1}}| \\ &+ \tau\|D\|_\infty |Z_t^{t,x,\phi^k} - Z_t^{t,x,\phi^{k-1}}|. \end{aligned} \tag{3.95}$$

We have $\phi^k, \phi^{k-1} \in \mathcal{V}_L$. Hence, using Theorems 3.5.7 and 3.5.8 we have

$$\begin{aligned} \left|\phi_t^k(x) - \phi_t^{k-1}(x)\right| &\leq \left|\phi^k - \phi^{k-1}\right|_0 (1+|x|), \\ \left|Y_t^{t,x,\phi^k} - Y_t^{t,x,\phi^{k-1}}\right| &\leq C_Y\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)\left|\phi^k - \phi^{k-1}\right|_0 (1+|x|), \\ \left|Z_t^{t,x,\phi^k} - Z_t^{t,x,\phi^{k-1}}\right| &\leq C_Z\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)|\phi^k - \phi^{k-1}|_0 (1+|x|). \end{aligned}$$

Substituting these into (3.95) gives

$$\begin{aligned} |\phi_t^{k+1}(x) - \phi_t^k(x)| \leq{}& (1-2\tau c_0)\left|\phi^k - \phi^{k-1}\right|_0 (1+|x|) \\ &+ \tau\|B\|_\infty C_Y\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)\left|\phi^k - \phi^{k-1}\right|_0 (1+|x|) \\ &+ \tau\|D\|_\infty C_Z\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)\left|\phi^k - \phi^{k-1}\right|_0 (1+|x|). \end{aligned}$$

Since $x \in \mathbb{R}^n$ and $t \in [0,T]$ were arbitrary, we can take the supremum over them to yield

$$\begin{aligned} |\phi^{k+1} - \phi^k|_0 \leq{}& (1-2\tau c_0)\left|\phi^k - \phi^{k-1}\right|_0 \\ &+ \tau\|B\|_\infty C_Y\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)\left|\phi^k - \phi^{k-1}\right|_0 \\ &+ \tau\|D\|_\infty C_Z\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right)\left|\phi^k - \phi^{k-1}\right|_0. \end{aligned}$$

By Theorem 3.5.5, we have $\sup_{k\in\mathbb{N}_0} \|\alpha^k\|_\infty \leq c_\alpha$. By Theorem 3.5.6 we have $\sup_{k\in\mathbb{N}_0} \|\beta^k\|_\infty \leq c_\beta$. Using this and that the functions C_Y and C_Z are increasing in all their inputs c.f. Remark 15, we have, for all $k \in \mathbb{N}_0$

$$\begin{aligned} C_Y\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right) &\leq C_Y(c_\beta, c_\alpha, c_\alpha), \\ C_Z\left(\|\beta^k\|_\infty, \|\alpha^k\|_\infty, \|\alpha^{k-1}\|_\infty\right) &\leq C_Z(c_\beta, c_\alpha, c_\alpha). \end{aligned}$$

We may therefore write

$$|\phi^{k+1} - \phi^k|_0 \leq \hat{c}|\phi^k - \phi^{k-1}|_0,$$

where

$$\hat{c} := (1 - 2\tau c_0) + \tau\|B\|_\infty C_Y\left(c_\beta, c_\alpha, c_\alpha\right) + \tau\|D\|_\infty C_Z\left(c_\beta, c_\alpha, c_\alpha\right). \tag{3.96}$$

We require $\hat{c} < 1$, which is equivalent to

$$\|B\|_\infty C_Y\left(c_\beta, c_\alpha, c_\alpha\right) + \|D\|_\infty C_Z\left(c_\beta, c_\alpha, c_\alpha\right) \leq 2c_0$$

which is exactly (3.94). □

3.5.4 Proof of Main Result

Under the assumptions of Theorem 3.5.9, we have shown that sequence of ϕ^k are Cauchy with respect to the norm $|\cdot|_0$. We can therefore establish convergence with respect to this norm in the space of measurable functions. We now show that this limiting function indeed induces the optimal control.

Theorem 3.5.10. *Let $\phi^0 \in \mathcal{V}_L$ satisfy (3.66), (3.73) and (3.94). Let $\tau > 0$ satisfy Assumption 1. Then there exists $\phi^* \in \mathcal{V}_L$ with associated control $u^{\phi^*} \in \mathcal{H}^2\left(\mathbb{R}^m\right)$, and constants $\hat{c} \in [0,1)$ and $\hat{C} \geq 0$ such that*

1. *For all $k \in \mathbb{N}_0$, $|\phi^k - \phi^*|_0 \leq \hat{c}^k |\phi^* - \phi^0|_0$.*
2. *For all $k \in \mathbb{N}_0$, $\|u^{\phi^k} - u^{\phi^*}\|_{\mathcal{H}^2} \leq \hat{C}\hat{c}^k |\phi^* - \phi^0|_0$.*
3. *u^{ϕ^*} is a stationary point of J defined in (3.4).*

Proof. Proof is adapted from (Reisinger et al., 2022, Theorem 3.13). By Theorem 3.5.9, the sequence $\left(\phi^k\right)_{k\in\mathbb{N}_0}$ is a Cauchy sequence in the space $\left(\mathcal{B}\left([0,T] \times \mathbb{R}^n; \mathbb{R}^m\right), |\cdot|_0\right)$, which is Banach by Lemma 3.3.2. Therefore by Banach's fixed point Theorem, there exists a unique $\phi^* \in \mathcal{B}\left([0,T] \times \mathbb{R}^n; \mathbb{R}^m\right)$ such that $\lim_{k\to\infty} |\phi^k - \phi^*|_0 = 0$. We now show that $\phi^* \in \mathcal{V}_L$. Note that by Lemma 3.5.2 and Theorem 3.5.9 we have

$$\max\left(\|\alpha^{k+1} - \alpha^k\|_\infty, \|\beta^{k+1} - \beta^k\|_\infty\right) \leq \hat{c}|\phi^{k+1} - \phi^k|_0, \qquad k \in \mathbb{N}_0,$$

where $\hat{c}$ is defined in (3.96). Therefore the sequences $\left(\alpha^k\right)_{k\in\mathbb{N}_0}, \left(\beta^k\right)_{k\in\mathbb{N}_0}$ are Cauchy in the Banach spaces $\left(\mathcal{C}^0\left([0,T];\mathbb{R}^{n\times m}\right), \|\cdot\|_\infty\right)$ and $\left(\mathcal{C}^0\left([0,T];\mathbb{R}^{m}\right), \|\cdot\|_\infty\right)$ respectively, hence there exists $\alpha^* \in \mathcal{C}^0\left([0,T];\mathbb{R}^{n\times m}\right), \beta^* \in \mathcal{C}^0\left([0,T];\mathbb{R}^{m}\right)$ such that the function $\tilde{\phi} \in \mathcal{V}_L$ defined by $\tilde{\phi}_t(x) = \alpha_t^*(x)\,x + \beta_t^*(x)$ satisfies

$$\lim_{k\to\infty} |\phi^k - \tilde{\phi}|_0 = \lim_{k\to\infty} \max\left(\|\alpha^k - \alpha^*\|_\infty, \|\beta^k - \beta^*\|_\infty\right) = 0.$$

Since ϕ^* was unique, we must have $\phi^* = \tilde{\phi} \in \mathcal{V}_L$.

By Theorems 3.5.7 and 3.5.8 there exists $C \geq 0$ such that for all $t \in [0,T]$, $x \in \mathbb{R}^n$ and $k \in \mathbb{N}_0$

$$\max\left(\left|Y_t^{t,x,\phi^k} - Y_t^{t,x,\phi^*}\right|, \left|Z_t^{t,x,\phi^k} - Z_t^{t,x,\phi^*}\right|\right) \leq C\left(1+|x|\right)|\phi^k - \phi^*|_0.$$

Therefore, for fixed $(t,x) \in [0,T]\times\mathbb{R}^n$ we have the pointwise limits $\lim_{k\to\infty} Y_t^{t,x,\phi^k} = Y_t^{t,x,\phi^*}$ and $\lim_{k\to\infty} Z_t^{t,x,\phi^k} = Z_t^{t,x,\phi^*}$, and we may write

$$\begin{aligned}
\phi_t^*(x) &= \lim_{k\to\infty} \phi_t^{k+1}(x) \\
&= \lim_{k\to\infty} \mathcal{L}\left(\phi^k\right)_t(x) \\
&= \lim_{k\to\infty} \phi_t^k(x) - \tau\left(\lim_{k\to\infty} B_t^\top Y_t^{t,x,\phi^k} + D_t^\top \lim_{k\to\infty} Z_t^{t,x,\phi^k} + R_t \lim_{k\to\infty} \phi_t^k(x) + S_t x\right) \\
&= \phi_t^*(x) - \tau\left(B_t^\top Y_t^{t,x,\phi^*} + D_t^\top Z_t^{t,x,\phi^*} + R_t\phi_t^*(x) + S_t x\right).
\end{aligned}$$

Therefore

$$B_t^\top Y_t^{t,x,\phi^*} + D_t^\top Z_t^{t,x,\phi^*} + R_t\phi_t^*(x) + S_t x = 0, \qquad (t,x) \in [0,T]\times\mathbb{R}^n. \tag{3.97}$$

In particular, we have $\mathcal{L}(\phi^*) = \phi^*$. We now proceed to prove the items in the Theorem. For item 1, fix $k \in \mathbb{N}_0$. Following the same ideas as Theorem 3.5.9 and using (3.97) yields

$$|\phi^{k+1} - \phi^*|_0 = |\mathcal{L}\left(\phi^k\right) - \mathcal{L}\left(\phi^*\right)|_0 \leq \hat{c}|\phi^k - \phi^*|_0..$$

Repeating this bound k times yields the result. For item 2, note that, for any $k \in \mathbb{N}_0$

$$\begin{aligned}||u^{\phi^{k+1}} - u^{\phi^*}||_{\mathcal{H}^2} &= ||\phi^{k+1}\left(X^{0,x_0,\phi^{k+1}}\right) - \phi^*\left(X^{0,x_0,\phi^*}\right)||_{\mathcal{H}^2} \\ &\le ||\phi^{k+1}\left(X^{0,x_0,\phi^{k+1}}\right) - \phi^{k+1}\left(X^{0,x_0,\phi^*}\right)||_{\mathcal{H}^2} + ||\phi^{k+1}\left(X^{0,x_0,\phi^*}\right) - \phi^*\left(X^{0,x_0,\phi^*}\right)||_{\mathcal{H}^2} \\ &\le [\phi^{k+1}]_1 ||X^{0,x_0,\phi^{k+1}} - X^{0,x_0,\phi^*}||_{\mathcal{H}^2} + |\phi^{k+1} - \phi^*|_0 \left(1 + ||X^{0,x_0,\phi^*}||^2_{\mathcal{H}}\right).\end{aligned}$$

Using Theorem A.1.9, one can show that there exists $C \ge 0$ such that $||X^{0,x_0,\phi^*}||^2_{\mathcal{H}} \le C$ and $||X^{0,x_0,\phi^{k+1}} - X^{0,x_0,\phi^*}||_{\mathcal{H}^2} \le C|\phi^{k+1} - \phi^*|_0$, yielding the result.

We now prove item 3. Consider the BSDE, for $s \in [0,T]$,

$$\begin{aligned}dX_s^{\phi^*} &= \left(A_s X_s^{\phi^*} + B_s \phi_s^*\left(X_s^{\phi^*}\right)\right) ds + \left(C_s X_s^{\phi^*} + D_s \phi_s^*\left(X_s^{\phi^*}\right)\right) dW_s, \\ X_0^{\phi^*} &= x_0, \\ dY_s^{\phi^*} &= -\left(A_s^\top Y_s^{\phi^*} + C_s^\top Z_s^{\phi^*} + S_s^\top \phi_s^*\left(X_s^{\phi^*}\right) + Q_s X_s^{\phi^*}\right) ds + \left(Z^{\phi^*}\right)^\top dW_s, \\ Y_T^{\phi^*} &= G X_T^{\phi^*}.\end{aligned} \tag{3.98}$$

By the Markov property, we have $\left(Y_t^{\phi^*}, Z_t^{\phi^*}\right) = \left(Y_t^{t,X_t^{\phi^*},\phi^*}, Z_t^{t,X_t^{\phi^*},\phi^*}\right)$ for all $t \in [0,T]$. Hence we can use (3.97) to write, for all $t \in [0,T]$,

$$\begin{aligned}\phi_t^*\left(X_t^{\phi^*}\right) &= \phi_t^*\left(X_t^{\phi^*}\right) - \tau\left(B_t^\top Y_t^{t,X_t^{\phi^*},\phi^*} + D_t^\top Z_t^{t,X_t^{\phi^*},\phi^*} + R_t^\top \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}\right) \\ &= \phi_t^*\left(X_t^{\phi^*}\right) - \tau\left(B_t^\top Y_t^{\phi^*} + D_t^\top Z_t^{\phi^*} + R_t \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}\right).\end{aligned}$$

Item 3 then follows from noticing that J is differentiable with

$$\nabla J\left(u^{\phi^*}\right)_t = B^\top Y_t^{\phi^*} + D_t^\top Z_t^{\phi^*} + R_t^\top \phi_t^*\left(X_t^{\phi^*}\right) + S_t X_t^{\phi^*}, \qquad 0 \le t \le T.$$

□

As a by-product, we show that the optimality condition of the processes α^*, β^* is equivalent to the Ricatti equation for this LQ problem. Define the value function by

$$v(t,x) := \inf_{u \in \mathcal{U}} \mathbb{E}\left[\int_t^T \frac{1}{2}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix}^\top \begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix}\begin{pmatrix} X_s^u \\ u_s \end{pmatrix} ds + \frac{1}{2}\left(X_T^u\right)^\top G X_T^u \middle| X_t^u = x\right].$$

for $t \in [0,T]$ and $x \in \mathbb{R}^n$. It can be shown that the value function has representation $v(t,x) = \frac{1}{2}x^\top P_t x$ for some symmetric matrix valued process P satisfying the Ricatti equation for this problem (Sun et al., 2016, Theorem 2.4.3). In particular, for any $t \in [0,T]$ we have

$$a_t^* x = Z_t^{t,x,\phi^*} = \partial_x v(t,x) = P_t x,$$

hence $a_t^* = P_t$ is also symmetric. We show that this Ricatti equation is recovered by the steady state solution of (3.53). To see this, note that the terms $Y_t^{\phi^*}, Z_t^{\phi^*}$ are determined by performing the same analysis as in (3.58) and (3.53), but using α^* and β^*. This yields, for $(t,x) \in [0,T] \times \mathbb{R}^n$,

$$\begin{aligned} Y_t^{t,x,\phi^*} &= a_t^* x + b_t^*, \\ Z_t^{t,x,\phi^*} &= a_t^* \left((C_t + D_t \alpha_t^*) x + D_t \beta_t^* \right), \end{aligned}$$

where the processes $a^* \in \mathcal{C}^1([0,T]; \mathbb{R}^{n\times n})$, $b^* \in \mathcal{C}^1([0,T]; \mathbb{R}^n)$ solve the ODEs, for $t \in [0,T]$

$$\begin{aligned} \dot{a}_t^* + a_t^* (A_t + B_t \alpha_t^*) + A_t^\top a_t^* + C_t^\top a_t^* (C_t + D_t \alpha_t^*) + \left(Q_t + S_t^\top \alpha_t^* \right) = 0, \\ \dot{b}_t^* + A_t^\top b_t^* + \left(a_t^* B_t + C_t^\top a_t^* D_t + S_t^\top \right) \beta_t^* = 0, \end{aligned}$$

with $a_T^* = G, b_T^* = 0$. Note that, by Theorem 3.5.6, we have $\beta_t^* = 0$. By inspection, we see that $b_t^* = 0$ is then the solution to the ODE determining b. The term α^* is determined by the optimality condition

$$B_t^\top Y_t^{\phi^*} + D_t^\top Z_t^{\phi^*} + R_t \phi_t^* \left(X_t^{\phi^*} \right) + S_t X_t^{\phi^*} = 0, \qquad 0 \le t \le T.$$

Substituting the definitions of ϕ_t^*, $Y_t^{\phi^*} = Y_t^{t,X_t^{\phi^*},\phi^*}$ and $Z_t^{\phi^*} = Z_t^{t,X_t^{\phi^*},\phi^*}$ yields

$$0 = B_t^\top a_t^* X_t^{\phi^*} + D_t^\top a_t^* \left((C_t + D_t \alpha_t^*) X_t^{\phi^*} \right) + R_t \alpha_t^* X_t^{\phi^*} + S_t X_t^{\phi^*}, \qquad 0 \le t \le T.$$

We rearrange this to find, assuming that for all $t \in [0,T]$, the matrix $\tilde{K}_t$ is invertible for brevity (this assumption can be removed by introducing Moore-Penrose pseudoinverses)

$$\alpha_t^* = -\tilde{K}_t^{-1} \tilde{L}_t, \qquad 0 \le t \le T,$$

where

$$\begin{aligned}\tilde{K}_t &= D_t^\top a_t^* D_t + R_t,\\ \tilde{L}_t &= B_t^\top a_t^* + D_t^\top a_t^* C_t + S_t.\end{aligned}$$

Substituting this into the ODE for a^* yields, for $t \in [0,T]$,

$$\begin{aligned}\dot{a}_t^* + a_t^* \left(A_t - B_t \tilde{K}_t^{-1} \tilde{L}_t\right) + A_t^\top a_t^* + C_t^\top a_t^* \left(C_t - D_t \tilde{K}_t^{-1} \tilde{L}_t\right) + \left(Q_t - S_t^\top \tilde{K}_t^{-1} \tilde{L}_t\right) &= 0\\ a_T^* &= G.\end{aligned}$$

Since a^* is symmetric, factoring out $\tilde{L}_t$ yields, for $t \in [0,T]$,

$$\begin{aligned}\dot{a}_t^* + a_t^* A_t + A_t^\top a_t^* + C_t^\top a_t^* C_t + Q_t - \tilde{L}_t^\top \tilde{K}_t^{-1} \tilde{L}_t &= 0,\\ a_T^* &= G.\end{aligned} \tag{3.99}$$

This is exactly the Ricatti equation for this problem.

Chapter 4

Numerical Examples

In this chapter we present numerical results for the problem studied in the previous chapters. We implement the three algorithms derived in Chapter 2 and apply to the Utility maximisation problem with various utilities and the investment consumption problem with randomly terminating income with power utility. For most of our examples we provide benchmark solutions with which to compare our numerical results. We also consider examples with non-Markovian structure for which analytical solutions have not been established.

We additionally apply our machine learning methods to the LQ problem presented in Chapter 3. We show equivalences between the DC2BSDE method and the PPGM method that we give convergence analysis for. Furthermore, in the unconstrained case we use the ODE representations of the adjoint processes to construct a simpler numerical algorithm, and we compare the results of this algorithm to the theoretically established convergence rates.

The chapter is organised as follows. Section 4.1 presents numeric results for our three algorithms for the Utility Maximisation problem, and compares to existing methods in the literature. Section 4.2 gives numerical results for the investment-consumption problem. Section 4.3 presents numerical results for implementations of the PPGM method for the LQ problem.

4.1 Utility Maximisation

In this section we apply the primal and dual DC2BSDE and DSMP algorithms to solve utility maximisation. To show the effectiveness of the algorithm we apply it to a series of utility problems, with certain constraints on the control space.

Example 4.1.1 (Unconstrained Non-HARA Utility Problem)**.** We start with the unconstrained case $K = \mathbb{R}^m$, corresponding to a complete market. We consider the utility problem where U has the following non-HARA form

$$U(x) = \frac{1}{3}H(x)^{-3} + H(x)^{-1} + xH(x), \qquad x > 0, \tag{4.1}$$

where $H(x) = \sqrt{2}\left(-1+\sqrt{1+4x}\right)^{-\frac{1}{2}}$. The dual utility is given by $\tilde{U}(\tilde{x}) = \frac{1}{3}\tilde{x}^{-3} + \tilde{x}^{-1}$. We have the dual constraint space $\tilde{K} = \{\tilde{\pi} \in \mathbb{R}^m : \delta_K(\tilde{\pi}) < \infty\} = \{0\}$, so the optimal dual control is $\tilde{\pi}^* = 0$. Suppose that $r_s = r$, $\Sigma_s = \Sigma$, $\mu_s = \mu$ and $\theta_s = \theta$, $s \in [0, T]$, are constant processes. It is shown in Li and Zheng (2018) that the optimiser $\tilde{x}^*$ and processes $\tilde{X}^{\tilde{x}^*,\tilde{\pi}^*}$ and $\tilde{Y}^{\tilde{x}^*,\tilde{\pi}^*}$ are given by

$$\begin{aligned}
\tilde{x}^* &= \frac{1}{\sqrt{2x_0}}\left[e^{\left(r+|\theta|^2\right)T} + \sqrt{e^{2(r+|\theta|^2)T} + 4x_0e^{3(r+2|\theta|^2)T}}\right]^{\frac{1}{2}} \\
\tilde{X}_t^{\tilde{x}^*,\tilde{\pi}^*} &= \tilde{x}^* \exp\left(-\left(r+\frac{1}{2}|\theta|^2\right)t - \theta^\top W_t\right) \\
\tilde{Y}_t^{\tilde{x}^*,\tilde{\pi}^*} &= a_1S_t^1 + a_2S_t^2,
\end{aligned}$$

where $a_1 = -(\tilde{x}^*)^{-4}e^{3\left(r+2|\theta|^2\right)T}$, $a_2 = -(\tilde{x}^*)^{-2}e^{\left(r+|\theta|^2\right)T}$, $S_t^1 = e^{\left(r-4|\theta|^2\right)t+2\theta^\top W_t}$ and $S_t^2 = e^{rt+2\theta^\top W_t}$ for $t \in [0, T]$. Furthermore, the dual value function is given by

$$\tilde{u}(t, \tilde{x}) = \frac{1}{3}\tilde{x}^{-3}e^{\left(3r+6|\theta|^2\right)(T-t)} + \tilde{x}^{-1}e^{\left(r+|\theta|^2\right)(T-t)}, \qquad (t, \tilde{x}) \in [0, T] \times \mathbb{R}^n.$$

From these analytical formulae, the primal processes can be derived using Theorem 2.3.2. For our application, we take $m = 5$, $T = 0.5$, $x_0 = 1$, $r = 0.05$, $\mu = 0.06\mathbb{1}_5$ and set Σ as a random matrix of positive numbers between 0 and 0.2 to introduce correlation of the underlying Brownian motions, with 0.2 added to the diagonal to ensure invertibility.

Figure 4.1 shows the development of the loss functions for the primal and dual algorithms with 50 time steps. For the control, we want to maximise the Hamiltonian, which means the derivative of the loss function should go to 0. Hence, we plot the function $j \mapsto \sum_{i=0}^{49} \sum_{\theta \in \theta_i^j} |\partial_\theta \mathcal{L}_2 \left(\theta_i^j, i\right)|$ using the notation of (2.6), where θ_i^j denotes the parameter set for the neural network forming π_i, at iteration step j of the algorithm. For the dual graph we plot the derivative of $\tilde{\mathcal{L}}_3$ in (2.18) with respect to $\tilde{x}$, since the function $\tilde{\mathcal{L}}_2$ in (2.17) is always zero as the dual control is forced to be zero, hence optimal, by the support function δ_K. Over the 10000 iteration steps both losses become small and settled, and therefore are deemed to have converged. In this method we choose an initial learning rate of 10^{-2} for the BSDE parameter updates and the start control $\tilde{x}$ in the dual algorithm, and 10^{-3} for the control part. Over the iterations, we divide these rates by 10 three times at regular intervals. We repeat this choice of learning rates for all applications in the sequel, though the number of iteration steps may change.

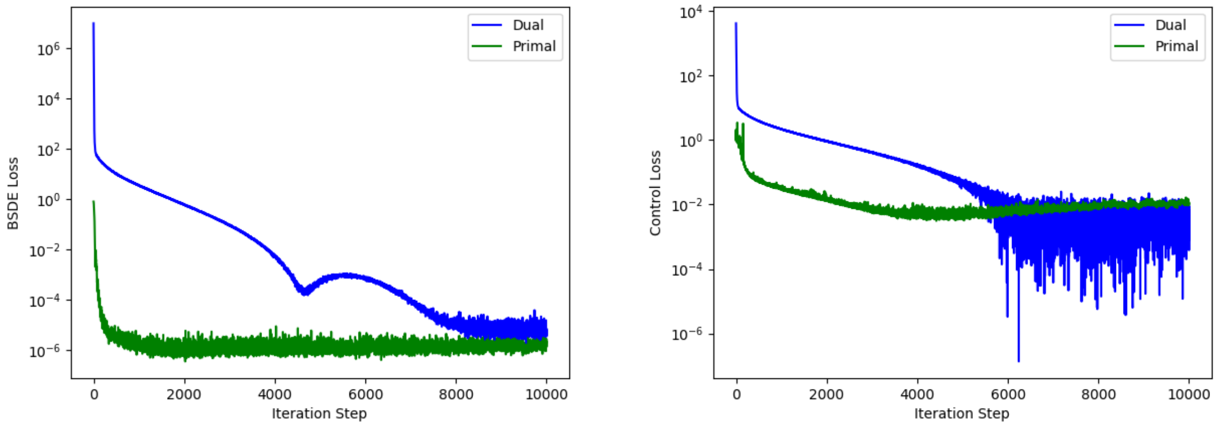
Figure 4.2 shows two simulations of the processes V^π and Y^π. The 'primal' processes are directly outputted by the primal algorithm, and the 'dual' process are given by the following relations for $t \in [0, T]$, using the outputted processes of the dual algorithm

$$
\begin{aligned}
X_t^\pi &= -\tilde{Y}_t^{\tilde{x},\tilde{\pi}}, \\
Y_t^\pi &= \tilde{X}_t^{\tilde{x},\tilde{\pi}}, \\
V_t^\pi &= \tilde{V}_t^{\tilde{x},\tilde{\pi}} - \tilde{X}_t^{\tilde{x},\tilde{\pi}} \tilde{Y}_t^{\tilde{x},\tilde{\pi}},
\end{aligned}
$$

using $N = 20$ time steps. We plot these, along with the explicit solution for reference. Both primal and dual approximations are very close to the solutions. The dual state process, appearing in the Y^π graph (but very close to the solution process), is closer than the corresponding primal process since there is no error for the dynamic dual control, only the dual start variable $\tilde{x}$.

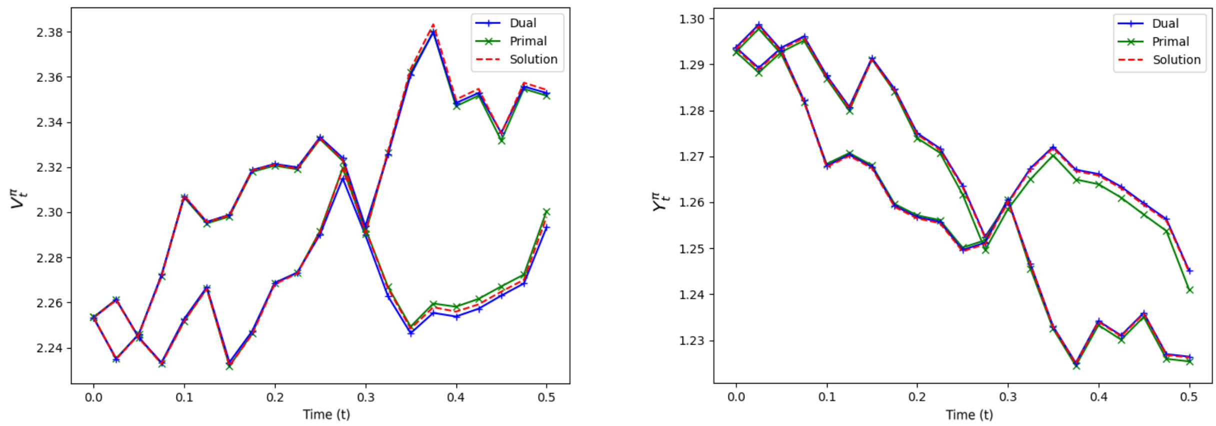
Figure 4.3 shows the evolution of the approximation of the initial value $v\left(0, x_0\right)$ against number of time steps for the primal and dual algorithms, where $x_0 = 1$. The dual value is higher than the primal value, as expected due to the duality relation (2.12), and the gap between these shrinks as the number of time steps increases. The final relative approximation error is $7 \times 10^{-4}\%$ for the primal problem, and $3 \times 10^{-4}\%$ for the dual problem, showing high accuracy of the proposed scheme at time 0.

The dual control loss appears 'noisier' than the primal counterpart. This is due to the fact that the dual control optimisation involves the initial state variable, which can be above or below the target variable, and the dual control process is set to 0. The derivative of the loss (2.18) repeatedly gets close to 0 as the value of the loss oscillates above and below the optimal value. This is not the case for the primal loss function, as the approximate Hamiltonian (2.6) is always a lower bound of the true Hamiltonian (up to numerical errors). Furthermore, the learning rate is higher for the dual initial state variable, potentially leading to higher volatility of the control approximation.



(a) Loss function of the BSDE. (b) Derivative of the control loss function.

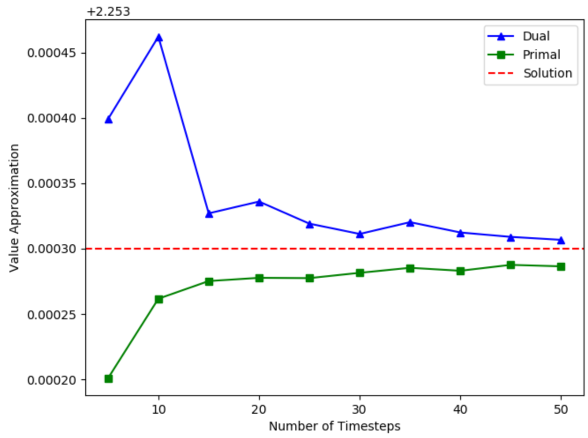
Figure 4.1: Loss functions against iteration step for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem.



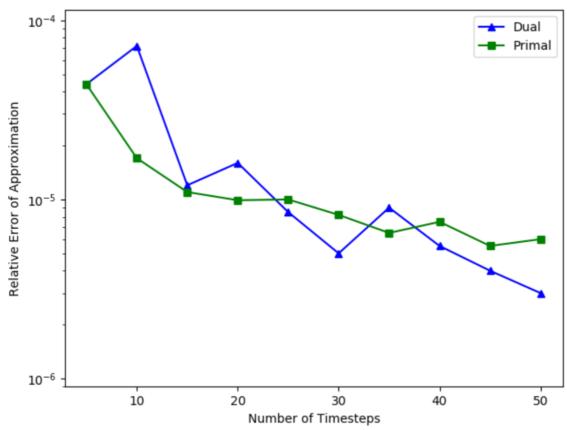
(a) Simulation of the process V^π. (b) Simulation of the process Y^π.

Figure 4.2: Two simulations of the value process and value-derivative process for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem. The displayed dual processes are those implied by the duality relations (2.21).

Example 4.1.2 (Cone-Constrained Merton Problem)**.** Now we consider the problem with power utility $U(x) = \frac{1}{p}x^p$ for some $0 < p < 1$ when the control space is the positive cone



(a) Value approximation.



(b) Relative error of value approximation.

Figure 4.3: Approximation and relative error of the value function against number of time steps N for the primal and dual deep controlled 2BSDE methods applied to the unconstrained non-HARA utility problem. Note in the left graph that the Y values on the axis are shifted down by 2.253.

$K = [0, \infty)^m$, corresponding to no short selling constraints. For this problem, the dual HJB equation tells us that the optimal dual control is given by

$$\tilde{\pi}_t^* = \arg\min_{\pi \in \tilde{K}} \left|\theta_t + \Sigma_t^{-1}\pi\right|^2, \qquad 0 \le t \le T. \tag{4.2}$$

The dual optimal state process is a geometric Brownian motion given by

$$\tilde{X}_t^{\tilde{x}^*, \tilde{\pi}^*} = \tilde{x}^* \exp\left(-\int_0^t \left(r_s + \frac{1}{2}|\theta_s^*|^2\right) ds - \int_0^t (\theta_s^*)^\top dW_s\right), \qquad 0 \le t \le T,$$

in terms of the optimal start point $\tilde{x}^* = \tilde{X}_0^{\tilde{x}^*, \tilde{\pi}^*}$ to be determined. The corresponding dual utility function is $\tilde{U}(\tilde{x}) = \frac{1-p}{p}\tilde{x}^{\frac{p}{p-1}}$. In (Li and Zheng, 2018, Subsection 4.1) the following solutions are found, for $t \in [0, T]$,

$$\begin{aligned} \tilde{x}^* &= x_0^{p-1} \exp\left((1-p)\int_0^T \left[\frac{p}{2(p-1)^2}|\theta_s^*|^2 - \frac{p}{p-1}r_s\right] ds\right) \\ \tilde{Y}_t^{\tilde{x}^*, \tilde{\pi}^*} &= x_0 \exp\left(\int_0^t \left[r_s + \frac{1-2p}{2(1-p)^2}|\theta_s^*|^2\right] ds + \frac{1}{1-p}\int_0^t (\theta_s^*)^\top dW_s\right), \end{aligned} \tag{4.3}$$

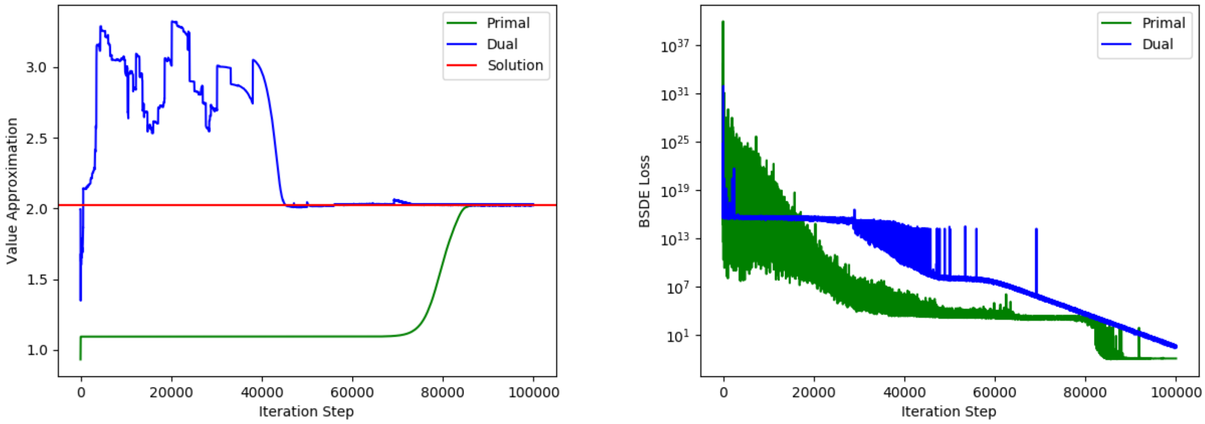
where $\theta_t^* = \theta_t + \Sigma_t^{-1}\tilde{\pi}_t^*$. This deterministic optimal dual control (4.2 can be found numerically using Python packages such as `Scipy`. The dual value function is given by

$$\tilde{v}(t,\tilde{x}) = \tilde{U}(\tilde{x}) \exp\left(\int_t^T \left[\frac{p}{2(p-1)^2}|\theta_s^*|^2 - \frac{p}{p-1} r_s\right] ds\right), \qquad (t,\tilde{x}) \in [0,T] \times \mathbb{R}. \tag{4.4}$$

For our implementation, we take $T = 0.5$, $x_0 = 1.0$, $m = 50$, $r_t = 0.05$ for $t \in [0,T]$ and $p = \frac{1}{2}$. We define the ith element of the drift μ at time $t \in [0,T]$ drift as $\mu^(i)_t = 0.04 + \sin(\pi t + A_i)/50$ where A_i is some arbitrarily chosen matrix for each $i = 1,\dots,50$. We choose this function so that in the unconstrained case it would be optimal to short sell at some points. We set Σ_t to be a constant matrix with 0.4 on the diagonal, and 0.2 everywhere else.

Figure 4.4 shows the results of applying the primal and dual 2BSDE methods to this problem. We use $N = 10$ time steps and run the algorithm for 100000 steps, notably more than for the lower dimensional unconstrained problem. This increases the run time to 5457 seconds. We start the BSDE and control learning rates at 0.01 and 0.001 respectively, then reduce these by a factor of 10 half way through training. We run both algorithms at the same time, which saves time as we can use the same generated Brownian motions for each iteration step for both algorithm. We see less convergent structure at the beginning of training, perhaps due to bad initial guesses for the neural networks. Indeed, we see that the value for the primal functions stays nearly constant until the loss function drops to a certain level, when the processes are accurate enough that the value approximation can move towards the analytical solution. This is likely due to the initialisation of variables, which clearly produce a large initial loss. However, we see convergent structure further along the training, suggesting robustness of our algorithm with respect to initialisation. The dual algorithm behaves similarly, but with more noise. At the end of training we have relative errors of 0.1% and 0.05% for the primal and dual algorithms respectively. This error is a few orders of magnitude higher than the unconstrained case due to the increase in dimension.

Table 4.1 shows the value approximations outputted when running the primal and dual algorithms with 50 time steps. For this problem, the exact solution can be computed as 2.02529. The table shows the upper and lower bound approximations for certain numbers of iterations of the algorithm, together with relative percentage error. In the dual algorithm, we use initial learning rates of 0.05 and 0.005 for the BSDE and control variables respectively. We manually choose the learning rate for the dual algorithm to be 5 times larger, possibly to account for



(a) Value approximation. (b) BSDE loss.

Figure 4.4: Approximation of the value function and bsde loss against iteration step for the primal and dual deep controlled 2BSDE methods applied to the cone constrained power utility problem.

the fact that the initial guess is further away. The inaccuracies of the initial guesses of the neural networks account for the dual value initially moving away from the true value. We then decrease all rates by a factor of 10 every 1000 steps. As the number of iterations increase, the duality gap becomes smaller, while in general still containing the true value. For comparison here, we also provide the run times of the respective algorithms. The dual algorithm is seen to take slightly longer than the primal problem. This is because there is an additional variable (the start variable $\tilde{x}$) to optimise.

Step	Primal Value	Rel-Err (%)	Time (s)	Dual Value	Rel-Err (%)	Time (s)
1	2.04799	1.12081E+00	483	2.13161	5.24978E+00	563
10	2.03847	6.51101E-01	498	3.22516	5.92447E+01	580
50	2.01880	3.20423E-01	505	1.91795	5.29974E+00	588
100	2.02469	2.94259E-02	514	2.10625	3.99787E+00	597
500	2.02518	5.42879E-03	582	2.02674	7.17313E-02	677
1000	2.02522	3.07794E-03	667	2.02545	8.18743E-03	786
2000	2.02524	2.13741E-03	837	2.02540	5.41970E-03	993
5000	2.02527	8.37157E-04	1348	2.02530	5.24745E-04	1575

Table 4.1: Value approximation and runtime at certain iteration steps for the primal and dual deep controlled 2BSDE methods applied to the cone-constrained power utility problem.

Example 4.1.3 (Ball Constrained Log Utility Problem)**.** In this example we consider the log utility problem when the control process is constrained to the ball $B(0,R)$ of radius $R>0$ in $\mathbb{R}^m$. This corresponds to the investor not wanting to invest more than a certain proportion of their wealth in risky stocks. In this case, the support function $\delta_K(z)=R|z|$ for $z\in\mathbb{R}^m$ does not vanish for all admissible dual controls. Therefore, the SDE (2.44) is harder to solve. In general this problem is also difficult to solve analytically because of the dependence of the dual controls $\tilde{x}^*$ and $\tilde{\pi}^*$ on each other. However, we can exploit the multiplicity property of the log function to decouple these controls to give analytical solutions, which we describe here. To this end let $U(x)=\log(x)$ be our utility function, for which the dual utility is $\tilde{U}(y)=-(1+\log(y))$. Since the dual state process $\tilde{X}^{\tilde{x},\tilde{\pi}}$ is a geometric Brownian motion for any controls $(\tilde{x},\tilde{\pi})$, we have

$$\mathbb{E}\left[\tilde{U}\left(\tilde{X}_T^{\tilde{x},\tilde{\pi}}\right)\right]=-1-\log(\tilde{x})+\int_0^T r_t+R|\tilde{\pi}_t|+\frac{1}{2}|\theta_t+\Sigma_t^{-1}\tilde{\pi}_t|^2dt.$$

The measurable process

$$\tilde{\pi}_t^*:=\arg\min_{\tilde{\pi}\in\mathbb{R}^m}\left\{R|\tilde{\pi}|+\frac{1}{2}|\theta_t+\Sigma_t^{-1}\tilde{\pi}|^2\right\},\qquad 0\le t\le T,$$

is an optimal control, and can be found using existing deterministic convex optimisation algorithms. Independently of this, the dual control $\tilde{x}^*$ is found via

$$\begin{aligned}\tilde{x}^*&=\arg\min_{\tilde{x}>0}\left\{x_0\tilde{x}-\log(\tilde{x})+\int_0^T r_t+\delta_K(\tilde{\pi}_t)+\frac{1}{2}|\theta_t+\Sigma_t^{-1}\tilde{\pi}_t|^2dt\right\}\\&=\arg\min_{\tilde{x}>0}\{x_0\tilde{x}-\log(\tilde{x})\}\\&=\frac{1}{x_0}.\end{aligned}$$

The value function is then given by

$$\begin{aligned}v(0,x_0)&=\mathbb{E}\left[\tilde{U}\left(\tilde{X}_T^{\tilde{x}^*,\tilde{\pi}^*}\right)\right]+\tilde{x}^*x_0 &(4.5)\\&=\log(x_0)+\int_0^T r_t+R|\tilde{\pi}_t|+\frac{1}{2}|\theta_t+\Sigma_t^{-1}\tilde{\pi}_t|^2dt. &(4.6)\end{aligned}$$

For our application, we take $R=1.0$, $T=0.5$, $x_0=5.0$, $m=20$, $r_t=0.05$ and $\mu_t=0.07\mathbb{1}_5$ for $t\in[0,T]$. We define the volatility as a diagonal matrix with the ith diagonal element $\Sigma_t^{(i)}$ equal to

$$\Sigma_t^{(i)}=0.4+0.2\sin(2\pi t+A_i),\qquad 0\le t\le T,$$

for some arbitrarily chosen numbers A_i to shift the peaks of the sin curve for each dimension. We choose this function so that in the unconstrained case it would be reasonable to invest more than we are allowed at points when the volatility is low. As $\Sigma_t^{(i)} > 0$ for all $t \in [0, T]$, the volatility is non-degenerate.

Table 4.2 shows the evolution of the approximated value function for the primal and dual algorithms, compared with the numerical solution 1.64755570 found using (4.6). The algorithm is run using $N = 10$ time steps, and the value approximation, relative error and (combined) runtime are recorded. We again perform both algorithm simultaneously using the same Brownian motions simulations. The value approximation initially moves away from the solution, but then converges. For the primal algorithm, the value moves away over the first 1000 steps, then goes to the solution. For the dual algorithm, the value does not move as far from the solution, but takes longer to do so, only starting moving towards the solution at 10000 steps. The dual value initially dips below the solution, then increases, which is why the relative error is lower at 1000 steps than at the start.

Step	Primal Value	Rel-Err (%)	Dual Value	Rel-Err (%)	Time (s)
1	0.99277	3.97426E+01	2.50412	5.19898E+01	128
1000	0.04637	9.71853E+01	1.57275	4.54015E+00	183
5000	0.22780	8.61735E+01	1.80325	9.44972E+00	405
10000	1.39359	1.54147E+01	2.01337	2.22035E+01	701
12000	1.62170	1.56918E+00	1.98620	2.05544E+01	815
18000	1.63545	7.34658E-01	1.68103	2.03169E+00	1149
20000	1.63543	7.35726E-01	1.65361	3.67465E-01	1259

Table 4.2: Value approximation and runtime at certain iteration steps for the primal and dual deep controlled 2BSDE methods applied to the ball-constrained log utility problem.

4.1.1 Methodology Comparison

Now we compare the performance of our algorithm to two existing in the literature. We first compare to the algorithm of Han and E (2016) where the primal gains function is optimised directly. This is not explicitly given a name by the authors, so we refer to it as the PVO (Primal Value Optimisation) algorithm. Secondly we compare to the Hybrid-Now algorithm of

Huré et al. (2021) (referred to in the sequel as Hybrid), where the control is optimised using an approximation of the value function, which is also found using neural networks. First, we note that while both algorithms search for a Markovian control, and as such solve Markovian control problems, they are potentially less restrictive than our algorithm. The maximum requirement for deriving these algorithms is satisfaction of the dynamic programming principle, as opposed to our derivation of the algorithm relying on the HJB equation. However it may be possible that our algorithm works in such a general setting as well (and resulting approximate solutions to the Hamiltonian PDE become viscosity solutions). The second algorithm is formalised in terms of successive approximations of the value function working backwards from terminal time in a discretisation. This method requires suitable knowledge of the distribution of the optimal state process at each time step, which we do not know a priori. Instead, we implement the algorithm in a similar vein to our own, generating sample paths of the state and control process and solving each optimisation problem (Bachouch et al., 2021, (2.1)) simultaneously. We run each algorithm for a fixed number of steps of gradient descent. This may not be the best way to compare the algorithms since they may converge after a different number of steps, but in practise it is difficult to identify exactly when this happens. We only compare to the primal value approximations of our DC2BSDE and DSMP methods, though it would of course be possible to use PVO and Hybrid to solve the dual problem.

Example 4.1.4 (Power Utility in Black Scholes Model)**.** Consider the utility maximisation problem as of Example 4.1.1, but we take the power ($p = \frac{1}{2}$) utility $U(x) = 2\sqrt{x}$. We fix the terminal time $T = 0.5$ but change the stock market dimension m and the number of time steps N. The other parameters remain the same as in Example 4.1.1. Table 4.3 shows the results of this comparison. The solutions can be found (as is found in Example 4.1.2 but with no constraints) to be 2.04864 and 2.09756 for dimensions 5 and 20 respectively. For our methods we ran the DC2BSDE with $m = 20$ for 10000 steps, and 5000 steps for the $m = 5$ case and the DSMP.

The comparison to the PVO algorithm indicates that our method sacrifices slight accuracy, but scales much better with number of time steps and dimension. Indeed, when $m = 20$ the lower accuracy can be compensated by doubling the number of time steps without reaching the runtime of the PVO algorithm. We ran the PVO algorithm with a learning rate of 0.01 for 10000 and 20000 steps for $m = 5, 20$ respectively. The results for the Hybrid algorithm originate from our own implementation of the method. There may be a number of differences from the original implementation which could lead to less accurate results than expected. For example, we are not sure how exactly the authors used the 'explore first, exploit later' technique (Huré et al., 2021, Section 3.3) to generate sample paths of the optimal state process without first determining a control. In Bachouch et al. (2021), a 100 dimensional PDE is solved with an accuracy of order 0.1% and a runtime of 2 - 4 hours, using $N = 40$ time steps and a terminal time of $T = 1.0$. This is much better than our preliminary results for this algorithm. However, we did not see any convergence like this in our implementation, in any time frame that was comparable to our methods. The hybrid algorithm is the fastest per iteration step, however we ran the algorithm for 50000 steps with a learning rate of 0.0001 to avoid instability issues, which is the cause of the large runtime.

m	N	DC2BSDE Value	Rel-Err (%)	Time (s)	DSMP Value	Rel-Err (%)	Time (s)
5	5	2.0483	1.4829E-02	54	2.0465	1.0476E-01	59
5	10	2.0485	9.2087E-03	128	2.0505	9.0012E-02	148
5	20	2.0486	4.4761E-03	309	2.0476	5.3657E-02	358
20	5	2.0942	1.6192E-01	92	2.0906	3.3063E-01	66
20	10	2.0958	8.2069E-02	223	2.0947	1.3682E-01	167
20	20	2.0961	6.5495E-02	538	2.0963	5.8714E-02	412
m	N	PVO Value	Rel-Err (%)	Time (s)	Hybrid Value	Rel-Err (%)	Time (s)
5	5	2.0497	4.9400E-02	79	2.0441	2.2415E-01	328
5	10	2.0478	4.3425E-02	266	2.0457	1.4347E-01	479
5	20	2.0483	1.8711E-02	950	2.0467	9.5190E-02	1630
20	5	2.0952	1.1184E-01	233	2.0431	2.5977E+00	343
20	10	2.0963	5.9899E-02	813	2.0536	2.0940E+00	813
20	20	2.0969	3.0629E-02	1824	2.0604	1.7720E+00	1747

Table 4.3: Comparison of the Primal DC2BSDE and (primal value approximation of the) DSMP algorithm to the PVO and Hybrid algorithms for the unconstrained Power Utility problem.

The example we consider is a toy example with an analytical solution. In particular, the supremum term inside the HJB equation (1.7) can be written exactly and for this problem we are left with the nonlinear PDE

$$\frac{\partial}{\partial t}v\left(t,x\right)=\frac{1}{2}|\theta_t|^2\frac{\partial_x v\left(t,x\right)^2}{\partial_{xx}^2 v\left(t,x\right)}-rx\partial_x v\left(t,x\right),\qquad (t,x)\in[0,T)\times\mathbb{R},$$

and $v(t,x)=U(x)$ for $x\in\mathbb{R}$. This PDE is control independent. In particular it is independent of the control dimension m apart from the negligible time taken to calculate θ. The PDE can be solved directly using the Deep 2BSDE method introduced in Beck et al. (2019b). This is also one of the algorithms described in Ji et al. (2022). The results of solving this for $m=20$ are given in Table 4.4. Due to the second derivative in the denominator, this PDE is unstable. In the algorithm we require 50000 iteration steps to see convergence results. Therefore, while this algorithm is faster per iteration step, it is slower overall. However its runtime would be comparable to our algorithm for larger value of m. It is also an order of magnitude less accurate, and can only be used in the special case where the supremum term can be written out as an explicit function of t and x.

The primal HJB equation, even in this simple case, can be very difficult to solve numerically. This further emphasises the utility of the algorithms presented in this paper, solving the equation by either finding the control without nonlinear dependence on the value function derivatives, or by solving the dual problem.

An implementation of each algorithm can be found in `https://github.com/Ashley-Davey/ML-For-Stochastic-Control`.

Remark 16. Another adaptation for this algorithm suggested in Ji et al. (2022) is to perform the maximisation over the control variate explicitly at each iteration step, rather than just performing one step of gradient descent. There may be difficulties in implementing this method without an analytical maximiser to plug in, in terms of time to run, dimensionality of the problem, and inaccuracies of the approximate value function derivatives that would need to be used. An additional approximation step may also be needed, as this maximisation would be pathwise yet we need the entire control function to simulate the state process in the next iteration step. We leave the effectiveness of this method to further research.

N	DBSDE Value	Rel-Err(%)	Time (s)	DC2BSDE Value	Rel-Err(%)	Time (s)
5	1.96570E+00	6.28613E+00	363	2.0942	1.6192E-01	92
10	2.05429E+00	2.06281E+00	865	2.0958	8.2069E-02	223
20	2.06381E+00	1.60905E+00	1980	2.0961	6.5495E-02	538

Table 4.4: Value, error and runtime of the Deep 2BSDE method and primal DC2BSDE method for the unconstrained Power Utility problem.

4.1.2 Quadratic Utility

The utility functions we have seen up to now do not satisfy global Lipschitz conditions that are often required for convergence analysis. Hence we consider a globally Lipschitz utility function that is a negative quadratic around 0 up to some point before its peak, where it becomes constant. Additionally, we will see that the optimal state process is bounded. This aligns with a sufficient condition in Huré et al. (2021) that the state process should be bounded for any choice of control.

Take $c < 0$, $H > 0$ and $0 < \alpha \leq 1$. Then define $U\colon [0, \infty) \to \mathbb{R}$ by

$$U(x) = \begin{cases} \frac{1}{2}cx^2 - cHx, & 0 \leq x \leq \alpha H, \\ c\alpha H^2 \left(\frac{\alpha}{2} - 1\right), & \alpha H \leq x, \end{cases} \tag{4.7}$$

for $x \geq 0$. Note that this will always be a negative quadratic, hence concave, as we take $c < 0$ to be strictly negative. Figure 4.5 shows some utilities for varying values of α and H. The parameter c is just a scaling parameter and has been fixed at -1.0 here. The parameter H indicates where along the x axis the peak should be, and α indicates at what point (as a proportion of H) the utility becomes constant. In order to ensure concavity of the utility function, we must have $\alpha \leq 1$ and $H \geq 0$.

The parameter α indicates how early we transition to the constant term. Note that the function U is smooth when we have $\alpha = 1.0$. We first find the dual utility, the fenchel-legendre transform

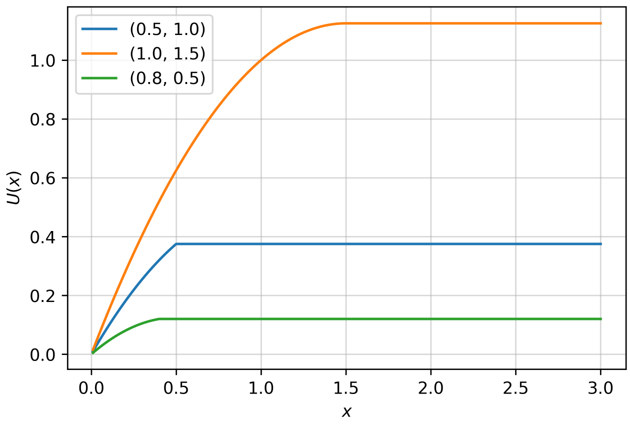


Figure 4.5: The quadratic utility U for varying (α, H).

of U given by

$$\tilde{U}(\tilde{x}) = \sup_{x \geq 0} \{U(x) - x\tilde{x}\} \tag{4.8}$$

$$= \max\left\{\sup_{0 \leq x \leq \alpha H} \{U(x) - x\tilde{x}\}, \sup_{\alpha H \leq x} \{U(x) - x\tilde{x}\}\right\} \tag{4.9}$$

$$= \max\left\{\sup_{0 \leq x \leq \alpha H} \{U(x) - x\tilde{x}\}, c\alpha H^2\left(\frac{\alpha}{2} - 1\right) - \alpha H\tilde{x}\right\} \tag{4.10}$$

$$= \max\left\{\sup_{0 \leq x \leq \alpha H} \{U(x) - x\tilde{x}\}, \alpha H\left(cH\left(\frac{\alpha}{2} - 1\right) - \tilde{x}\right)\right\} \tag{4.11}$$

Define $f\colon [0, \alpha H] \to \mathbb{R}$ by $f(x) = U(x) - x\tilde{x}$. We then have $f'(x) = cx - cH - \tilde{x}$ giving a potential optimiser at $x = H + \frac{\tilde{x}}{c}$. However it may not be in the range of f. Case 1: $x \geq \alpha H$, or equivalently $0 \leq \tilde{x} \leq cH(\alpha - 1)$. In this case optimiser is at $\hat{x} = \alpha H$, and both sides of the max are equal. Case 2: $x \leq 0$, or $\tilde{x} \geq -cH$. Then the optimiser is $\hat{x} = 0$ and $f(\hat{x}) = 0 \geq H\left(cH\left(\frac{\alpha}{2} - 1\right) - \tilde{x}\right)$. Case 3: $x \in [0, \alpha H]$, then $\hat{x} = x$ and

$$f(\hat{x}) = f\left(H + \frac{\tilde{x}}{c}\right) = \frac{-1}{2c}(cH + \tilde{x})^2.$$

Plugging these into (4.11) yields

$$\tilde{U}(\tilde{x}) = \begin{cases} \alpha H\left(cH\left(\frac{\alpha}{2} - 1\right) - \tilde{x}\right), & \tilde{x} \leq cH(\alpha - 1), \\ \frac{-1}{2c}(cH + \tilde{x})^2, & cH(\alpha - 1) \leq \tilde{x} \leq -cH, \\ 0, & -cH \leq \tilde{x}, \end{cases} \tag{4.12}$$

Now we have the dual utility, we can calculate the dual value function

$$\tilde{v}(t, \tilde{x}) = \mathbb{E}\left[\tilde{U}\left(\tilde{X}_T^{t,\tilde{x}}\right)\middle|\tilde{X}_t^{t,\tilde{x}} = \tilde{x}\right], \tag{4.13}$$

where

$$d\tilde{X}_s^{t,\tilde{x}} = -\tilde{X}_s^{t,\tilde{x}}(r ds + \theta dW_s), \qquad t \le s \le T. \tag{4.14}$$

Fix $t \in [0, T]$, $\tilde{x} \geq 0$ and define $\tau = T - t$. Since the process $\tilde{X}^{t,\tilde{x}}$ is a geometric Brownian Motion, we can write, given $\tilde{X}_t^{t,\tilde{x}} = \tilde{x}$,

$$\begin{aligned}
\tilde{X}_T^{t,\tilde{x}} &= \tilde{x}\exp\left(-\left(r + \frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}Z\right), \\
\left(\tilde{X}_T^{t,\tilde{x}}\right)^2 &= \tilde{x}^2\exp\left(-\left(2r + \theta^2\right)\tau - 2\theta\sqrt{\tau}Z\right),
\end{aligned}$$

almost surely, where Z is a standard normal random variable, with density $\phi(z) = \frac{1}{\sqrt{2\pi}}e^{\frac{-z^2}{2}}$, $z \in \mathbb{R}$, and CDF $\Phi(x) = \int_{-\infty}^{x}\phi(z)\,dz$, $x \in \mathbb{R}$. We then note that $\tilde{X}_T^{t,\tilde{x}} \leq cH(\alpha - 1)$ if and only if

$$k(\alpha) := \frac{\log(\tilde{x}) - \log(cH(\alpha - 1)) - \left(r + \frac{\theta^2}{2}\right)\tau}{\theta\sqrt{\tau}} \leq Z \tag{4.15}$$

and $\tilde{X}_T^{t,\tilde{x}} \leq -cH$ if and only if $k(0) \leq Z$. We then write

$$\begin{aligned}
\tilde{v}(t, \tilde{x}) &= \int_{-\infty}^{\infty}\tilde{U}\left(\tilde{x}\exp\left(-\left(r + \frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)d\mathbb{P}_Z(z) \\
&= \int_{-\infty}^{\infty}\tilde{U}\left(\tilde{x}\exp\left(-\left(r + \frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)\left(\mathbb{1}_{z\geq k(\alpha)} + \mathbb{1}_{k(0)\leq z\leq k(\alpha)} + \mathbb{1}_{z\leq k(0)}\right)\frac{e^{-\frac{z^2}{2}}}{2}dz \\
&=: I_1 + I_2 + I_3,
\end{aligned}$$

where

$$I_1 = \int_{-\infty}^{\infty} \tilde{U}\left(\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)\mathbb{1}_{z\geq k(\alpha)}\frac{e^{-\frac{z^2}{2}}}{2}dz,$$
$$I_2 = \int_{-\infty}^{\infty} \tilde{U}\left(\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)\mathbb{1}_{k(0)\leq z\leq k(\alpha)}\frac{e^{-\frac{z^2}{2}}}{2}dz,$$
$$I_3 = \int_{-\infty}^{\infty} \tilde{U}\left(\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)\mathbb{1}_{z\leq k(0)}\frac{e^{-\frac{z^2}{2}}}{2}dz.$$

We know that $I_3 = 0$, since $\tilde{U} = 0$ on the region of integration, so it remains to calculate I_1 and I_2. We complete the square to calculate I_1 as

$$\begin{aligned} I_1 &= \int_{k(\alpha)}^{\infty}\left(\frac{\alpha c H^2}{2}(\alpha-2) - \alpha H\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau - \theta\sqrt{\tau}z\right)\right)\frac{e^{-\frac{z^2}{2}}}{2}dz \\ &= \frac{\alpha c H^2}{2}(\alpha-2)\Phi(-k(\alpha)) - \alpha H\tilde{x}\int_{k(\alpha)}^{\infty} e^{\left(-r-\frac{\theta^2}{2}\right)\tau}e^{\frac{\theta^2}{2}\tau}\frac{e^{-\frac{\left(z+\theta\sqrt{\tau}\right)^2}{2}}}{2}dz \\ &= \frac{\alpha c H^2}{2}(\alpha-2)\Phi(-k(\alpha)) - \alpha H\tilde{x}e^{-r\tau}\Phi\left(-k(\alpha)-\theta\sqrt{\tau}\right). \end{aligned}$$

We then multiply out the brackets and complete the square twice to calculate I_2 as

$$\begin{aligned}
I_2 &= \int_{k(0)}^{k(\alpha)} \frac{-1}{2c}\left[\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau-\theta\sqrt{\tau}z\right)+cH\right]^2\frac{e^{-\frac{z^2}{2}}}{2}dz \\
&= \int_{k(0)}^{k(\alpha)} \frac{-cH^2}{2}\frac{e^{-\frac{z^2}{2}}}{2}dz \\
&- \int_{k(0)}^{k(\alpha)} H\tilde{x}\exp\left(-\left(r+\frac{\theta^2}{2}\right)\tau-\theta\sqrt{\tau}Z\right)\frac{e^{-\frac{z^2}{2}}}{2}dz \\
&- \int_{k(0)}^{k(\alpha)} \frac{1}{2c}\tilde{x}^2\exp\left(-\left(2r+\theta^2\right)\tau-2\theta\sqrt{\tau}Z\right)\frac{e^{-\frac{z^2}{2}}}{2}dz \\
&= \frac{-cH^2}{2}\int_{k(0)}^{k(\alpha)} \frac{e^{-\frac{z^2}{2}}}{2}dz \\
&- H\tilde{x}\int_{k(0)}^{k(\alpha)} e^{-\left(r+\frac{\theta^2}{2}\right)\tau}e^{\frac{\theta^2}{2}\tau}\frac{e^{-\frac{(z+\theta\sqrt{\tau})^2}{2}}}{2}dz \\
&- \frac{1}{2c}\tilde{x}^2\int_{k(0)}^{k(\alpha)} e^{-\left(2r+\theta^2\right)\tau}e^{2\theta^2\tau}\frac{e^{-\frac{(z+2\theta\sqrt{\tau})^2}{2}}}{2}dz \\
&= \frac{-cH^2}{2}\left[\Phi\left(k\left(\alpha\right)\right)-\Phi\left(k\left(0\right)\right)\right] \\
&- H\tilde{x}e^{-r\tau}\left[\Phi\left(k\left(\alpha\right)+\theta\sqrt{\tau}\right)-\Phi\left(k\left(0\right)+\theta\sqrt{\tau}\right)\right] \\
&- \frac{1}{2c}\tilde{x}^2e^{\left(-2r+\theta^2\right)\tau}\left[\Phi\left(k\left(\alpha\right)+2\theta\sqrt{\tau}\right)-\Phi\left(k\left(0\right)+2\theta\sqrt{\tau}\right)\right].
\end{aligned}$$

Note that we can write $\tilde{v}\left(t,\tilde{x}\right)=g_\alpha\left(t,\tilde{x}\right)-g_0\left(t,\tilde{x}\right)$ where

$$\begin{aligned}
g_\alpha\left(t,\tilde{x}\right) &= \frac{\alpha cH^2}{2}\left(\alpha-2\right)\Phi\left(-k\left(\alpha\right)\right) \\
&- \alpha H\tilde{x}e^{-r\tau}\Phi\left(-k\left(\alpha\right)-\theta\sqrt{\tau}\right) \\
&- \frac{cH^2}{2}\Phi\left(k\left(\alpha\right)\right) \\
&- H\tilde{x}e^{-r\tau}\Phi\left(k\left(\alpha\right)+\theta\sqrt{\tau}\right) \\
&- \frac{1}{2c}\tilde{x}^2e^{\left(-2r+\theta^2\right)\tau}\Phi\left(k\left(\alpha\right)+2\theta\sqrt{\tau}\right).
\end{aligned}$$

The primal value function is given by $u\left(t,x\right)=\inf_{\tilde{x}\geq 0}\{\tilde{v}\left(t,\tilde{x}\right)+xy\}$. The optimal wealth process is given by

$$X_t^{\pi^*}=-\frac{\partial\tilde{v}}{\partial\tilde{x}}\tilde{v}\left(t,\tilde{X}_t^{\tilde{x}^*}\right),$$

where we denote $\tilde{X}^{\tilde{x}^*} = \tilde{X}^{0,\tilde{x}^*}$ and

$$\tilde{x}^* = \arg\min_{\tilde{x}\geq 0}\{\tilde{v}\left(0,\tilde{x}\right) + x\tilde{x}\}.$$

To find the partial derivative of $\tilde{v}$, we first differentiate g_α. To this end, writing $k = k\left(\alpha\right)$, simple calculus gives us

$$\begin{aligned}
\frac{\partial}{\partial\tilde{x}} g_\alpha\left(t,\tilde{x}\right) &= -\frac{\partial k}{\partial\tilde{x}}\phi\left(-k\right)\frac{\alpha c H^2}{2}\left(\alpha-2\right) \\
&+ \frac{\partial k}{\partial\tilde{x}}\phi\left(-k-\theta\sqrt{\tau}\right)\alpha H\tilde{x}e^{-r\tau} \\
&- \Phi\left(-k-\theta\sqrt{\tau}\right)\alpha H e^{-r\tau} \\
&- \frac{\partial k}{\partial\tilde{x}}\phi\left(-k\right)\frac{cH^2}{2} \\
&- \frac{\partial k}{\partial\tilde{x}}\phi\left(k+2\theta\sqrt{\tau}\right)\frac{1}{2c}\tilde{x}^2 e^{\left(-2r+\theta^2\right)\tau} \\
&- \left(1-\Phi\left(-k-2\theta\sqrt{\tau}\right)\right)\frac{1}{c}\tilde{x}e^{\left(-2r+\theta^2\right)\tau} \\
&- \frac{\partial k}{\partial\tilde{x}}\phi\left(k+\theta\sqrt{\tau}\right)H\tilde{x}e^{-r\tau} \\
&- \left(1-\Phi\left(-k-\theta\sqrt{\tau}\right)\right)He^{-r\tau} \\
&= A + B\frac{\partial k}{\partial\tilde{x}}
\end{aligned}$$

for some $A, B \in \mathbb{R}$ depending on t and $\tilde{x}$ to be determined. To compute B, we will use the following relations, which can be derived from the definitions of $\phi\left(z\right) = \frac{1}{\sqrt{2\pi}}e^{\frac{-z^2}{2}}$ and $k = k\left(\alpha\right)$ in (4.15),

$$\begin{aligned}
\phi\left(-k-\theta\sqrt{\tau}\right) &= \phi\left(-k\right)\frac{e^{r\tau}cH\left(\alpha-1\right)}{\tilde{x}}, \\
\phi\left(-k-2\theta\sqrt{\tau}\right) &= \phi\left(-k\right)\frac{e^{\left(2r-\theta^2\right)\tau}c^2H^2\left(\alpha-1\right)^2}{\tilde{x}^2}.
\end{aligned} \tag{4.16}$$

We then have

$$
\begin{aligned}
B &= \phi\left(-k\right)\Bigg(-\frac{\alpha c H^2}{2}\left(\alpha-2\right)+\alpha H\tilde{x}e^{-r\tau}\frac{e^{r\tau}cH\left(\alpha-1\right)}{\tilde{x}}-\frac{cH^2}{2} \\
&\qquad -\frac{1}{2c}\tilde{x}^2 e^{\left(-2r+\theta^2\right)\tau}\frac{e^{\left(2r-\theta^2\right)\tau}c^2H^2\left(\alpha-1\right)^2}{\tilde{x}^2}-H\tilde{x}e^{-r\tau}\frac{e^{r\tau}cH\left(\alpha-1\right)}{\tilde{x}}\Bigg) \\
&= \phi\left(-k\right)cH^2\left(-\frac{1}{2}\alpha^2+\alpha+\alpha^2-\alpha-\frac{1}{2}-\frac{1}{2}\alpha^2+\alpha-\frac{1}{2}-\alpha+1\right) \\
&= 0.
\end{aligned}
$$

Hence

$$
\begin{aligned}
\frac{\partial}{\partial\tilde{x}}g_\alpha\left(t,\tilde{x}\right)=A=&-\alpha He^{-r\tau} \\
&-\Phi\left(k+\theta\sqrt{\tau}\right)\left(1-\alpha\right)He^{-r\tau} \\
&-\Phi\left(k+2\theta\sqrt{\tau}\right)\frac{1}{c}\tilde{x}e^{\left(-2r+\theta^2\right)\tau}.
\end{aligned}
$$

Taking $\alpha=0$ gives us the derivative of g_0, and subtraction yields

$$
\begin{aligned}
\frac{\partial}{\partial\tilde{x}}\tilde{v}\left(t,\tilde{x}\right) &= \frac{\partial}{\partial\tilde{x}}g_\alpha\left(t,\tilde{x}\right)-\frac{\partial}{\partial\tilde{x}}g_0\left(t,\tilde{x}\right) \\
&= -\alpha He^{-r\tau} \\
&-\left[\Phi\left(k\left(\alpha\right)+\theta\sqrt{\tau}\right)-\Phi\left(k\left(0\right)+\theta\sqrt{\tau}\right)\right]\left(1-\alpha\right)He^{-r\tau} \\
&-\left[\Phi\left(k\left(\alpha\right)+2\theta\sqrt{\tau}\right)-\Phi\left(k\left(0\right)+2\theta\sqrt{\tau}\right)\right]\frac{1}{c}\tilde{x}e^{\left(-2r+\theta^2\right)\tau}.
\end{aligned}
\tag{4.17}
$$

This is not an (analytically) invertible function of $\tilde{x}$, so we cannot use the relation $X_0^{\pi^*}=\frac{\partial}{\partial\tilde{x}}\tilde{v}\left(0,\tilde{X}_0^{\tilde{x}^*}\right)$ to find $\tilde{x}=\tilde{X}_0^{\tilde{x}^*}$, and hence the process $\tilde{X}^{\tilde{x}^*}$. However we can find it numerically. Given this, we have

$$
\begin{aligned}
X_t^{\pi^*}=-\frac{\partial}{\partial\tilde{x}}\tilde{v}\left(t,\tilde{X}_t^{\tilde{x}^*}\right) &= \alpha He^{-r\tau} \\
&+\left[\Phi\left(k\left(\alpha,\tilde{X}_t^{\tilde{x}^*}\right)+\theta\sqrt{\tau}\right)-\Phi\left(k\left(0,\tilde{X}_t^{\tilde{x}^*}\right)+\theta\sqrt{\tau}\right)\right]\left(1-\alpha\right)He^{-r\tau} \\
&+\left[\Phi\left(k\left(\alpha,\tilde{X}_t^{\tilde{x}^*}\right)+2\theta\sqrt{\tau}\right)-\Phi\left(k\left(0,\tilde{X}_t^{\tilde{x}^*}\right)+2\theta\sqrt{\tau}\right)\right]\frac{1}{c}\tilde{X}_t^{\tilde{x}^*}e^{\left(-2r+\theta^2\right)\tau}
\end{aligned}
$$

where we use the notation $k(\alpha,\tilde{x})$ here to remind us that k depends on $\tilde{x}$. We note that, since

$c<0$ and $\Phi\left(k\left(\cdot,\tilde{x}\right)\right)$ is increasing in α, we have

$$0\leq X_t^{\pi^*}\leq\alpha He^{-r\tau}+\left[\Phi\left(k\left(\alpha,\tilde{X}_t^{t,\tilde{x}}\right)+\theta\sqrt{\tau}\right)-\Phi\left(k\left(0,\tilde{X}_t^{\tilde{x}^*}\right)+\theta\sqrt{\tau}\right)\right]\left(1-\alpha\right)He^{-r\tau}\leq He^{-r\tau}$$

Even though the process $\tilde{X}^{\tilde{x}^*}$ is unbounded, the process X^{π^*} is bounded. We can see this intuitively from the fact that, even though $\tilde{X}^{\tilde{x}^*}$ can be arbitrarily large, the term $\Phi\left(k\left(\alpha,\tilde{X}_t^{\tilde{x}^*}\right)+\theta\sqrt{\tau}\right)-\Phi\left(k\left(0,\tilde{X}_t^{\tilde{x}^*}\right)+\theta\sqrt{\tau}\right)$ is of the form $\Phi\left(\log\left(\tilde{X}_t^{\tilde{x}^*}\right)+\epsilon\right)-\Phi\left(\log\left(\tilde{X}_t^{\tilde{x}^*}\right)\right)$, for some $\epsilon>0$ and becomes arbitrarily small for large $\tilde{X}^{\tilde{x}^*}$. We now find the second space derivative of $\tilde{v}$, again by differentiating g_α. To this end we use (4.16) (and that ϕ is even) to yield

$$\begin{aligned}
\frac{\partial^2}{\partial\tilde{x}^2}g_\alpha\left(t,\tilde{x}\right)=&-\frac{\partial k}{\partial\tilde{x}}\left[\phi\left(k+\theta\sqrt{\tau}\right)\left(1-\alpha\right)He^{-r\tau}+\phi\left(k+2\theta\sqrt{\tau}\right)\frac{\tilde{x}}{c}e^{\left(-2r+\theta^2\right)\tau}\right]\\
&-\Phi\left(k+2\theta\sqrt{\tau}\right)\frac{1}{c}e^{\left(-2r+\theta^2\right)\tau}\\
=&-\frac{\partial k}{\partial\tilde{x}}\phi\left(k\right)\left[\left(1-\alpha\right)He^{-r\tau}\frac{e^{r\tau}cH\left(\alpha-1\right)}{\tilde{x}}+\frac{1}{c}\tilde{x}e^{\left(-2r+\theta^2\right)\tau}\frac{e^{\left(2r-\theta^2\right)\tau}c^2H^2\left(\alpha-1\right)^2}{\tilde{x}^2}\right]\\
&-\Phi\left(k+2\theta\sqrt{\tau}\right)\frac{1}{c}e^{\left(-2r+\theta^2\right)\tau}\\
=&-\Phi\left(k+2\theta\sqrt{\tau}\right)\frac{1}{c}e^{\left(-2r+\theta^2\right)\tau}.
\end{aligned}$$

Hence

$$\frac{\partial^2}{\partial\tilde{x}^2}\tilde{v}\left(t,\tilde{x}\right)=-\frac{1}{c}e^{\left(-2r+\theta^2\right)\tau}\left(\Phi\left(k\left(\alpha\right)+2\theta\sqrt{\tau}\right)-\Phi\left(k\left(0\right)+2\theta\sqrt{\tau}\right)\right).\tag{4.18}$$

We use (4.18) to derive the optimal control

$$\pi_t^*=\frac{\theta}{\sigma}\tilde{X}_t^{\tilde{x}^*}\frac{\partial^2}{\partial\tilde{x}^2}\tilde{v}\left(t,\tilde{X}_t^{\tilde{x}^*}\right),\qquad 0\leq t\leq T.$$

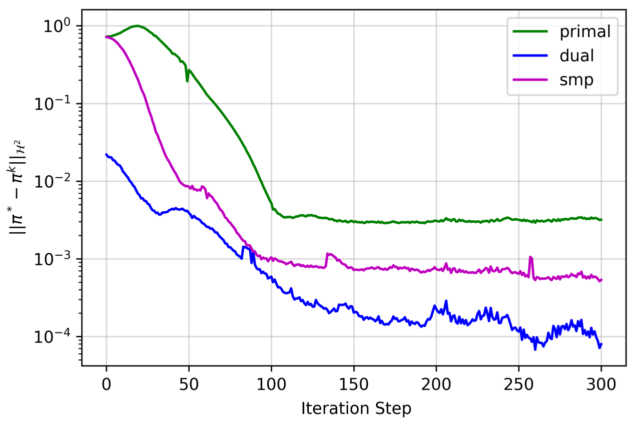
Figure 4.6 shows the error of our approximate control against π^*, and the value approximations at time 0 for our 3 algorithms. We take $\alpha=0.55$, $H=2$ and $c=-1$. For the market parameters, we take $\mu=0.06$, $r=0.05$ and $\sigma=0.2$. We then take $T=1.0$ and $N=20$ for our algorithm.

Motivated by Theorem 3.4.9, we would expect to see convergence of the form

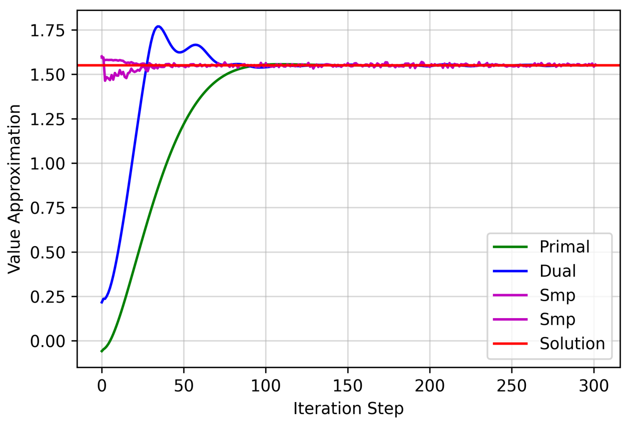
$$\|\pi^k-\pi^*\|_{\mathcal{H}^2}\leq C\hat{c}^k,\qquad k\in\mathbb{N}_0$$

for some $C \geq 0$ and $\hat{c} \in (0,1)$, where π^k is the control process at the kth iteration step of the DC2BSDE algorithm. We log scale the error of the control, so should see linear convergence. Due to the stochastic nature of the algorithm, high approximation volatility is introduced from the small 'mini batch' sample size at each step (in this case 64). This explains why we only see approximately linear decay up to a point where the error becomes roughly constant.

From (b), we see converge of the value function for all 3 graphs, with the 'dual' value approximations forming upper bounds of the value. There are two lines for the SMP, indicating the upper and lower bounds v_{high} and v_{low} respectively, c.f. (2.23). we see that they are the same at the beginning of the algorithm, possibly due to the initialisation of the parameter $\tilde{x}$, but quickly form tight bounds of the value at time 0. However, we see some deviation of the value approximation over all iterations. this is due to the volatility of using Monte Carlo methods with a batch size of 64 at each iteration step for the DCSMP algorithm.



(a) Control error against iteration step (k).



(b) Value approximation against iteration step.

Figure 4.6: Control error (in $\mathcal{H}^2$) and value approximation of the primal and dual DC2BSDE methods and the DSMP method for the unconstrained quadratic utility problem.

Example 4.1.5 (Power Utility in a Stochastic Volatility Model)**.** In this example we consider the stochastic volatility model (2.10). When we use the power utility $U(x) = \frac{x^p}{p}$ for $0 < p < 1$, exact solutions can be obtained for this problem. A solution is derived in Ma et al. (2020) by assuming that the value function v has the form

$$v(t,x,\Sigma) = \frac{x^p}{p} \exp\left(C(t) + D(t)\Sigma\right), \qquad (t,x,\Sigma) \in [0,T] \times \mathbb{R}^2 \tag{4.19}$$

in terms of two unknown functions C and D that satisfy a system of Riccati type of ordinary differential equations for which numerical solutions can be found. We use $v(t,x,\Sigma)$ as a benchmark. For our application we take $r = 0.05$, $\rho = -0.5$, $p = 0.5$, $\kappa = 1$, $\bar{\Sigma} = 0.05$, $\xi = 0.5$, $\theta = 0.5$, $x_0 = 1$, and $\Sigma_0 = 0.5$.

Table 4.5 shows the relative error and run time of the primal and dual DC2BSDE algorithms when we use a range of terminal times T and numbers of time steps N. We contrast this to the results of Ma et al. (2020), in which, with $T = 1.0$ and $N = 100$, upper and lower bounds for the value were found with relative errors of 0.003% in 2240 seconds and 27600 seconds respectively, albeit for slightly different parameter choices. The upper bound was found faster using a convenient guess of the form the dual control γ. We are not using 100 time steps for our algorithms due to computer memory constraints. In fact, the 50 time step algorithm is also slower than expected due to a memory bottleneck. For $T = 0.2$ the solution is 2.03289, for $T = 0.5$ it is 2.07559 and when $T = 1.0$ the solution is 2.13420. The run time for the $T = 0.5$ and $T = 1.0$ algorithms is higher than for $T = 0.2$ since twice the iteration steps were taken, 10000 compared to 5000 for $T = 0.2$. While the errors for the primal algorithm reduce as N increases, the value does not lie within the duality gap for some terminal times. This may be down to a discretisation error, and would be fixed by taking more time steps. We see convergence of the approximations to the true value as N increases and this convergence seems to be linear, as we saw in Example 4.1.1.

Now, let us treat Σ as some generic random process, meaning that we can only sample Σ, and not the underlying Brownian motion W^Σ. We cannot directly apply Theorem 2.3.3 as Σ is not measurable with respect to the filtration generated by W^S. To circumvent this problem, we adopt a market completion method (c.f. Karatzas and Shreve (1998) for example) by first constructing an artificial stock driven by W^Σ, adding it to our market, and then making it unavailable for trading. To be specific, we define a new asset price S^Σ by

$$dS_t^\Sigma = S_t^\Sigma \left(r dt + dW_t^\Sigma \right), \qquad 0 \le t \le T,$$

which ensures that Σ is adapted to the natural filtration of the Brownian motions driving the market. For simplicity we assume that $m = 1$, but this analysis extends naturally to multiple dimensions. We invest in this market with a 2-dimensional portfolio, restricted to the set $K \times \{0\}$, ensuring that our wealth process X^π is still defined as in (2.8), controlled by a 1-dimensional K-valued process π. Now the dual state process (2.44) is controlled by $\tilde{x}$ and a 2-dimensional dynamic control denoted by $(\tilde{\pi}, \gamma)$ valued in $\tilde{K} \times \mathbb{R}$, the polar cone of $K \times \{0\}$.

T	N	Primal Value	Rel-Err (%)	Time (s)	Dual Value	Rel-Err (%)	Time (s)
0.2	5	2.03310	1.01654E-02	83	2.03378	4.35850E-02	98
0.2	10	2.03301	5.74161E-03	206	2.03344	2.66047E-02	239
0.2	20	2.03296	3.27670E-03	591	2.03314	1.23181E-02	604
0.2	50	2.03289	2.81206E-04	2849	2.03295	2.89678E-03	2622
0.5	5	2.07616	2.75979E-02	152	2.07969	1.97584E-01	183
0.5	10	2.07582	1.14780E-02	362	2.07747	9.08529E-02	411
0.5	20	2.07572	6.36082E-03	927	2.07670	5.38088E-02	1030
0.5	50	2.07562	1.55757E-03	5070	2.07600	1.98421E-02	4181
1.0	5	2.13328	4.32410E-02	154	2.14847	6.68466E-01	170
1.0	10	2.13377	1.99985E-02	362	2.14264	3.95457E-01	400
1.0	20	2.13399	9.92254E-03	918	2.13715	1.38171E-01	981
1.0	50	2.13403	7.97535E-03	4230	2.13548	6.01058E-02	4395

Table 4.5: Accuracy and runtime for the deep controlled 2BSDE method for various terminal times T and numbers of time steps N applied to the unconstrained power utility problem in a stochastic volatility model.

We would like to show that we can take $\gamma = 0$, as this would return the dual problem to the 1-dimensional control case independent of W^{Σ}. We want this as we cannot 'see' W^{Σ}, and can only sample Σ directly at any time. To this end, let $\left(\tilde{Y}^{\tilde{x},\tilde{\pi},\gamma}, \tilde{Z}^{\tilde{x},\tilde{\pi},\gamma}\right)$ be the dual adjoint solutions. Fix a time $t \in [0,T]$ and let $\tilde{Z}^{\tilde{x},\tilde{\pi},\gamma} = (z^1, z^2)$ for $z^1, z^2 \in \mathcal{H}^2(\mathbb{R})$. By Theorem 2.3.3 (2) we have $\left(\tilde{Y}_t^{\tilde{x},\tilde{\pi},\gamma}\right)^{-1} z_t^2 = 0$ for all $t \in [0,T]$, hence $z^2 = 0$. However, this means that condition (3) becomes

$$\tilde{Y}_t^{\tilde{x},\tilde{\pi},\gamma} \delta_K(\tilde{\pi}_t) + \left(z_t^1\right)^{\top} \Sigma_t^{-1} \tilde{\pi}_t = 0, \qquad 0 \le t \le T,$$

which is independent of γ, leading to a free control. We therefore can take $\gamma = 0$, removing the dependence of the dual state process on W^{Σ}. Note that this choice of control may not be optimal, and the decision to remove dependence on W^{Σ} may lead to a non-zero duality gap.

Table 4.6 shows the accuracy and runtime of the deep SMP method with a range of terminal times and step sizes, when we run it on the same problem formulation as above. We note that the upper bounds are an order of magnitude less accurate the lower bounds for this problem. This indicates that the implied primal process approximations are more accurate than the dual approximations in this algorithm. Comparing these results to Table 4.5, the SMP method gives roughly the same accuracy of bounds as the dual deep controlled 2BSDE method, but with a lower computation time. In particular, the algorithm here only needed 5000 iterations steps, compared to 10000 for both the 2BSDE algorithms.

T	N	v_{low}	Rel-Diff (%)	v_{high}	Rel-Diff (%)	Time (s)
0.2	5	2.03271	9.17595E-03	2.03348	2.90360E-02	103
0.2	10	2.03280	4.40981E-03	2.03323	1.64607E-02	273
0.2	20	2.03284	2.60193E-03	2.03305	7.49044E-03	781
0.2	50	2.03289	5.76610E-05	2.03295	2.54626E-03	4788
0.5	5	2.07512	2.25897E-02	2.07974	2.00124E-01	102
0.5	10	2.07538	9.97694E-03	2.07738	8.63624E-02	301
0.5	20	2.07563	2.04008E-03	2.07656	4.67255E-02	778
0.5	50	2.07556	1.16593E-03	2.07595	1.74975E-02	3818
1.0	5	2.13025	1.84896E-01	2.14786	6.40234E-01	104
1.0	10	2.13232	8.78697E-02	2.13995	2.69313E-01	263
1.0	20	2.13489	3.25920E-02	2.13750	1.54844E-01	761
1.0	50	2.13440	9.35931E-03	2.13551	6.11910E-02	4604

Table 4.6: Accuracy and runtime for the deep SMP method for various terminal times T and numbers of time steps N applied to the unconstrained power utility problem in a stochastic volatility model.

Example 4.1.6 (Heston with non-HARA utility)**.** We now look at a higher dimensional example with no explicit solution. Consider the same stochastic volatility market, but with multiple stocks. For simplicity we assume all stocks are independent with the same parameters, but this is not a necessary restriction. We let U have the form (4.1). For this model we do not have a convenient ansatz for v to obtain a solution, so we simply compare our algorithms. Additionally, we increase the dimensionality of the problem and allow Σ to be a $R^{m\times m}$ valued matrix.

For this example let us take $T = 0.2$, $N = 20$, and $m = 10$. We then have $n = 11$, and $d = 20$ in (3.12). For the 2BSDE algorithms, the combined dimensions of the unknown processes are 131 for the primal and 141 for the dual. We consider the unconstrained problem, so we can set $\eta = 0$. For the SMP algorithm we only need to find processes in 20 dimensions. We run the algorithm for 10000 steps. Table 4.7 shows the results of this. We give the value approximation for the 2BSDE algorithms, and the average of the upper and lower bound for the SMP. We also provide the runtime to show the efficiency of the algorithms as a whole, but the individual differences should be taken with a pinch of salt due to implementation differences. There are a number of factors that affect the algorithm run times. For example, we see that the dual algorithm takes twice as long as the primal, but we did not simplify the algorithm to reflect the constraints of the problem. Making this change would have reduced the run time. The run time has increased from the previous low dimensional example, but sub-linearly. The relative difference between approximations has increased to about 1%.

Method	Value	Time (s)
Primal	2.61059	2316
Dual	2.63758	5636
SMP	2.62514	1840

Table 4.7: Value approximations for the DC2BSDE and DCSMP algorithms applied to the unconstrained non-HARA utility problem in a stochastic volatility model.

Example 4.1.7 (Power Utility in a Path Dependent Volatility Model)**.** In this example not only is the control problem non-Markovian, but it cannot be made Markovian by including additional information in our state space. This means that we cannot apply our DC2BSDE method.

This example is motivated by the concept that the volatility of a stock should be dependent on its value. If the stock is doing well the volatility is assumed to be lower. This may be due to infrequent trading of the stock, or the stability of the large company to which the stock is assigned. We consider an extreme example, where there are two levels of volatility. The normal higher level is maintain until the stock reaches its historical maximum, in which case it instantly transitions to the lower volatility.

Consider the market (2.7), where the process Σ is a function of the stock history. For each $t \in [0, T]$, let Σ_t be a diagonal matrix, with ith diagonal entry $\Sigma_t^{(i)}$ defined as

$$\Sigma_t^{(i)} = \begin{cases} \Sigma_{\text{high}}, & S_t^{(i)} < \sup_{s \le t} S_s^{(i)}, \\ \Sigma_{\text{low}}, & S_t^{(i)} = \sup_{s \le t} S_s^{(i)}, \end{cases}$$

where $\Sigma_{\text{low}} < \Sigma_{\text{high}}$ are constants and $S_s^{(i)}$ is the ith stock price at time $s \in [0, T]$. In this model, the volatility of a stock is inversely proportional to the ratio of its value and the running maximum. This makes the diffusion coefficient of the wealth process X a function of the entire path of S, instead of just at one time. This path dependence ensures that the utility maximisation problem is truly non-Markovian, in the sense that it cannot be made Markovian by the introduction of a new variable. We do not need to apply an artificial stock market argument here, as the volatility is a measurable function of the adapted stock market process up to time t, hence is progressively measurable.

For our application, we consider a 2-dimensional stock market with cone constraint $K = \mathbb{R}_+^2$. We take $\Sigma_{\text{low}} = 0.2$ and $\Sigma_{\text{high}} = 0.3$. We then take $r_t = 0.05$, $\mu_t = 0.06\mathbb{1}_2$ for $t \in [0, T]$ and $p = 0.5$. Table 4.8 shows the duality gap when this algorithm is run for varying terminal times and numbers of time steps. We run the algorithms for 10000 iteration steps. The 'Diff' and 'Rel-Diff' columns show the absolute and relative distances between the upper and lower approximations respectively. We see in all cases that the duality gap reduces as the number of time steps increases. We do not have a benchmark to compare these results to, as the path dependence and incompleteness of the problem make it very difficult to solve analytically.

4.2 Investment-Consumption with Randomly Terminating Income

We now provide numerical results for the example given in section 2.4.

4.2.1 Finding the true state process

The DC2BSDE algorithm gives us the value function in terms of paths of $X^{\pi^*, c^*, a}$, the state process with perpetual income. However, we wish to find the optimal state process X^{π^*, c^*} with random income termination. We can do this using the information we have obtained from the algorithm and the no income solutions. We cannot directly compute $v\left(t, X_t^{\pi^*, c^*}\right)$ for $t \in [0, T]$, as we do not have a functional representation of v. However, we do have $\partial_{xx}^2 v\left(t, X_t^{\pi^*, c^*}\right)$ as the output of our neural network, so if we have the optimal state and control processes, we

T	N	v_{low}	v_{high}	Diff	Rel-Diff (%)
0.2	5	2.01057	2.01066	8.99222E-05	4.47246E-03
0.2	10	2.01054	2.01063	8.55058E-05	4.25287E-03
0.2	20	2.01057	2.01063	6.00701E-05	2.98772E-03
0.2	50	2.01061	2.01062	1.30984E-05	6.51462E-04
0.5	5	2.02652	2.02682	3.03732E-04	1.49879E-02
0.5	10	2.02654	2.02676	2.11323E-04	1.04278E-02
0.5	20	2.02648	2.02665	1.73093E-04	8.54166E-03
0.5	50	2.02637	2.02650	1.31536E-04	6.49120E-03
1.0	5	2.05333	2.05421	8.87694E-04	4.32320E-02
1.0	10	2.05337	2.05397	5.93055E-04	2.88821E-02
1.0	20	2.05327	2.05368	4.16821E-04	2.03005E-02
1.0	50	2.05307	2.05339	3.20542E-04	1.56128E-02

Table 4.8: Duality gap for the deep SMP method for various terminal timed T and numbers of time steps N applied to the cone-constrained power utility problem with path dependent volatility.

can plug these into our 2BSDE with $X^{\pi^*,c^*,a}$ replaced by X^{π^*,c^*}. Furthermore we note that, for $t \in [0,T]$,

$$\begin{aligned}
c^*\left(t, X_t^{\pi^*,c^*}\right) &= I\left(\partial_x v\left(t, X_t^{\pi^*,c^*}\right)\right) \\
&= I\left(\partial_x v\left(t, X_t^{\pi^*,c^*}\right)\right)\left(\mathbb{1}_{\tau>t} + \mathbb{1}_{\tau\leq t}\right) \\
&= I\left(\partial_x v\left(t, X_t^{\pi^*,c^*,a}\right)\right)\mathbb{1}_{\tau>t} + I\left(\partial_x v_0\left(t, X_t^{\pi^*,c^*}\right)\right)\mathbb{1}_{\tau\leq t}
\end{aligned}$$

The first term is an output of our DC2BSDE algorithm, and the second can be computed analytically (where we assume that τ is independent of W). We have a similar result for π^*. We can therefore simulate the value process and optimal state and control processes as follows. Until the random time τ, we simulate the paths of X^{π^*,c^*} as we did for $X^{\pi^*,c^*,a}$, and the same for the value processes. Then, once the income terminates, we update the processes using the optimal no-income controls. We can still use the neural network forming $\Gamma^{\pi^*,c^*,a}$ to update, we just apply them at X^{π^*,c^*} instead of $X^{\pi^*,c^*,a}$. Using this network assures us that the terminal condition holds up to some small approximation error. To be specific, we use the following scheme. The input is the parameters $v_0, z_0, \lambda_i, \theta_i$ from the (primal or dual) DC2BSDE method.

Set $X_0 = x_0$, $V_0 = v_0$, $Y_0 = z_0$, $(\pi_0, c_0) = \mathcal{N}_{\theta_0}(x_0)$. Let $\tau \sim \mathrm{EXP}(\eta)$. Then for $i = 0, \dots N-1$:

$$\begin{aligned}
\Gamma_i &= \mathcal{N}_{\lambda_i}(X_i), \\
X_{i+1} &= X_i + \left[X_i\left(r_{t_i} + \pi_i^\top(\mu_{t_i} - r_{t_i})\right) - c_i + a_{t_i}\right] h + X_i \pi_i^\top \Sigma_{t_i} \Delta W_i, \\
V_{i+1} &= V_i + \left[\alpha V_i - U(c_i)\right] h + Y_i X_i \pi_i^\top \Sigma_{t_i} \Delta W_i, \\
Y_{i+1} &= Y_i + \left[\alpha Y_i - \left(Y_i\left(r_{t_i} + \pi_i^\top(\mu_{t_i} - r_{t_i})\right) + \Gamma_i X_i |\Sigma_{t_i}^\top \pi_i|^2\right)\right] h + \Gamma_i X_i \pi_i^\top \Sigma_{t_i} \Delta W_i, \\
(\pi_{i+1}, c_{i+1}) &= \begin{cases} \mathcal{N}_{\theta_{i+1}}(X_{i+1}) & t_{i+1} < \tau, \\ -\Sigma_t^{-1}\theta_t\left(\frac{\partial_x v_0(t_{i+1}, X_{i+1})}{\partial_{xx}^2 v_0(t_{i+1}, X_{i+1}) X_{i+1}}, I\left(\partial_x v_0(t_{i+1}, X_{i+1})\right)\right) & \text{otherwise,} \end{cases}
\end{aligned}$$

where ΔW_i is a multivariate normal $\mathcal{N}_d(0, (t_{i+1} - t_i)\mathbb{1}_d)$ random variable. For the dual problem, we can use a similar method, where we use the dual-primal relations, for $t \in [0, T]$,

$$\begin{aligned}
\Gamma_t^{\pi^*, c^*} &= \frac{1}{\tilde{\Gamma}_t^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}}, \\
\pi_t^* &= -\Sigma_t^{-1}\theta_t \frac{\tilde{\Gamma}_t^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*} \tilde{X}_t^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}}{\tilde{Y}_t^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}}, \\
c_t^* &= I\left(-\tilde{X}_t^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}\right),
\end{aligned}$$

where $\left(\tilde{X}^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}, \tilde{Y}^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}, \tilde{\Gamma}^{\tilde{x}^*, \tilde{\pi}^*, \gamma^*}\right)$ are the dual 2BSDE processes arising from the dual problem (2.45) for optimal dual controls $(\tilde{x}^*, \tilde{\pi}^*, \gamma^*)$. Again, these controls are used before the income terminates, and then we use the analytical no income controls.

4.2.2 Numerical Examples

4.2.2.1 Power Utility

We take $m = 1$, $K = \mathbb{R}$ and let r, σ, λ be constants. Consider the power utility case $U_1(x) = U(x) = x^p/p$, $p < 1$, $p \neq 0$. Furthermore, suppose that $\eta = 0$, meaning that the income is either non-terminating or zero. Then we can solve the HJB equation exactly using a suitable ansatz for the value function. To this end, suppose that v has the form

$$v(t, x) = (x + h(t) a)^p \frac{f(t)}{p}, \qquad (t, x) \in [0, T] \times (0, \infty)$$

for some functions f and h to be determined. We can then write the optimal Markovian using (2.37) controls as

$$\pi^*(x,t) = \frac{\lambda(x + h(t)a)}{\sigma x(1-p)}, \qquad c^*(x,t) = (x + h(t)a) f(t)^{\frac{1}{p-1}}.$$

Substituting these into the HJB equation (1.7), dividing by $(x + h(t)a)^{p-1} \neq 0$ and multiplying by p, we have , for $(t,x) \in [0,T] \times (0,\infty)$,

$$0 = (x + h(t)a)\left(f'(t) - \delta f(t) + \frac{p}{q} f(t)^q + prf(t) - \frac{1}{2}\lambda^2 q f(t)\right) + apf(t)(h'(t) - rh(t) + 1),$$

where $q = \frac{p}{p-1}$. Since this holds for all x, we must have

$$\begin{cases} f'(t) - \delta f(t) + \frac{p}{q} f(t)^q + prf(t) - \frac{1}{2}\lambda^2 q f(t) = 0 \\ f(t)(h'(t) - rh(t) + 1) = 0 \end{cases} \tag{4.20}$$

with terminal conditions $f(T) = 1$ and $h(T) = 0$. To solve the first equation in (4.20), let $f(t) = g(t)^{1-p}$, then g satisfies a linear ODE

$$\begin{aligned} g'(t) &= kg(t) - 1 \qquad 0 \leq t \leq T \\ g(T) &= 1, \end{aligned}$$

where

$$k = (1-q)\delta + qr + \frac{1}{2}\lambda^2(1-q)q.$$

The solution of this ODE is

$$g(t) = \left(1 - \frac{1}{k}\right)e^{-k(T-t)} + \frac{1}{k}, \qquad 0 \leq t \leq T,$$

if $k \neq 0$ and $g(t) = T - t + 1$ otherwise. In particular, $g(t) > 0$ for all t, so f is strictly positive, we can cancel it in the second equation of (4.20) to get an equation for h only. The solution is (we assume $r > 0$)

$$h(t) = \frac{1}{r}\left(1 - e^{-r(T-t)}\right), \qquad 0 \leq t \leq T.$$

This gives the value function as

$$v(t,x) = \frac{1}{p} C(T-t)^{1-p} \left(x + \frac{a}{r}\left(1 - e^{-r(T-t)}\right)\right)^p, \qquad (t,x) \in [0,T] \times (0,\infty).$$

where $C(t) = \left(1 - \frac{1}{k}\right) e^{-kt} + \frac{1}{k}$. Note that the value function for a lifetime optimal consumption problem can be derived by letting T tend to infinite, provided $k > 0$ or equivalently $\delta > rp - \frac{1}{2}\lambda^2 q$. We may find the primal and dual no-income value functions v_0 and $\tilde{v}_0$ as

$$\begin{aligned} v_0(t,x) &= \frac{1}{p} x^p C(T-t)^{1-p}, \\ \tilde{v}_0(t,\tilde{x}) &= \sup_x \{v_0(t,x) - xy\} = -\frac{1}{q} \tilde{x}^q C(T-t), \end{aligned}$$

for $(t,x,\tilde{x}) \in [0,T] \times (0,\infty)^2$. Note that the value function for the perpetual optimal consumption problem can be derived by letting $T \to \infty$, provided $k > 0$ or equivalently $\delta > \frac{1}{2} q\lambda^2 - rp$. We use these functions to build the true optimal control processes.

We first provide a table of value approximations of the primal and dual algorithms. When available, we compare these to analytical solutions. We also consider the duality gap relative to the primal approximation. For this application we take the values $\delta = 0.1$, $r = 0.05$, $\sigma = 0.2$, $\lambda = 0.05$, $p = 0.5$ and $T = 1.0$. Tables 4.11 and 4.12 show the approximations for varying a, η and number of time steps N. The closed-form solution is 2.80532 when $a = 0.1$ and $\eta = 0$, and 2.67777 when $a = 0$. For all cases we see a decrease in the duality gap that is linear with the increasing number of time steps, however in some instances the dual value approximation is below the solution. It is also notable that the introduction of a positive η improves the accuracy of the algorithm. This may be due to the extra penalty terms in the Hamiltonian and process dynamics, which help the training of the neural networks.

N	Primal Value	rel-err (%)	Dual Value	rel-err (%)	duality gap (%)
5	2.66833	3.52523E-01	2.67541	8.80398E-02	2.65419E-01
10	2.67262	1.92247E-01	2.67657	4.47551E-02	1.47776E-01
20	2.67475	1.12496E-01	2.67716	2.26510E-02	8.99458E-02
50	2.67636	5.23836E-02	2.67754	8.48612E-03	4.39205E-02

Table 4.9: Results for $a = 0, \eta = 0$ for power utility.

Figure 4.7 shows how the value approximation behaves between these corner cases. Graph (a)

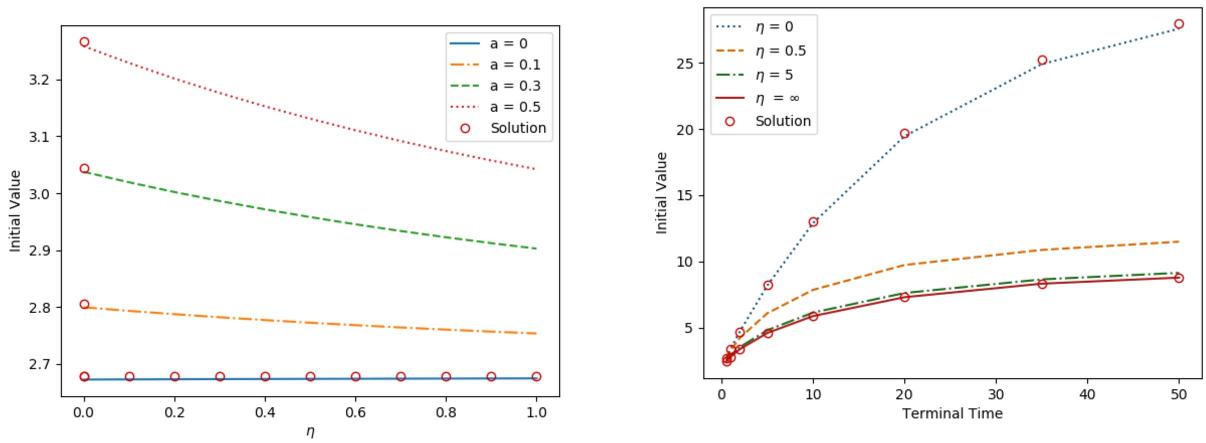
N	Primal Value	rel-err (%)	Dual Value	rel-err (%)	duality gap (%)
5	2.67389	1.44731E-01	2.67728	1.82146E-02	1.26700E-01
10	2.67575	7.53019E-02	2.67735	1.56624E-02	5.96844E-02
20	2.67659	4.37925E-02	2.67755	8.27258E-03	3.55354E-02
50	2.67725	1.91742E-02	2.67768	3.26298E-03	1.59143E-02

Table 4.10: Results for $a = 0, \eta = 2$ for power utility.

N	Primal Value	rel-err (%)	Dual Value	rel-err (%)	duality gap (%)
5	2.79487	3.72507E-01	2.80100	1.53993E-01	2.19431E-01
10	2.79962	2.03185E-01	2.80312	7.84224E-02	1.24986E-01
20	2.80202	1.17634E-01	2.80420	3.99241E-02	7.78521E-02
50	2.80371	5.76122E-02	2.80501	1.11405E-02	4.64984E-02

Table 4.11: Results for $a = 0.1, \eta = 0$ for power utility.

shows the relation of the termination frequency η to the income a. We see that the value is more sensitive to changes in the income size than the termination frequency. Graph (b) shows the value approximation as the terminal time increases for different termination frequencies.



(a) Sensitivity of value to a and η.

(b) Initial value approximation against terminal time for different values of η.

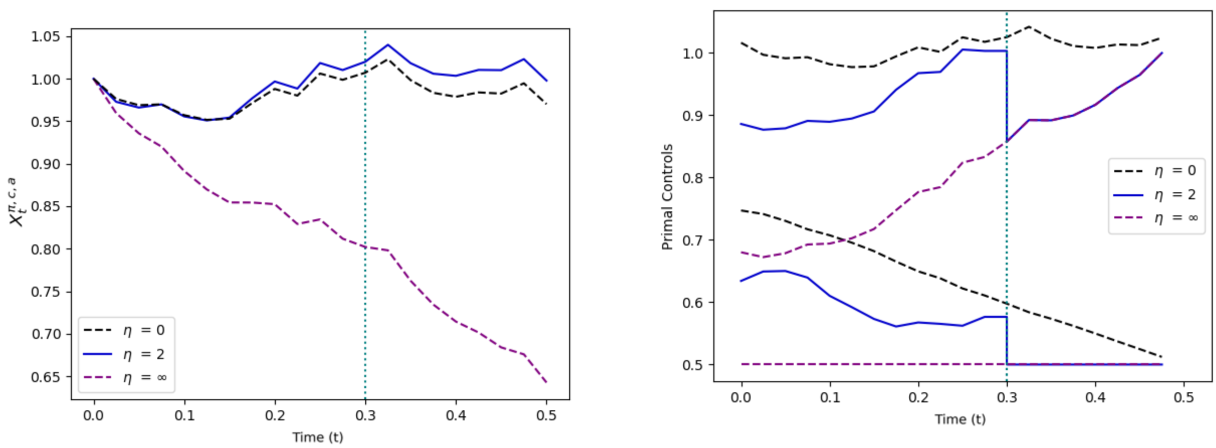
Figure 4.7: Sensitivity of initial value for power utility.

Figure 4.8 shows a sample path of the state, control, value and value derivative processes when there is an income of $a = 1.0$ that terminates with intensity $\eta = 2$. This is the most general case and as such there are no solution paths. The control choice changes from the neural

N	Primal Value	Dual Value	duality gap (%)
5	2.72674	2.72798	4.62485E-02
10	2.72996	2.73066	2.25402E-02
20	2.73167	2.73215	1.51645E-02
50	2.73284	2.73305	7.74071E-03

Table 4.12: Results for $a = 0.1, \eta = 2$ for power utility.

network control to the analytical control at the time indicated by the vertical dotted line. For comparison, we add the paths when there is income, and where there is no income. The state paths are the closed form solution paths, and the control paths are the solution functions applied to the random income state path, so we can compare the functions directly.



(a) Simulation of the process X.

(b) Simulation of the control. Upper is c, lower is π.

Figure 4.8: A sample path of random income state and control processes, $T = 0.5$, $N = 20$ with $\eta = 2$, $a = 1.0$ for power utility. The sample termination time is indicated by the vertical dotted line.

Considering Figure 4.8 (b), we see a jump in the control at the termination time, to a more conservative one where we do not invest as much and hold on to more wealth, rather than consuming it. The result of this choice is that the wealth process increases compared to the perpetual wealth, as per graph (a), to perhaps compensate for the income we are no longer receiving.

Comparing the control to the benchmark cases, we see that the initial consumption and investment are lower than the perpetual benchmark control but, while the income has not terminated, gradually get closer. This makes sense intuitively, as both the probability of losing the income and potential income lost reduce as time goes on.

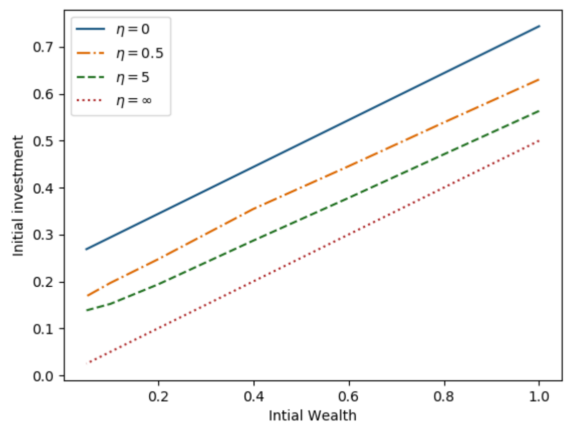
An interesting property of this problem is that even if one starts with zero initial wealth, the income allows us to generate wealth, and as such the agent's value will not be zero. In Figure 4.9 we look at the limit of the initial investment and consumption strategy, value, and derivative of the value as the initial wealth x_0 goes to zero. Due to instability issues (and the benchmark investment going to infinity), we cannot simply plug in zero into our algorithms, so we consider sufficiently small values of x_0 instead. We compare the initial terms for various valued of η, while fixing $a = 0.5$. The lines corresponding to $\eta = 0$ and ∞ are calculated using the closed form solutions evaluated at time zero. The two intermediate cases are found using our primal DC2BSDE algorithm. We see that the physical amount consumed or invested decreased linearly with the initial wealth, but is higher for smaller values of η, which corresponds to expecting to receive income for longer. The value has a roughly square root relationship with initial wealth, the same as the utility. The derivative process increases drastically at zero as η decreases, indicating the stark difference between some initial wealth and no wealth when the income terminates rapidly.

4.3 LQ Problem

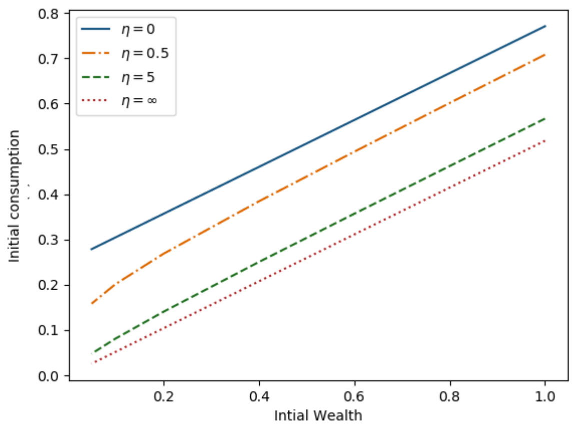
In this section we provide numerics for the LQ problem (3.1).

4.3.1 Unconstrained

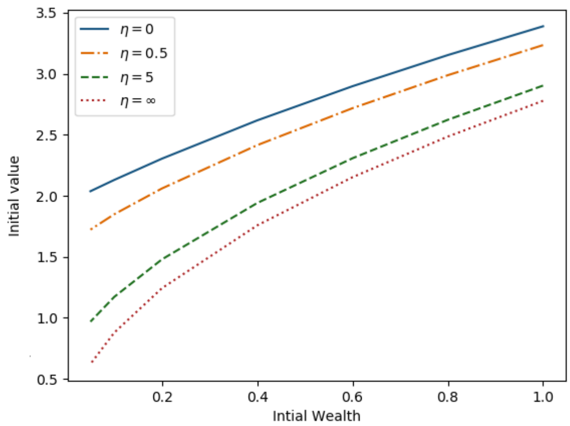
With the unconstrained problem, we know the controls are linear, so given some initialised $\alpha^0 \in \mathcal{C}^0([0,T];\mathbb{R}^{m\times n})$ and $\beta^0 \in \mathcal{C}^0([0,T];\mathbb{R}^m)$ we can implement (3.59) to get successive controls ϕ^k, $k \in \mathbb{N}_0$.



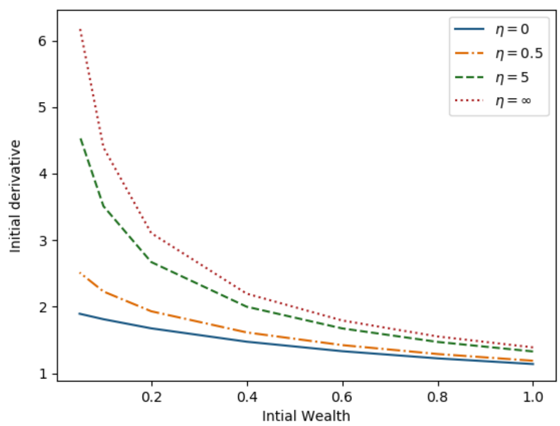
(a) Initial investment strategy.



(b) Initial consumption strategy.



(c) Initial value.



(d) Initial space derivative of value.

Figure 4.9: Initial control strategies and value for different initial wealths for power utility.

4.3.1.1 Implementation and an Example

Recalling (3.59), we are implementing the following update formula for $k \in \mathbb{N}_0$, $t \in [0, T]$

$$\begin{aligned}
\alpha_t^{k+1} &= \alpha_t^k - \tau \left(B_t^\top a_t^k + D_t^\top a_t^k \left(C_t + D_t \alpha_t^k \right) + R_t \alpha_t^k + S_t \right), \\
\beta_t^{k+1} &= \beta_t^k - \tau \left(B_t^\top b_t^k + \left(D_t^\top a_t^k D_t + R_t \right) \beta_t^k \right),
\end{aligned}$$

where a^k, b^k solve the ODE in $s \in [0, T]$

$$\begin{aligned}
\dot{a}_s^k + a_s^k \left(A_s + B_s \alpha_s^k \right) + A_s^\top a_s^k + C_s^\top a_s^k \left(C_s + D_s \alpha_s^k \right) + \left(Q_s + S_s^\top \alpha_s^k \right) &= 0, \\
\dot{b}_s^k + A_s^\top b_s^k + \left(a_s^k B_s + C_s^\top a_s^k D_s + S_s^\top \right) \beta_s^k &= 0.
\end{aligned} \tag{4.21}$$

In the implementation, we solve these ODEs using an ODE solver in Python. It may be faster to implement the update formulas (3.60) using a finite sum approximation

$$(1 - L_{\alpha^k})^{-1} \approx \sum_{j=0}^{M} L_{\alpha^k}^j,$$
$$(1 - L_A)^{-1} \approx \sum_{j=0}^{M} L_A^j,$$

for all $k \in \mathbb{N}_0$, for some sufficiently large $M \in \mathbb{N}$. Due to the complexity of the integral operators we chose not to do this. We discretise the time space $[0, T]$ into a partition $(t_i)_{i=0}^N$ for some $N \in \mathbb{N}$ and output the optimal control evaluated at these time points. In our implementation we will take $t_i = i\frac{T}{N}$. For the ODE solver, we need a function, not just discrete points, so we use a piecewise constant approximation of α^k and β^k.

The algorithm will perform this update until a convergence criterion is reached. This is either a maximum number of iterations, or if the algorithm does not improve enough over a single iteration. In this case, as we have a benchmark solution, we can compare the error directly, and deem the algorithm to converge if the relative error difference

$$\left| \frac{\left\|\alpha^k - \hat{\alpha}\right\|_\infty - \left\|\alpha^{k-1} - \hat{\alpha}\right\|_\infty}{\left\|\alpha^{k-1} - \hat{\alpha}\right\|_\infty} \right|, \qquad k \in \mathbb{N},$$

is sufficiently small.

An implementation of this algorithm in Python can be found at `https://github.com/Ashley-Davey/PGM-for-LQ-Problem`

Algorithm 6: Unconstrained Policy Gradient Method for LQ Problem

Result: Optimal control and value function

for $i = 0, 1, 2, \dots, N-1$ **do**

  Initialise $\alpha_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{m\times n}\right)$ and $\beta_i^0 \sim \text{Unif}\left([-0.1, 0.1]^{m}\right)$;

  Set $k = 0$;

end

while α^k, β^k *not converged* **do**

  Solve ODEs (4.21) to output $a_i^k = a_{t_i}^k$ and $b_i^k = b_{t_i}^k$ for $i = 0, 1, 2, \dots, N-1$;

  Generate α_i^k, β_i^k using (3.59) with $t = t_i$ for $i = 0, 1, 2, \dots, N-1$;

  Set $k = k + 1$;

end

Output control $x \mapsto \alpha_i^k x + \beta_i^k$ for $i = 0, 1, 2, \dots, N-1$, and value function $x \mapsto x^\top a_0^k x + x^\top b_0^k$;

For our implementation of the unconstrained problem we choose coefficients dependent on the state dimension. We set each of A, B, C, D, Q, S to be a constant random matrix (with Q symmetric), then scale by $n^{\frac{-3}{2}}$. This scaling factor is used to ensure that the set of coefficients satisfies the assumptions of our convergence analysis for each n, and was chosen by inspection. Informally speaking, this scaling outweighs the increase in size of the matrices when the dimension increases, leading to smaller values of the drift and diffusion of X, along with Q and S, and hence reducing the value function. We set G to be a random symmetric matrix, and take R as the identity matrix, multiplied by some $r \in \mathbb{N}$. Note that in the case of random coefficients, the matrices

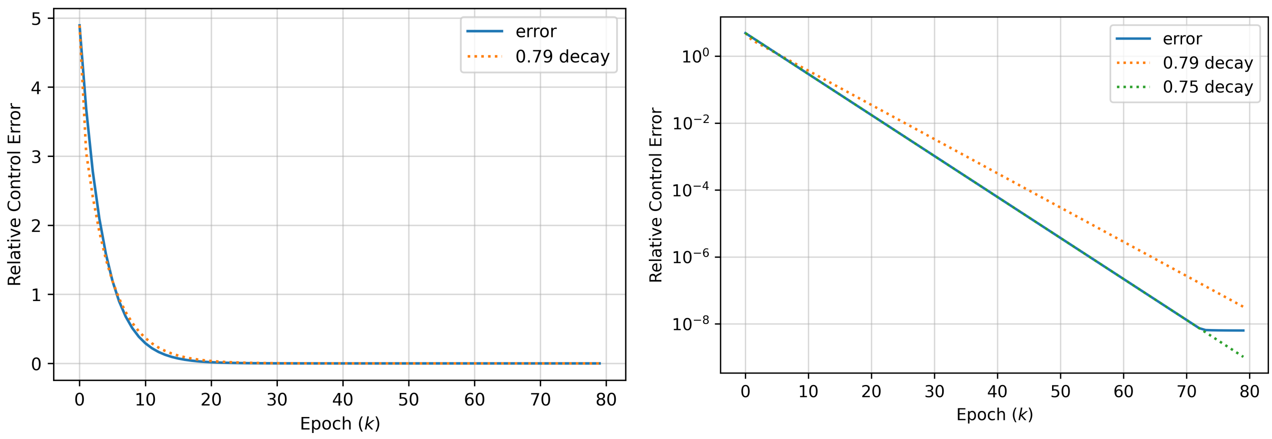
$$\begin{pmatrix} Q_s & S_s^\top \\ S_s & R_s \end{pmatrix}, \qquad 0 \le s \le T$$

may not be positive definite. Indeed, one can simply test if the eigenvalues of this symmetric matrix are positive, and more often than not this is not the case. However, there does not seem to be any impact on the solvablilty of the problem, or convergence of the PPGM algorithm. We set $T = 1.0$ throughout. The learning rate is chosen to be

$$\tau = 0.9 \min\left(\frac{2}{\mu + \|R\|_\infty}, \frac{\mu + \|R\|_\infty}{2\mu\|R\|_\infty}\right),$$

in line with Assumption 1.

Example 4.3.1. Let us first consider one problem with $n = m = 10$. For these coefficients, we can calculate $c^* = 0.79$. We test if this convergence rate is matched by the algorithm. We run the algorithm for 80 epochs, with a total runtime of 7 seconds.



(a) Relative control error.

(b) Relative control error (log scaled)

Figure 4.10: Relative control error of the unconstrained PGM method applied to the LQ problem, compared to approximated and theoretical convergence rates.

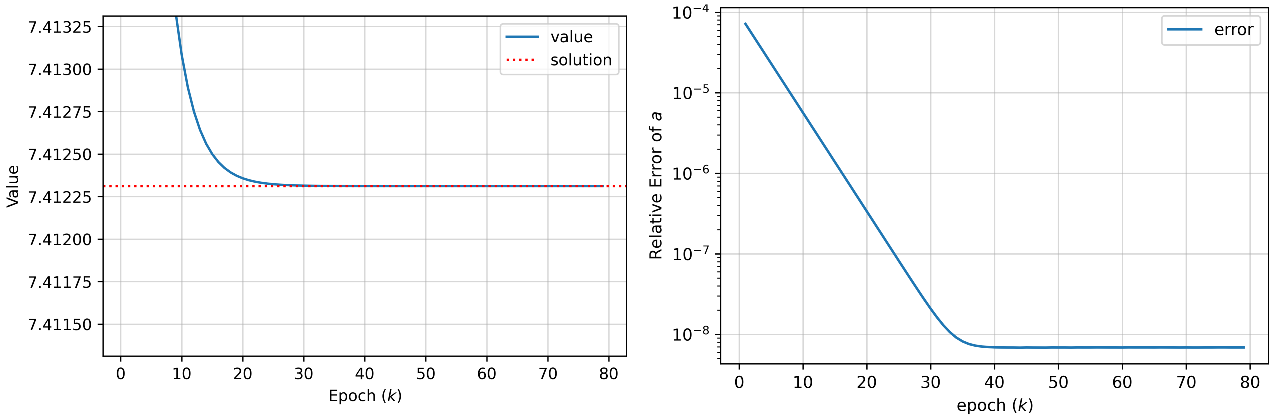
Figure 4.10 numerically verifies the first statement of Theorem 3.5.10. We plot the relative control error, given at epoch $k \in \mathbb{N}_0$ by

$$\frac{\max\left(\|\alpha^k - \hat{\alpha}\|_\infty, \|\beta^k\|_\infty\right)}{\|\hat{\alpha}\|_\infty}.$$

and compare it to the theoretical error rate $\hat{c}^k$ (multiplied by the initial error). The first graph clearly shows the error going to 0. The second graph is log-scaled so we can see the decay more closely.

We see that the error rate (the blue line) is lower than the theoretical rate (the orange dotted line), as per Theorem 3.5.10(i). To see the tightness of our theoretical bound on the convergence rate, we log-scale the error. From here, we can perform linear regression on the results (up to the point where the error flattens out, at just over 70 epochs) to obtain a decay coefficient of 0.75 for our algorithm, given by the green dotted line. The error flattening out indicated that the algorithm cannot produce more accuracy beyond standard numerical errors.

Figure 4.10 shows the error of the value function approximation. We see that the error is also exponentially decaying, as it was for the control. The value matrix process a^k is compared to the solution, denoted here as a^*, of the Ricatti equation (3.99). We see that the log scaled



(a) Value approximation.

(b) Value function error (log scaled)

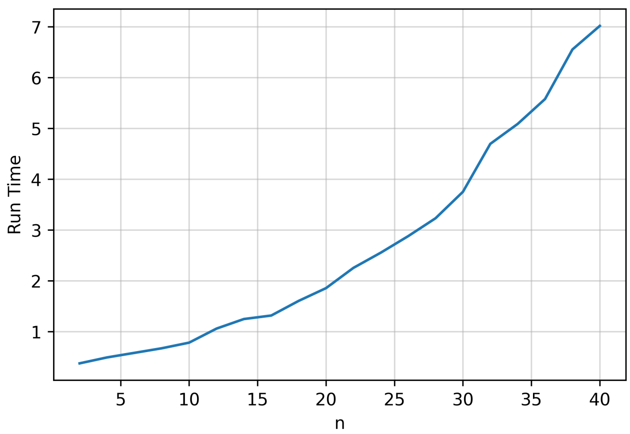
Figure 4.11: Value function approximation, in terms of initial value and error of the process a^k in $\mathcal{L}^\infty$.

error flattens out faster than it did for the control error, suggesting that the value function reached the terminal accuracy while the control process was still somewhat inaccurate. The solution benchmark can be calculated as 7.41231 via solving the Ricatti equations, and the value approximation in very accurate, with a final relative accuracy of 10^{-8}.

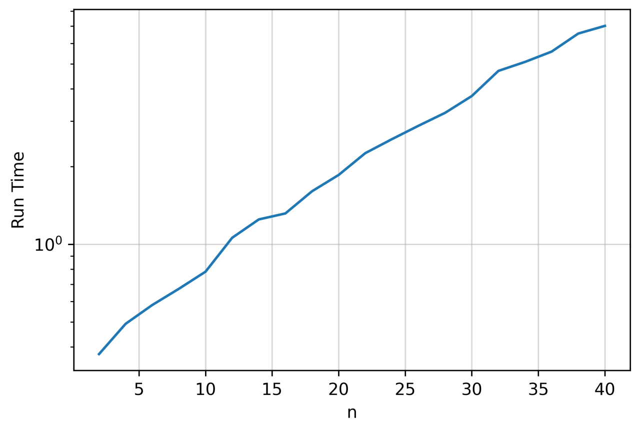
Furthermore, the convergence in $\mathcal{L}^\infty$ tells us that the sequence a^k converges to a symmetric matrix. However, for any particular $k \in \mathbb{N}_0$, a^k may be a non symmetric matrix, as it is defined via a non-symmetric ODE (4.21).

4.3.1.2 Dimensionality

Example 4.3.2. We now consider changes in the dimensions n and m of the problem. As the underlying coefficients depend on n (in order to ensure the assumptions of the convergence results hold) the accuracy of the algorithm artificially increases as n increases. We therefore only consider the effect on the runtime. Figure 4.12 shows the effect of the runtime on the dimension when we set $n = m$ to range between 1 and 40. We log scale the results to see that the runtime is not quite exponential in terms of dimension, but is still super-linear. This is possibly due to the solving method for the underlying ODEs.



(a) Run time against dimension.



(b) Run time against dimension (log scaled)

Figure 4.12: Run time in seconds against dimension $n = m$.

4.3.1.3 Assumptions

We establish the details of Remark 14. To this end, let all coefficients other than R be fixed and define

$$R_t = r\mathbb{1}_{m,m}, \qquad 0 \leq t \leq T,$$

for some $r > 0$. In this instance we have $\mu = r$ and $\|R\|_\infty = r\sqrt{m}$. Let K denote some constant independent of r, whose precise definition may change from line to line. We can write the previously defined constants as follows (excluding those independent of r, or not used in the assumptions). Without loss of generality, we assume $\phi^0 = 0$.

$$\begin{aligned}
c_0 &= Kr, \\
c_7 &= 1 + \frac{K}{r}, \\
c_8 &= K\left(1 + e^{\frac{K}{r}}\right), \\
c_9 &= K\left(1 + C_a\left(c_\alpha\right)\right), \\
c_\alpha &= \frac{Ke^{\frac{K}{r}}}{r}, \\
C_a\left(c_\alpha\right) &= Ke^{\frac{K}{r}e^{\frac{K}{r}}}\left(1 + \frac{e^{\frac{K}{r}}}{r}\right).
\end{aligned}$$

The assumptions are then as follows (note that once $r > 0$ is chosen, τ can be chosen to satisfy Assumption 1).

$$\begin{aligned}
Ke^{c_7} &\leq c_0, \\
Kc_7e^{c_7} &\leq c_0, \\
K\left(c_9 + C_a\left(c_\alpha\right)\right) &< c_0, \\
K\left(e^{Kc_\alpha}\left(1 + C_a\left(c_\alpha\right)\right)\left(1 + c_\alpha\right)\right) &\leq c_0.
\end{aligned}$$

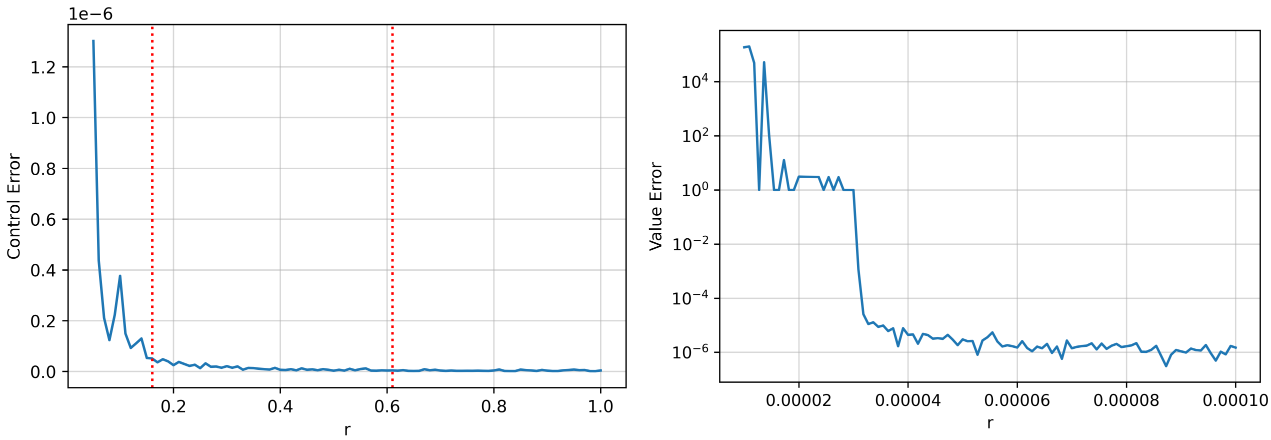
Taking the limit as r tends to infinity we see

$$\begin{aligned}
c_7 &\to 1, \\
c_8 &\to K, \\
c_\alpha &\to K, \\
C_a\left(c_\alpha\right) &\to K, \\
c_9 &\to K.
\end{aligned}$$

Therefore the left hand sides are bounded, and we may take r sufficiently large such that c_0 is larger than this bound.

Example 4.3.3. We fix $n = m = 5$ and consider the problem varying only the value of r. For this problem there exist $r_1 > r_2 > 0$ such that in the region $[r_1, \infty)$ all assumptions are satisfied, in the region $[r_1, r_2)$ at least one of the assumptions are not satisfied, typically one of (3.66). In the region $[0, r_2)$ we have $\hat{c} > 1$, so we do not have any theoretical guarantees of convergence. We numerically determine $r_1 = 0.61$ and $r_2 = 0.16$. Figure 4.13 (a) shows the final control error of the algorithm for different values of r. We see in all cases the error is very small (note the 1e-6 scaling of the y axis). However, we also see a pattern of the error increasing between the dotted line, which are placed at r_1 and r_2. In the right-most region where all assumptions are satisfied, the graph is flat, in the middle region the error is decreasing in r, and in the left-most region the error increases dramatically. We take r as small as possible without incurring any errors in the underlying code, in this case $r = 0.05$.

Example 4.3.4. We now consider a case with r arbitrarily small. We cannot take $r = 0$ as the PPGM method becomes chaotic. We take $n = m = 1$, and take r very small. The results are given in Figure 4.13 (b). We see that for r smaller than some critical value $r_{\min}$, that the algorithm does not accurately return the value at time 0, and appears to be chaotic. For this particular example, it appears that the critical value is $r_{\min} = 0.000033$.



(a) Control error vs r.

(b) Value error vs r (log scaled).

Figure 4.13: Control error vs r for large values of r with $m = 5$, and values error against r for small values of r with $m = 1$.

4.3.2 Constrained

4.3.2.1 Equivalence of PGM and DCBSDE

We consider the numerical application of the PPGM scheme. We cannot directly implement the method, so we instead use approximating functions at each stage. Consider first the unconstrained scheme, for $k \in \mathbb{N}_0$ and $(t, x) \in [0, T] \times \mathbb{R}^n$,

$$\phi_t^{k+1}(x) := \phi_t^k(x) - \tau\left(B_t^\top Y_t^{t,x,\phi^k} + D_t^\top Z_t^{t,x,\phi^k} + R_t\phi_t^k(x) + S_t x\right), \tag{4.22}$$

where for given $\phi \in \mathcal{V}$ the processes $\left(X^{t,x,\phi}, Y^{t,x,\phi}, Z^{t,x,\phi}\right) \in \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{S}^2(t, T; \mathbb{R}^n) \times \mathcal{H}^2(t, T; \mathbb{R}^n)$ are determined by the following FBSDE, for $s \in [t, T]$

$$\begin{aligned}
dX_s^{t,x,\phi} &= \left(A_s X_s^{t,x,\phi} + B_s\phi_s\left(X_s^{t,x,\phi}\right)\right) ds + \left(C_s X_s^{t,x,\phi} + D_s\phi_s\left(X_s^{t,x,\phi}\right)\right) dW_s, \\
X_t^{t,x,\phi} &= x, \\
dY_s^{t,x,\phi} &= -\left(A_s^\top Y_s^{t,x,\phi} + C_s^\top Z_s^{t,x,\phi} + S_s^\top\phi_s\left(X_s^{t,x,\phi}\right) + Q_s X_s^{t,x,\phi}\right) ds + \left(Z_s^{t,x,\phi}\right)^\top dW_s, \\
Y_T^{t,x,\phi} &= G X_T^{t,x,\phi}.
\end{aligned} \tag{4.23}$$

We may not be able to exactly solve this FBSDE, so we do so numerically. One way to do it is via machine learning. We can use the Deep BSDE algorithm Han et al. (2018) to determine pathwise solutions

$$\left(X_i^j, Y_i^j, Z_i^j\right) \qquad i = 1, \ldots, N, j = 1, \ldots B.$$

for some discretisation size N and batch size B. We can then find, for $i = 1, \ldots, N$ and $j = 1, \ldots, B$,

$$\tilde{\phi}_{i+1}^j := \phi_i\left(X_i^j\right) - \tau\left(B_{t_i}^\top Y_i^j + D_{t_i}^\top Z_i^j + R_{t_i}\phi_i\left(X_i^j\right) + S_{t_i}X_i^j\right), \tag{4.24}$$

however this does not give us a function to use in the next stage. We therefore need to use find a functional approximation of $\tilde{\phi}_{i+1}$ given input-output data $\left(X_i^j, \tilde{\phi}_{i+1}^j\right)_{j=1}^B$ for each i. We again use ML for this, setting, for $x \in \mathbb{R}^n$

$$\begin{aligned}
\phi_i(x) &= \mathcal{N}_\theta(x), \\
\tilde{\phi}_i(x) &= \mathcal{N}_{\tilde{\theta}}(x,),
\end{aligned}$$

for some parameter sets $\theta, \tilde{\theta}$. We have the loss function

$$\mathcal{L}(\theta) = \sum_{j=1}^B \left|\mathcal{N}_\theta\left(X_i^j\right) - \tilde{\phi}_{i+1}^j\right|^2,$$

and then perform the gradient descent

$$\begin{aligned}
\tilde{\theta} &= \theta - \rho\partial_\theta\mathcal{L}(\theta) \\
&= \theta - 2\rho\sum_j \left(\partial_\theta\mathcal{N}_\theta\left(X_i^j\right)\right)\left(\mathcal{N}_\theta\left(X_i^j,\right) - \tilde{\phi}_{i+1}^j\right) \\
&= \theta - 2\rho\sum_j \left(\partial_\theta\mathcal{N}_\theta\left(X_i^j,\right)\right)\left(\mathcal{N}_\theta\left(X_i^j\right) - \phi_i\left(X_i^j\right) + \tau\left(B_{t_i}^\top Y_i^j + D_{t_i}^\top Z_i^j + R_{t_i}\phi_i\left(X_i^j\right) + S_{t_i}X_i^j\right)\right) \\
&= \theta - 2\tau\rho\sum_j \left(\partial_\theta\mathcal{N}_\theta\left(X_i^j,\right)\right)\left(B_{t_i}^\top Y_i^j + D_{t_i}^\top Z_i^j + R_{t_i}\phi_i\left(X_i^j\right) + S_{t_i}X_i^j\right) \\
&=: \theta - \mu\sum_j \left(\partial_\theta\mathcal{N}_\theta\left(X_i^j,\right)\right)\partial_\phi H\left(t_i, X_i^j, \mathcal{N}_\theta\left(X_i^j,\right), Y_i^j, Z_i^j\right) \\
&= \theta - \mu\sum_j \partial_\theta H\left(t_i, X_i^j, \mathcal{N}_\theta\left(X_i^j,\right), Y_i^j, Z_i^j\right).
\end{aligned}$$

This is exactly the update in the DC2BSDE.

In the constrained case we do not apply the proximal function at each stage. Instead we implement soft-constraints: we add an additional function to the loss function that is proportional to the distance of the control from the constraint space. To be specific, we perform the update

$$\tilde{\theta} := \theta - \mu \sum_j \partial_\theta \left[H\left(t_i, X_i^j, \mathcal{N}_\theta\left(X_i^j, \right), Y_i^j, Z_i^j\right) + \operatorname{dist}\left(\mathcal{N}_\theta\left(X_i^j, \right), U\right)\right].$$

This update does not guarantee that at any stage our control satisfies the control constraints, but in the limit we should have a control within the constraint space.

We run the algorithm with a certain learning rate μ until a convergence criterion is met, either a maximum number of iterations, or if the BSDE and control losses are (on average) not changing with iteration step. In practise we take a linear regression of the last 50 iteration steps for both losses, and check if the linear coefficients are sufficiently small.

Example 4.3.5. Let us consider the cone constraint $U = [0, \infty)^m$. We choose our coefficients to have time dependence in this example, so there are periods where the optimal control is in the interior of the space and others where it is on the boundary. We set, for $t \in [0, T]$,

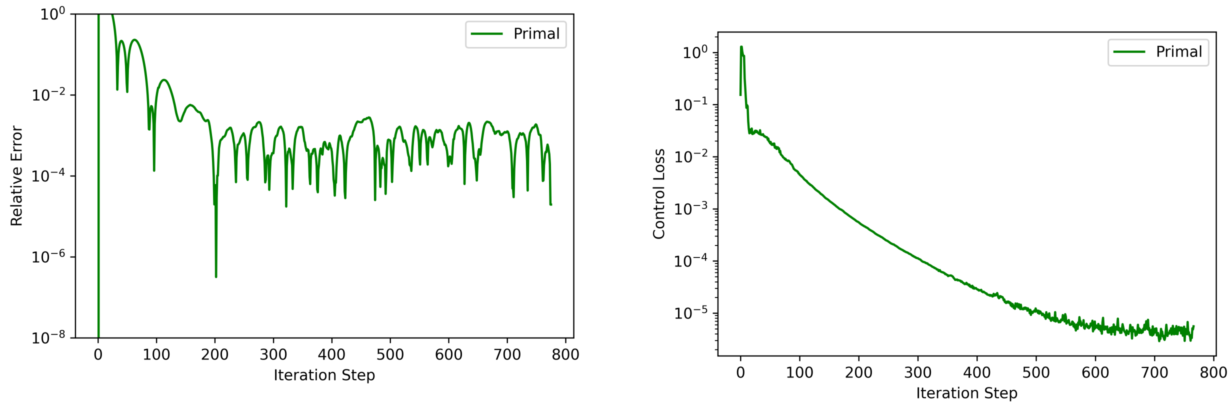
$$\begin{aligned} A_t &= \tilde{A} \sin(2\pi t) \\ B_t &= \tilde{B} \cos(2\pi t), \\ C_t &= \tilde{C} \cos(2\pi t), \\ D_t &= \tilde{D} \sin(2\pi t), \end{aligned}$$

for some arbitrary matrices $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}$ in the appropriate dimensions, scaled by $n^{-3/2}$. We take Q, S to be constant matrices, scaled by $n^{-3/2}$, with Q symmetric, G as a symmetric matrix and R to be a constant identity matrix multiplied by some constant $r > 0$. Given this cone constraint, we use benchmark solutions given by solving the Ricatti equations derived in Wu et al. (2020). To agree with these equations, we take $n = 1$. Figures 4.14, 4.15 and 4.16 show the results. We take $N = 50$, $T = 1.0$ and $m = 1$. We use the learning rate $\mu = 0.01$.

Figure 4.14 shows the evolution of the relative error of the value approximation and (derivative of) the control loss over the algorithm. We see that there is more variation in the error here than in the ODE based PGM for the unconstrained case, due to the stochastic gradient descent adding extra volatility to the approximates. We see the derivative of the control loss goes to zero as expected for a machine learning, and the relative error becomes small.

Figure 4.15 shows the accuracy of the state and control process approximations. We see that our approximations of the state and control process closely match the benchmark solutions. As previously mentioned, the control process moves out of the constraint space, but only by a small amount. Once we have the output control process, we can project this onto the constraint space to fix this issue.

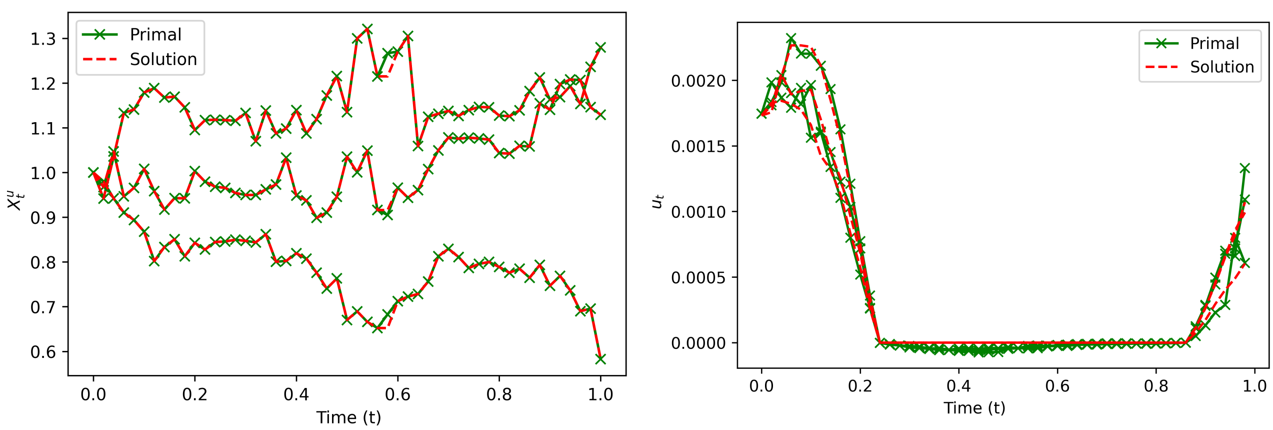
Figure 4.16 shows the accuracy of the approximation of the value and adjoint processes V^u and Y^u respectively, suggesting good approximation capabilities of the algorithm for the value function and its derivative.



(a) Relative error of value against iteration step (log scaled).

(b) Control loss against iteration step (log scaled).

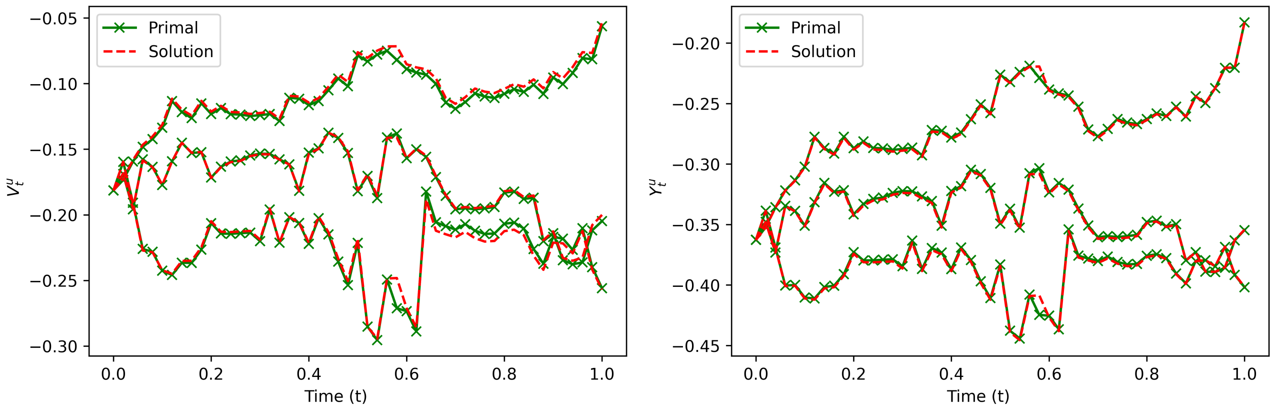
Figure 4.14: Relative Error and control loss function against iteration step for the cone constrained LQ problem.



(a) State process

(b) Control process.

Figure 4.15: 3 Simulations of the state and control process for the cone constrained LQ problem.



(a) Value process

(b) Adjoint process.

Figure 4.16: 3 Simulations of the value and adjoint processes for the cone constrained LQ problem.

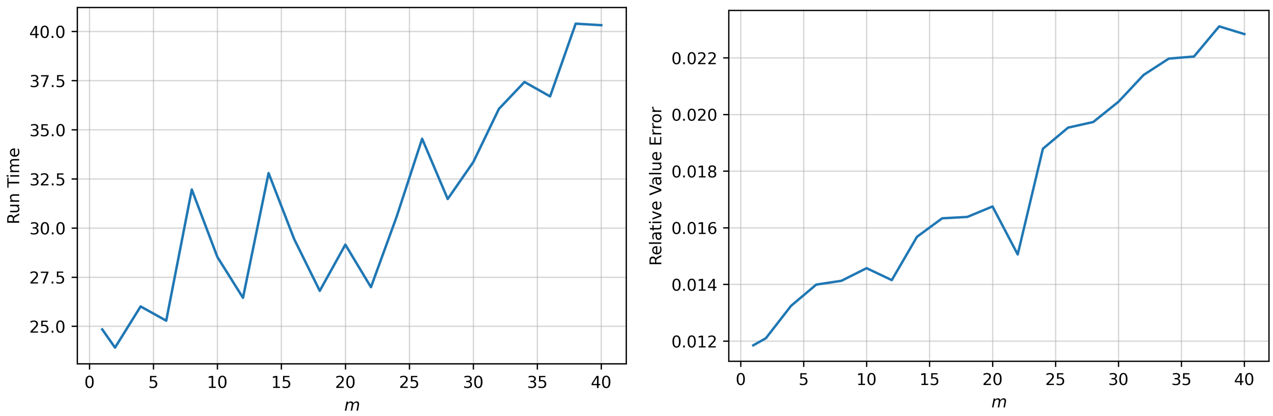
4.3.2.2 Dimensionality

Example 4.3.6. We again consider the effect of on the runtime of the algorithm. In this instance, we will only increase m. Unlike the previous, we do not get an artificial increase in the accuracy of the algorithm, as the coefficients do not scale with m, the scaling only depends on n which is set to 1. We set $N = 10$ and $T = 1$. To reduce the impact of dimension on the optimal control and solution, we take all coefficients (other than R) to be constants, with 0.1 in each element. In this case, the optimal control is identically equal to 0, and the optimal state process is independent of m. It is therefore clearer to see the impact of dimension on the error of the algorithm. We use a piecewise constant learning rate, starting at $\mu = 0.1$ and reducing to 0.01 halfway through the (maximum) 3000 iteration steps.

Figure 4.17 shows the results. We see an increase in runtime and relative error with dimension, as is to be expected. Due to the stochastic nature of the algorithm, it is not clear to see the exact relationship between the two. Due to the additional complexity of the algorithm, the runtime is a factor of ten higher than the unconstrained algorithm.

4.3.2.3 Neural Network Approximation Capabilities

Finally, we consider the $m = n = 1$ unconstrained case and look at the control process at a specific time point. We take $N = 50$ and consider the penultimate time point $t_{N-1} = 0.98$. To capture a wider control space, we run the algorithm at two start points, $x_0 = 1$ and $x_0 = 2$.



(a) Run time against dimension.

(b) Relative error of value again dimension.

Figure 4.17: Runtime and accuracy against control dimension for the cone constrained LQ problem.

Figure 4.18 shows the results. The back ground dotted lines indicate the density of $X_{t_{N-1}}$, and a large value would indicate a high amount of X data is used in training, so the accuracy of the control function should be high. We indeed see that in the regions that the density is high, the control approximations in blue and orange for $x_0 = 1$ and $x_0 = 2$ respectively are accurate, but move away from the solution (red) at points with low densities. Note that the state space does not pass $x = 0$ as the optimal state process is a geometric Brownian motion.

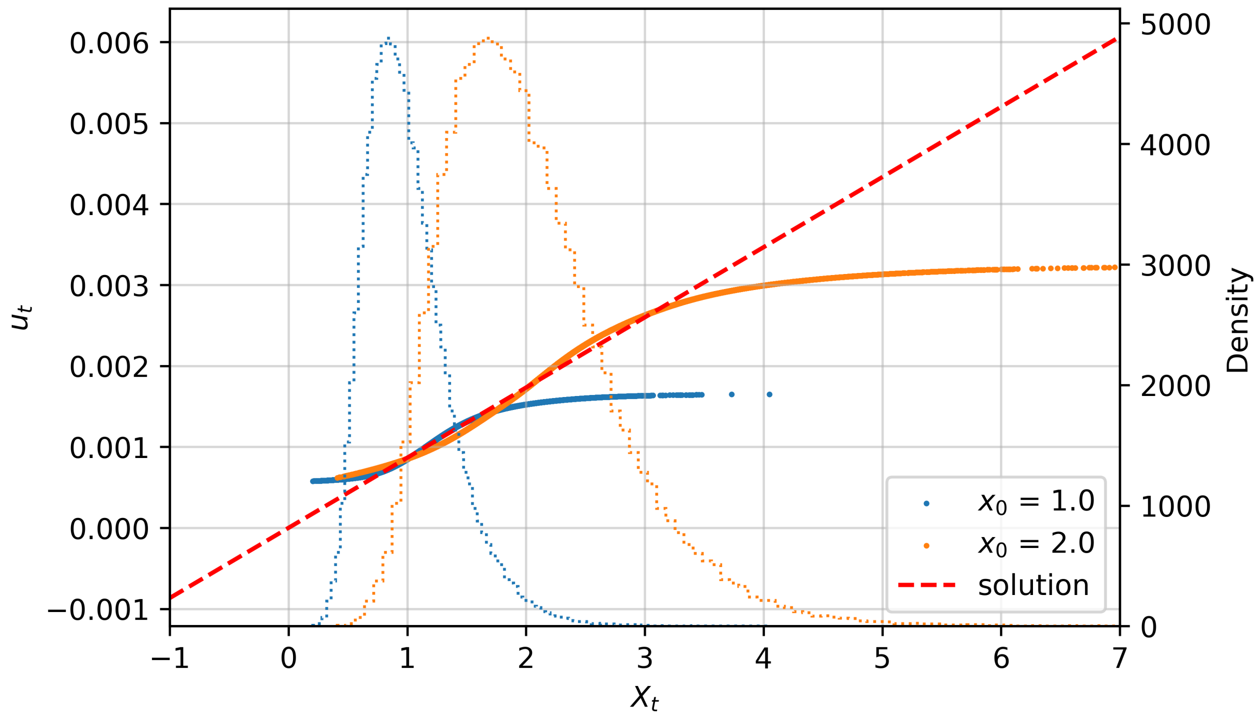


Figure 4.18: Control neural network and solution function at $t = 0.98$ for the unconstrained LQ problem.

Chapter 5

Conclusions

In this thesis we developed methods for solving stochastic control problems using techniques from machine learning, stochastic analysis and duality theory. Our novel approach involves solving optimality conditions to find the optimal control, using BSDE solutions of the adjoint equation, and the BSDE representation of the HJB equation.

To study convergence of this algorithm, we convert our so-called DC2BSDE algorithm into a theoretical PPGM algorithm. The existing literature for convergence of this kind of algorithm does not cover our case, as there is control in the diffusion of the state. For the linear quadratic problem, we establish sufficient conditions under which the control induced by the PPGM converges linearly to the optimal control.

This thesis is contained with a very active research area, and there are a number of outstanding open problems that are left for further research.

- The dual problem for the utility maximisation problem was presented in the 1-dimensional state space case. The dual problem can also be established in the case that the state space is multidimensional, but the Brownian motion is one dimensional, for example the linear quadratic problem. However a dual problem in the general case, and for a more general class of problems, is yet to be established.
- The machine learning method can be adapted to suit a number of control problems, as we have seen for the Heston problem and the investment consumption problem with randomly terminating income. It will be interesting to see how this methodology can be

applied in future to a wider range of problems, perhaps with less model-specific stricture as we have seen in our implementation.

- We present a number of sufficient conditions for convergence of the PPGM method for a linear quadratic problem. It would be interesting to see if these conditions can be relaxed, or how they extend to a more general class of problems.

Bibliography

Anthony, M. and Bartlett, P. L. (2009). *Neural Network Learning: Theoretical Foundations*. Cambridge University Press.

Bach, F. (2017). Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681.

Bachouch, A., Huré, C., Langrené, N., and Pham, H. (2021). Deep neural networks algorithms for stochastic control problems on finite horizon: numerical applications. *Methodology and Computing in Applied Probability*.

Beck, C., Becker, S., Cheridito, P., Jentzen, A., and Neufeld, A. (2019a). Deep splitting method for parabolic PDEs. *arXiv:1907.03452*.

Beck, C., E, W., and Jentzen, A. (2019b). Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *Journal of Nonlinear Science*, 29(4):1563–1619.

Becker, S., Cheridito, P., and Jentzen, A. (2019). Deep optimal stopping. *Journal of Machine Learning Research*, 20(74):1–25.

Bismut, J.-M. (1973). Conjugate convex functions in optimal stochastic control. *Journal of Mathematical Analysis and Applications*, 44(2):384–404.

Buehler, H., Gonon, L., Teichmann, J., and Wood, B. (2019). Deep hedging. *Quantitative Finance*, 19(8):1271–1291.

Carmona, R., Laurière, M., and Tan, Z. (2019). Linear-quadratic mean-field reinforcement learning: convergence of policy gradient methods. *arXiv preprint arXiv:1910.04295*.

Cassel, A. B. and Koren, T. (2021). Online policy gradient for model free learning of linear quadratic regulators with $\sqrt{T}$ regret. In *International Conference on Machine Learning*, pages 1304–1313. PMLR.

Davey, A., Monoyios, M., and Zheng, H. (2021). Duality for optimal consumption with randomly terminating income. *Mathematical Finance*, 31(4):1275–1314.

Davey, A. and Zheng, H. (2022). Deep learning for constrained utility maximisation. *Methodology and Computing in Applied Probability*, 24(2):661–692.

De Spiegeleer, J., Madan, D. B., Reyners, S., and Schoutens, W. (2018). Machine learning for quantitative finance: Fast derivative pricing, hedging and fitting. *Quantitative Finance*, 18(10):1635–1643.

E, W., Han, J., and Jentzen, A. (2017). Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4):349–380.

Eckstein, S. and Kupper, M. (2019). Computation of optimal transport and related hedging problems via penalization and neural networks. *Applied Mathematics & Optimization*.

El Karoui, N., Peng, S., and Quenez, M. C. (1997). Backward stochastic differential equations in finance. *Mathematical finance*, 7(1):1–71.

Exarchos, I. and Theodorou, E. A. (2018). Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. *Automatica*, 87:159–165.

Fazel, M., Ge, R., Kakade, S., and Mesbahi, M. (2018). Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 1467–1476. PMLR.

Giegrich, M., Reisinger, C., and Zhang, Y. (2022). Convergence of policy gradient methods for finite-horizon stochastic linear-quadratic control problems. *arXiv preprint arXiv:2211.00617*.

Gonon, L., Muhle-Karbe, J., and Shi, X. (2020). Asset pricing with general transaction costs: Theory and numerics. *SSRN 3387232*.

Grohs, P., Hornung, F., Jentzen, A., and Von Wurstemberger, P. (2018). A proof that artificial neural networks overcome the curse of dimensionality in the numerical approximation of black-scholes partial differential equations. *arXiv:1809.02362*.

Hambly, B., Xu, R., and Yang, H. (2021a). Policy gradient methods find the nash equilibrium in n-player general-sum linear-quadratic games. *arXiv preprint arXiv:2107.13090*.

Hambly, B., Xu, R., and Yang, H. (2021b). Policy gradient methods for the noisy linear quadratic regulator over a finite horizon. *SIAM Journal on Control and Optimization*, 59(5):3359–3391.

Han, J. and E, W. (2016). Deep learning approximation for stochastic control problems. *arXiv:1611.07422*.

Han, J., Jentzen, A., and E, W. (2018). Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510.

Henry-Labordere, P. (2017). Deep primal-dual algorithm for BSDEs: Applications of machine learning to CVA and IM. *SSRN 3071506*.

Henry-Labordere, P., Litterer, C., and Ren, Z. (2016). A dual algorithm for stochastic control problems: Applications to uncertain volatility models and cva. *SIAM Journal on Financial Mathematics*, 7(1):159–182.

Huré, C., Pham, H., Bachouch, A., and Langrené, N. (2021). Deep neural networks algorithms for stochastic control problems on finite horizon: convergence analysis. *SIAM Journal on Numerical Analysis*, 59(1):525–557.

Imkeller, P., Dos Reis, G., and Salkeld, W. (2019). Differentiability of sdes with drifts of super-linear growth. *Electronic journal of probability*, 24:1–43.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 37:448–456.

Ji, S., Peng, S., Peng, Y., and Zhang, X. (2022). Solving stochastic optimal control problem via stochastic maximum principle with deep learning method. *Journal of Scientific Computing*, 93(1):30.

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

Karatzas, I. and Shreve, S. E. (1998). *Methods of Mathematical Finance*. Springer.

Karoui, N. E. and Tan, X. (2013). Capacities, measurable selection and dynamic programming part i: abstract framework. *arXiv preprint arXiv:1310.3363*.

Kerimkulov, B., Šiška, D., and Szpruch, L. (2021). A modified msa for stochastic control problems. *Applied Mathematics & Optimization*, pages 1–20.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Labbé, C. and Heunis, A. J. (2009). Conjugate duality in problems of constrained utility maximization. *Stochastics*, 81(6):545–565.

Leshno, M., Lin, V. Y., Pinkus, A., and Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867.

Li, Y. and Zheng, H. (2018). Dynamic convex duality in constrained utility maximization. *Stochastics*, 90(8):1145–1169.

Liu, S., Oosterlee, C. W., and Bohte, S. M. (2019). Pricing options and computing implied volatilities using neural networks. *Risks*, 7(1):16.

Ma, J., Li, W., and Zheng, H. (2020). Dual control monte-carlo method for tight bounds of value function under heston stochastic volatility model. *European Journal of Operational Research*, 280(2):428–440.

Nesterov, Y. (2003). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Pereira, M., Wang, Z., Exarchos, I., and Theodorou, E. A. (2019). Neural network architectures for stochastic control using the nonlinear feynman-kac lemma. *arXiv:1902.03986*.

Pham, H. (2009). *Continuous-Time Stochastic Control and Optimization with Financial Applications*. Springer.

Possamaï, D., Tan, X., and Zhou, C. (2018). Stochastic control for a class of nonlinear kernels and applications. *The Annals of Probability*, 46(1):551–603.

Reisinger, C., Stockinger, W., and Zhang, Y. (2022). Linear convergence of a policy gradient method for finite horizon continuous time stochastic control problems. *arXiv preprint arXiv:2203.11758*.

Sun, J., Li, X., and Yong, J. (2016). Open-loop and closed-loop solvabilities for stochastic linear quadratic optimal control problems. *SIAM Journal on Control and Optimization*, 54(5):2274–2308.

Wang, H., Zariphopoulou, T., and Zhou, X. Y. (2020). Reinforcement learning in continuous time and space: A stochastic control approach. *The Journal of Machine Learning Research*, 21(1):8145–8178.

Wang, W., Han, J., Yang, Z., and Wang, Z. (2021). Global convergence of policy gradient for linear-quadratic mean-field control/game in continuous time. In *International Conference on Machine Learning*, pages 10772–10782. PMLR.

Wang, Z., Pereira, M. A., and Theodorou, E. A. (2019). Deep 2FBSDEs for systems with control multiplicative noise. *arXiv:1906.04762*.

Wu, W., Gao, J., Lu, J.-G., and Li, X. (2020). On continuous-time constrained stochastic linear–quadratic control. *Automatica*, 114:108809.

Zhang, J. (2017). *Backward Stochastic Differential Equations*. Springer.

Zhou, M. and Lu, J. (2023). A policy gradient framework for stochastic optimal control problems with global convergence guarantee. *arXiv preprint arXiv:2302.05816*.

Appendix A

Additional Material

A.1 Auxiliary Results

In this section we present results that are used throughout the convergence analysis.

Theorem A.1.1 (Young Inequality). *Let $p, q \in [1, \infty)$ such that $\frac{1}{p} + \frac{1}{q} = 1$. Given $a, b \in \mathbb{R}$ and $\epsilon > 0$ we have*

$$ab \leq \frac{\epsilon a^q}{q} + \frac{b^p}{\epsilon^{p-1} p}.$$

Theorem A.1.2 (Holder Inequality). *Let (S, σ, μ) be a measure space and let $p, q \in [1, \infty]$ with $\frac{1}{p} + \frac{1}{q} = 1$. Then for all measurable $f \in \mathcal{L}^p(S)$ and $g \in \mathcal{L}^q(S)$*

$$\|fg\|_1 \leq \|f\|_p \|g\|_q.$$

Theorem A.1.3 (Burkholder-Davis-Gundy Inequality). *For all $p \geq 0$ there exist positive constants K_p, k_p such that, for all $t \geq 0$ and continuous local martingales $M = (M_s)_{s \in [0,t]}$ we have*

$$k_p \mathbb{E}\left[\langle M \rangle_t^{\frac{p}{2}}\right] \leq \mathbb{E}\left[\sup_{0 \leq s \leq t} |M_t|^p\right] \leq K_p \mathbb{E}\left[\langle M \rangle_t^{\frac{p}{2}}\right],$$

where $\langle M \rangle$ is the quadratic variation of M.

Theorem A.1.4 (BSDE functional representation, Zhang (2017) Theorem 5.1.3). *Let $b\colon [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$, $\sigma\colon [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$, $f\colon [0,T] \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ and $g\colon \mathbb{R}^n \to \mathbb{R}^n$ be deterministic* *uniformly Lipschitz functions, with* $b(\cdot,0)$, $\sigma(\cdot,0)$, *and* $f(\cdot,0,0,0)$ *uniformly bounded. Suppose further that* b, σ *and* f *are uniformly Holder* $\left(\frac{1}{2}\right)$ *continuous in* t. *Fix* $t \in [0,T]$ *and consider the BSDE, for any* $x \in \mathbb{R}^n$, $s \in [t,T]$,

$$X_s^{t,x} = x + \int_t^s b\left(r, X_r^{t,x}\right) dr + \int_t^s \sigma\left(r, X_r^{t,x}\right) dW_r,$$
$$Y_s^{t,x} = g\left(X_T^{t,x}\right) + \int_s^T f_t\left(X_r^{t,x}, Y_r^{t,x}, Z_r^{t,x}\right) dr + \int_s^T Z_r^{t,x} dW_r.$$

Then there exists a version $(Y^{t,x}, Z^{t,x})$ *for each* x *such that the mapping*

$$(\omega, s, x) \mapsto \left(Y_s^{t,x}(\omega), Z_s^{t,x}(\omega)\right), \qquad (\omega, s, x) \in \Omega \times [t,T] \times \mathbb{R}^n,$$

is $\mathcal{F}_t$*-progressively measurable.*

Theorem A.1.5 (Blumenthal's 0-1 Law)**.** *Suppose that* $X = (X_t)_{t\geq 0}$ *is an adapted right continuous Feller process on a probability space* $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, \mathbb{P})$ *such that* X_0 *is constant with probability one. Let* $\mathcal{F}_t^X := \sigma(X_s; s \leq t)$ *and* $\mathcal{F}_{t^+}^X := \bigcap_{s>t} \mathcal{F}_s^X$. *Then any event in* $\Lambda \in \mathcal{F}_{0+}^X$ *has either* $\mathbb{P}(\Lambda) = 0$ *or* $\mathbb{P}(\Lambda) = 1$.

Theorem A.1.6 (Stationary point characterisation, Reisinger et al. (2022) Theorem 3.10)**.** *Let* $(X, ||\cdot||_X)$ *be a Hilbert space,* $F \colon X \to \mathbb{R}$ *be a differentiable function and* $\tilde{F} \colon X \to \mathbb{R} \cup \{\infty\}$ *be a proper, lower semicontinuous and convex function. Let* $x^* \in X$ *such that* $\tilde{F}(x^*) < \infty$. *Then* x^* *is a stationary point of* $F + \tilde{F}$ *if and only if for all* $\tau > 0$

$$x^* = \operatorname{prox}_{\tau\tilde{F}}\left(x^* - \tau\nabla F(x^*)\right),$$

where, for all $x \in X$, $\operatorname{prox}_{\tau\tilde{F}}(x) = \arg\min_{z\in X}\left(\frac{1}{2}||z-x||_X + \tau\tilde{F}(z)\right)$.

Theorem A.1.7 (Zhang (2017) Theorem 3.4.3)**.** *Consider the following* n*-dimensional SDE.*

$$X_t = \eta + \int_0^t b_s(X_s)\, ds + \int_0^t \sigma_s(X_s)\, dW_s, \qquad 0 \leq t \leq T,$$

where $\eta \in \mathcal{L}^0(\mathbb{R}^n)$, *and the functions* $b \colon \Omega \times [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ *and* $\sigma \colon \Omega \times [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ *are measurable. Let* $p \geq 2$. *Assume the following:*

1. b *and* σ *are uniformly Lipschitz in* x, *with Lipschitz coefficient* L.

2. *$\eta \in \mathcal{L}^2(\mathbb{R}^n)$, and the processes $b^0 := b(0)$ and $\sigma^0 := \sigma(0)$ satisfy*

$$\mathbb{E}\left[\left(\int_0^T |b_t^0| dt\right)^p + \left(\int_0^T |\sigma_t^0|^2 dt\right)^{\frac{p}{2}}\right] < \infty.$$

Then there exists $C = C(p, T, L, n) \geq 0$ such that

$$\mathbb{E}\left[\sup_{t\in[0,T]} |X_t|^p\right] \leq C\mathbb{E}\left[|\eta|^p + \left(\int_0^T |b_t^0| dt\right)^p + \left(\int_0^T |\sigma_t^0|^2 dt\right)^{\frac{p}{2}}\right].$$

Theorem A.1.8 (Rademacher's Theorem)**.** *Let $E \subset \mathbb{R}^m$ be open, and $f\colon E \to \mathbb{R}^n$ Lipschitz. Then f is differentiable almost everywhere on E.*

The next two Theorems concern estimates on the $\mathcal{S}^2$ and $\mathcal{H}^2$ norms of SDE and BSDE solutions.

Theorem A.1.9 (Zhang (2017), Theorem 3.2.2)**.** *Consider the following n-dimensional SDE.*

$$X_t = \eta + \int_0^t b_s(X_s) ds + \int_0^t \sigma_s(X_s) dW_s, \qquad 0 \leq t \leq T,$$

where η and the functions $b\colon \Omega \times [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ and $\sigma\colon \Omega \times [0,T] \times \mathbb{R}^n \to \mathbb{R}^n$ are measurable. Assume the following:

1. *b_s and σ_s are uniformly Lipschitz in x for all $s \in [0,T]$ with Lipschitz coefficients L_b and L_σ.*

2. *$\eta \in \mathcal{L}^2(\mathbb{R}^n)$, and the processes $b^0 := b(0)$ and $\sigma^0 := \sigma(0)$ satisfy*

$$\mathbb{E}\left[\left(\int_0^T |b_t^0| dt\right)^2 + \left(\int_0^T |\sigma_t^0|^2 dt\right)\right] < \infty.$$

Then there exists $C = C_{SDE}(L_b, L_\sigma) \geq 0$ such that

$$\mathbb{E}\left[\sup_{t\in[0,T]} |X_t|^2\right] \leq C\mathbb{E}\left[|\eta|^2 + \left(\int_0^T |b_t^0| dt\right)^2 + \left(\int_0^T |\sigma_t^0|^2 dt\right)\right].$$

Theorem A.1.10 (Zhang (2017), Theorem 4.2.1)**.** *Consider the following BSDE.*

$$Y_t = \xi + \int_t^T f_s(Y_s, Z_s) ds - \int_t^T Z_s dW_s,$$

where $Y \in \mathcal{S}^2(\mathbb{R}^n)$, $Z \in \mathcal{H}^2(\mathbb{R}^n)$ *and* $f\colon \Omega \times [0,T] \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ *is measurable. Assume the following:*

1. f *is uniformly Lipschitz in* (y, z) *with Lipschitz constants* L_Y *and* L_Z *respectively.*
2. $\xi \in \mathcal{L}^2(\mathbb{R}^n)$, *and* $f^0 := f(0,0)$ *satisfies*

$$\mathbb{E}\left[\left(\int_0^T |f_t^0| dt\right)^2\right] < \infty.$$

Then there exists $C = C_{BSDE}(L_Y, L_Z) \geq 0$ *such that*

$$\mathbb{E}\left[\sup_{t\in[0,T]} |Y_t|^2 + \int_0^T |Z_t|^2 dt\right] \leq C\mathbb{E}\left[|\xi|^2 + \left(\int_0^T |f_t^0| dt\right)^2\right].$$

The following is a consequence of (Nesterov, 2003, Theorem 2.1.12), and we provide an adapted proof for this sepcial case.

Lemma A.1.11. *Define* $f\colon \mathbb{R}^m \to \mathbb{R}$ *by* $f(u) = \frac{1}{2}u^\top A_t u$, *for some* $A \in \mathcal{C}^0([0,T];\mathbb{R}^{m\times m})$ *satisfying* $A_t succ\mu > 0$ *for all* t. *Then for any* $u_1, u_2 \in \mathbb{R}^m$

$$\begin{aligned}&(u_1 - u_2)^\top (\nabla f(u_1) - \nabla f(u_2)) \\ &\geq \frac{\mu\|A\|_\infty}{\mu + \|A\|_\infty}|u_1 - u_2|^2 + \frac{1}{\mu + \|A\|_\infty}|\nabla f(u_1) - \nabla f(u_2)|^2.\end{aligned} \tag{A.1}$$

Proof. We have $\nabla f(u) = A_t u$, and f is μ-strongly convex in u and ∇f is $\|A\|_\infty$-Lipschitz in u. Define a convex function $g\colon \mathbb{R}^m \to \mathbb{R}$ by

$$g(u) = f(u) - \frac{1}{2}\mu|u|^2, \qquad u \in \mathbb{R}^m.$$

Then $\nabla g(u) = (\nabla f(u) - \mu\mathbb{1}_m)u$ for $u \in \mathbb{R}^m$, where $\mathbb{1}_m \in \mathbb{R}^m$ is a vector of ones. Note that, as $A_t \succ \mu 1_m$, for any $u \in \mathbb{R}^m$ we have

$$\mu|u|^2 \leq u^\top A_t u \leq \|A\|_\infty |u|^2.$$

Hence $\mu \leq \|A\|_\infty$. In particular, ∇g is $(\|A\|_\infty - \mu)$ - Lipschitz. If $\|A\|_\infty - \mu = 0$, then we must have $A = \mathbb{1}_m\mu$ and the result follows immediately from substitution, so suppose that $\|A\|_\infty - \mu > 0$. As ∇g is Lipschitz, by (Nesterov, 2003, Theorem 2.1.5) an equivalent characterisation for the convexity of g is that for any $u_1, u_2 \in \mathbb{R}^m$

$$\left(\nabla g\left(u_1\right) - \nabla g\left(u_2\right)\right)^\top \left(u_1 - u_2\right) \geq \frac{1}{\|A\|_\infty - \mu} \left|\nabla g\left(u_1\right) - \nabla g\left(u_2\right)\right|^2.$$

By plugging in the definition of g and rearranging, we see that this condition is equivalent to, for any $u_1, u_2 \in \mathbb{R}^m$,

$$\begin{aligned}
&\left(1 - \frac{2\mu}{\|A\|_\infty - \mu}\right) \left(u_1 - u_2\right)^\top \left(\nabla f\left(u_1\right) - \nabla f\left(u_2\right)\right) \\
&\geq \left(\frac{\mu^2}{\|A\|_\infty - \mu} - \mu\right) \left|u_1 - u_2\right|^2 + \frac{1}{\|A\|_\infty - \mu} \left|\nabla f\left(u_1\right) - \nabla f\left(u_2\right)\right|^2,
\end{aligned}$$

from which (A.1) follows. □

We proceed to show the Malliavin differentiability of $\left(X_s^{t,x,\phi}, Y_s^{t,x,\phi}, Z_s^{t,x,\phi}\right)$ for any $t \in [0,T]$, $x \in \mathbb{R}^n$, $\phi \in \mathcal{V}$ and $s \in [t,T]$. Since the function ϕ is non-differentiable, we will use its generalised derivative, which is given as the weak limit of the derivative of mollifiers of ϕ. Note that, by Rademancher's Theorem (Theorem A.1.8), for any $s \in [0,T]$ the function ϕ_s is differentiable almost everywhere, and this derivative coincides with $\delta\phi_s$. Furthermore, as ϕ is Lipschitz, for any $s \in [0,T]$ we have $\left|\delta\phi_s\left(x\right)\right| \leq [\phi]_1$ almost everywhere.

Proof of 3.4.2. We want to apply Theorem 1.6.3 at $X = X^{t,x,\phi}$. Given the definition (3.10) of the SDE, we have , for $s \in [t,T]$, and $\tilde{x} \in \mathbb{R}^n$

$$\begin{aligned}
\eta &= x, \\
b_s\left(\tilde{x}\right) &= A_s\tilde{x} + B_s\phi_s\left(\tilde{x}\right), \\
\sigma_s\left(\tilde{x}\right) &= C_s\tilde{x} + D_s\phi_s\left(\tilde{x}\right).
\end{aligned}$$

Since ϕ is Lipschitz and integrable, the conditions of Theorem 1.6.3 are satisfied. Since ϕ is not assumed to be differentiable, we take the generalised derivative, denoted by $\delta\phi$. The result follows by Theorem 1.6.3 and we can write, for $t \leq s \leq T$,

$$\begin{aligned}\mathcal{D}_t X_s^{t,x,\phi} &= C_t x + D_t \phi_t(x) \\ &+ \int_t^s \left(A_r + B_r \delta\phi_r\left(X_r^{t,x,\phi}\right)\right) \mathcal{D}_t X_r^{t,x,\phi} dr \\ &+ \int_t^s \left(C_r + D_r \delta\phi_r\left(X_r^{t,x,\phi}\right)\right) \mathcal{D}_s X_r^{t,x,\phi} dW_r.\end{aligned} \tag{A.2}$$

□

Proof of 3.4.3. We wish to apply Theorem 1.6.4 to $\left(Y^{t,x,\phi}, Z^{t,x,\phi}\right)$. Using the definition (3.10) of the BSDE, we have, for $s \in [t,T]$ and $y, z \in \mathbb{R}^n$,

$$\begin{aligned}\xi &= G X_T^{t,x,\phi}, \\ f_s(y,z) &= A_s^\top y + C_s^\top z + S_s^\top \phi_s\left(X_s^{t,x,\phi}\right) + Q_s X_s^{t,x,\phi}.\end{aligned}$$

Since $X_s^{t,x,\phi} \in \mathbb{D}^{1,2}(\mathbb{R}^n)$, and using the chain rule we have, for $s \in [t,T]$ and $y, z \in \mathbb{R}^n$,

$$\begin{aligned}\mathcal{D}_t \xi &= G \mathcal{D}_t X_T^{t,x,\phi}, \\ \mathcal{D}_t f_s(y,z) &= \left(S_s^\top \delta\phi_s\left(X_s^{t,x,\phi}\right) + Q_s\right) \mathcal{D}_t X_s^{t,x,\phi}.\end{aligned}$$

We want to show that $\mathcal{D}_t \xi$ and $\mathcal{D}_t f_s(y,z)$ have sufficient integrability properties. Firstly, as ϕ is Lipschitz, $\left|\delta\phi_s\left(X_s^{t,x,\phi}\right)\right| \leq [\phi]_1$ almost surely for all $s \in [t,T]$. We therefore only need integrability estimates of $\mathcal{D}_t X_s^{t,x,\phi}$. Let us apply Theorem A.1.7 to $\left(\mathcal{D}_t X_s^{t,x,\phi}\right)_{s\in[t,T]}$. Using the definition (A.2), we have, for $s \in [t,T]$ and $\tilde{x} \in \mathbb{R}^n$,

$$\begin{aligned}\eta &= C_t x + D_t \phi_t(x), \\ b_s(\tilde{x}) &= \left(A_s + B_s \delta\phi_s\left(X_s^{t,x,\phi}\right)\right) \tilde{x}, \\ \sigma_s(\tilde{x}) &= \left(C_s + D_s \delta\phi_s\left(X_s^{t,x,\phi}\right)\right) \tilde{x}.\end{aligned}$$

As $\delta\phi_s\left(X_s^{t,x,\phi}\right)$ is bounded a.s., the conditions of Theorem A.1.7 are satisfied, hence for any $p \geq 2$, there exists $K_p \geq 0$ such that

$$\mathbb{E}\left[\left|\mathcal{D}_t X_s^{t,x,\phi}\right|^p\right] \leq K_p \mathbb{E}\left[|\eta|^p\right] \leq K_p\left(\|C\|_\infty + \|D\|_\infty [\phi]_1\right)^p |x|^p < \infty. \tag{A.3}$$

We can now verify the conditions of Theorem 1.6.4 for $\left(Y^{t,x,\phi}, Z^{t,x,\phi}\right)$.

1. As $X^{t,x,\phi} \in \mathcal{S}^p\left(t,T;\mathbb{R}^n\right)$ for all $p \geq 2$, this condition is satisfied.

2. $f_s\left(\omega,\cdot,\cdot\right)$ is linear for any $(s,\omega) \in [t,T] \times \Omega$, hence continuously differentiable.

3. $f_s\left(y,z\right) \in \mathbb{D}^{1,2}\left(\mathbb{R}^n\right)$ for any $(s,y,z) \in [t,T] \times \mathbb{R}^n \times \mathbb{R}^n$ as $X_s^{t,x,\phi} \in \mathbb{D}^{1,2}\left(\mathbb{R}^n\right)$ and ϕ is Lipschitz, and we have
$$\mathbb{E}\left[\int_t^T |f_s^0|^4 ds\right] = \mathbb{E}\left[\int_t^T |S_s^\top \phi_s\left(X_s^{t,x,\phi}\right) + Q_s X_s^{t,x,\phi}|^4 ds\right] < \infty,$$
as $X^{t,x,\phi} \in \mathcal{S}^4\left(t,T;\mathbb{R}^n\right)$. Using the chain rule, we have
$$\left(\mathcal{D}_t f\right)_s\left(y,z\right) = \left(S_s^\top \delta\phi_s\left(X_s^{t,x,\phi}\right) + Q_s\right)\mathcal{D}_t X_s^{t,x,\phi}, \qquad t \leq s \leq T.$$

4. We have, for any $r \in [t,T]$
$$\mathbb{E}\left[\int_t^T |\mathcal{D}_s\xi|^2 ds\right] = \mathbb{E}\left[\int_t^T |G\mathcal{D}_s X_T^{t,x,\phi}|^2 ds\right] < \infty,$$
$$\mathbb{E}\left[\int_t^T |\left(\mathcal{D}_s f\right)_r\left(Y_r^{t,x,\phi}, Z_r^{t,x,\phi}\right)|^2 ds\right] = \mathbb{E}\left[\int_t^T |\left(S_r^\top \delta\phi_r\left(X_r^{t,x,\phi}\right) + Q_r\right)\mathcal{D}_s X_r^{t,x,\phi}|^2 ds\right] < \infty.$$

5. Since $\left(\mathcal{D}_s f\right)_t\left(y,z\right)$ is independent of $(y,z) \in \mathbb{R}^n \times \mathbb{R}^n$ for any $s \in [t,T]$, this condition is automatically satisfied with $K_{s,t} = 0$.

Therefore, by Theorem 1.6.4 we have $Y_s^{t,x,\phi}, Z_s^{t,x,\phi} \in \mathbb{D}^{1,2}\left(\mathbb{R}^n\right)$ for all $s \in [t,T]$ and
$$\begin{aligned}
\mathcal{D}_t Y_s^{t,x,\phi} &= G\mathcal{D}_t X_T^{t,x,\phi} \\
&\quad + \int_t^T \left(A_s^\top \mathcal{D}_t Y_s^{t,x,\phi} + C_s^\top \mathcal{D}_t Z_s^{t,x,\phi} + \left(S_s^\top \delta\phi_s\left(X_s^{t,x,\phi}\right) + Q_s\right)\mathcal{D}_t X_s^{t,x,\phi}\right) ds \\
&\quad - \int_t^T \mathcal{D}_t Z_s^{t,x,\phi} dW_s.
\end{aligned} \tag{A.4}$$

□

A.2 List of Constants

For reference, we provide the definitions of all constants used in Chapter 3.

Constants for Section 3.4

$$
\begin{aligned}
\tilde{c}_1 &= C_{\mathrm{BSDE}}\left(\|A\|_\infty, \|C\|_\infty\right), \\
\tilde{c}_2 &= C_{\mathrm{SDE}}\left(\|A\|_\infty, \|C\|_\infty\right), \\
\tilde{c}_3 &= \frac{1}{2}\left(\|B\|_\infty^2 T + \|D\|_\infty^2\right), \\
\tilde{c}_4 &= |G| + \|Q\|_\infty T, \\
\tilde{c}_5 &= \|S\|_\infty, \\
\tilde{c}_6 &= \sqrt{\tilde{c}_1 \tilde{c}_2}\tilde{c}_4, \\
\tilde{c}_7 &= \sqrt{\tilde{c}_1 \tilde{c}_2}\tilde{c}_5 T, \\
\tilde{c}_8 &= \tilde{c}_6\|C\|_\infty + \frac{1}{2}\left(\tilde{c}_6\|C\|_\infty + \tilde{c}_7\|D\|_\infty\right)^2, \\
\tilde{c}_9 &= \tilde{c}_7\|D\|_\infty + \frac{1}{2}\left(\tilde{c}_6\|C\|_\infty + \tilde{c}_7\|D\|_\infty\right)^2, \\
\tilde{c}_{10} &= \max\left(T\tilde{c}_3[\phi^0]_1, \frac{T\tilde{c}_3\|S\|_\infty + 3c_0}{c_0}\right), \\
\tilde{c}_{11} &= \|S\|_\infty + \left(\|B\|_\infty \tilde{c}_6 e^{\tilde{c}_{10}^2} + \|D\|_\infty \tilde{c}_7\right) e^{\tilde{c}_{10}^2} + \|D\|_\infty C_L\left(\frac{\tilde{c}_{10}}{T\tilde{c}_3}\right), \\
M\left(\phi^0\right) &= [\phi^0]_1 + \frac{\tilde{c}_{11}}{c_0}, \\
\hat{c} &= (1 - 2\tau c_0) + \tau\left(\|B\|_\infty C_Y\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right) + \tau\|D\|_\infty C_Z\left(C_K, M\left(\phi^0\right), M\left(\phi^0\right)\right)\right)
\end{aligned}
$$

Constants for Section 3.5

$$
\begin{aligned}
c_1 &= 2\|A\|_\infty + \|C\|_\infty^2, \\
c_2 &= \|B\|_\infty + \|C\|_\infty\|D\|_\infty, \\
c_3 &= |G| + T\|Q\|_\infty, \\
c_4 &= T\|S\|_\infty, \\
c_5 &= c_3\|C\|_\infty + \frac{1}{2}\left(c_3\|C\|_\infty + c_4\|D\|_\infty\right)^2, \\
c_6 &= c_4\|D\|_\infty + \frac{1}{2}\left(c_3\|C\|_\infty + c_4\|D\|_\infty\right)^2, \\
c_7 &= \max\left(c_2T[\phi^0]_1, \frac{c_2T\|S\|_\infty + c_0}{c_0}\right), \\
c_8 &= \|S\|_\infty + \left(\|B\|_\infty c_3 + \|D\|_\infty c_5 e^{c_1T+c_7}\right), \\
c_9 &= Te^{\|A\|_\infty T}\left(c_2C_a\left(c_\alpha\right) + \|S\|_\infty\right), \\
c_{10} &= \left(1 - 2\tau c_0\right) + \tau\left(c_9 + \|D\|_\infty^2 C_a\left(c_\alpha\right)\right), \\
c_\alpha &= [\phi^0]_1 + \frac{c_8}{c_0}, \\
c_\beta &= \|\beta^0\|_\infty, \\
\hat{c} &= \left(1 - 2\tau c_0\right) + \tau\|B\|_\infty C_Y\left(c_\beta, c_\alpha, c_\alpha\right) + \tau\|D\|_\infty C_Z\left(c_\beta, c_\alpha, c_\alpha\right).
\end{aligned}
$$