

# Efficient Probabilistic Pricing Algorithms for Multiple Exercise Options

by

Nicolas Essis-Breton

A Thesis

In the Department

of

Mathematics and Statistics

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor in Philosophy (Mathematics)

at

Concordia University

Montreal, Quebec, Canada

May 2020

© Nicolas Essis-Breton, 2020

**CONCORDIA UNIVERSITY**  
**SCHOOL OF GRADUATE STUDIES**

This is to certify that the thesis prepared

By: Nicolas Breton Essis

Entitled: Efficient Probabilistic Pricing Algorithms for Multiple Exercise Options

and submitted in partial fulfillment of the requirements for the degree of

Doctor Of Philosophy (Mathematics)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

\_\_\_\_\_Chair  
Dr. Sébastien Caquard

\_\_\_\_\_External Examiner  
Dr. Luc Devroye

\_\_\_\_\_External to Program  
Dr. Ravi Mateti

\_\_\_\_\_Examiner  
Dr. Simone Brugiapaglia

\_\_\_\_\_Examiner  
Dr. Cody Hyndman

\_\_\_\_\_Thesis Supervisor  
Dr. Patrice Gaillardetz

Approved by

\_\_\_\_\_  
Dr. Galia Dafni, Graduate Program Director

May 1, 2020

\_\_\_\_\_  
Dr. André Roy, Dean  
Faculty of Arts and Science

# Abstract

## Efficient probabilistic pricing algorithms for multiple exercise options

Nicolas Essis-Breton, Ph.D.

Concordia University, 2020

This thesis presents efficient probabilistic algorithms for the pricing of multiple exercise options. The algorithms handle the large class of of constrained multiple exercise American options. Two algorithms are presented. Single Pass Lookahead Search provides a lower estimate of an option price, while Nearest-Neighbor Martingale provides an upper estimate. The convergence of the algorithms is proved through a Vapnik-Chernovenkis dimension analysis. Their efficiency is illustrated on several examples, including a swing option with four constraints, and a passport option with 16 constraints. The algorithms are also applied to the pricing of equity-linked product offering a reinvestment option.

# Contribuon of Authors

Nicolas Essis-Breton conceived the presented ideas, developed the theory, performed the computation, and wrote the thesis and the journal articles. Patrice Gaillardetz supervised the findings of this work.

# Table of Contents

1	Introduction	1
2	Fast Lower and Upper Estimates for the Price of Constrained Multiple Exercise American Options by Single Pass Lookahead Search and Nearest-Neighbor Martingale	5
3	Valuation of the Reinvestment Option in Equity-Indexed Annuities	96
4	Conclusion and Future Work	118
	References	120

# Chapter 1

## Introduction

The fundamental object of study of this thesis is stochastic control. Stochastic control is a sub-field of control theory where one tries to influence the evolution of a system driven by random noise. Influencing the system is done through a control variable for which one can choose the value at any time. A famous modern example of such stochastic control is the placement of ads by Google and Facebook. In this particular example, the problem is to decide which ad should be presented to a customer so that the customer clicks on the ad. The random noise comes from the customer response to the presented ads. Other modern examples of stochastic control are self-driving car and chatbots. More classical examples are board games such as chess or Go. As these examples suggest, stochastic control is an ubiquitous topic studied in many fields, and a rich literature is available on the topic. In the field of finance, stochastic control is studied in many sub-field such as optimal asset allocation and option pricing.

For stochastic control in finance, the random noise comes mainly from the price of an asset, for example a stock price, and the control variable is the quantity of asset purchased. The randomness of asset prices comes from market participants whose willingness to buy an asset at a certain price can only be observed through actual transactions. The goal of the controller is usually to maximize a payoff that is contingent on an asset price. For example, for an American call, the payoff is the difference between the stock price and a reference price called the strike. The controller can only claim once this difference. The stochastic control problem is then to choose the best time to claim

the difference.

While stochastic control is an ubiquitous topic, research on computer solution of such problems concentrate on dynamic programming. A recent example is the breakthrough of Google DeepMind which found a dynamic programming algorithm for the game of Go. The game of Go has long been taught as untractable for computers because among other reasons, the search space size in the game is larger than the number of atoms in the universe. This breakthrough resides in combining machine learning methods with classical dynamic programming methods. This example demonstrates a trend in the literature on stochastic control where most algorithmic solution are refinement of classical dynamic programming algorithm. In the financial literature, this trend is also present. In particular, in the literature on multiple exercise options, most algorithms are based on dynamic programming.

While dynamic programming solutions to control problems are prevalent, this solution approach comes with many well-known challenges. One such well-known challenge is the curse of dimensionality, where enormous computing power is necessary to solve apparently simple problems. For example, for the game of go, the computing power used by Google DeepMind is equivalent to 10,000 years of computing on a personal laptop. Finance is renowned for the fast pace at which transactions take place, and requiring such an amount of computing for every control decision on a transaction is unpractical. This situation points to a lack of research on the passage from a stochastic control problem to its algorithmic solution. We call this passage the algorithmic translation problem, and the goal of this thesis is to propose a new solution to this problem.

The innovation of the thesis is to approach the algorithmic translation problem in the same way that predictive algorithm are designed in machine learning. This standpoint starts with the recognition that any convergence guarantee on the obtention of a predictor is limited to a particular choice of predictor family. Within this predictor family, the convergence guarantees are universal to any distribution of the data. By considering first the predictor family, rather than the algorithmic solution itself, machine learning focuses on building a large bestiary of predictor family. This approach has many advantages, one of them is that each predictor family can directly be mapped to an algorithmic solution, so that having a large choice of predictor families implies that a large choice of algorithmic solutions is available.

In finance, and in particular in option pricing, the machine learning approach to the algorithmic translation problem demands to identify families of control that possess both good convergence guarantees and tractable algorithmic solution. Such good families are exhibited in the thesis and their tractability is demonstrated on several examples. While the results of the thesis focus on option pricing, the results may be applied to the solution of stochastic control problem in other fields.

This thesis is formed by two articles. The first article presents efficient probabilistic algorithms for the pricing of multiple exercise options. The second article applies these algorithms to the reinvestment option found in equity-linked annuity.

A fully general pricing algorithm for multiple exercise options is a long standing challenge in derivative pricing. Since the pioneering work of Black and Scholes (1973) and Merton (1973), many pricing algorithms have appeared for single exercise options, and many of these algorithms have been extended to multiple exercise options. However, conforming to their single exercise origins, all these extensions are rooted in dynamic programming techniques, whether they are finite difference based or simulation based. Further, these extensions were not designed to provide a fully general pricing algorithm, and their applications are often tedious. For example, none of these extensions provide a method to enumerate the constraints state, a fundamental requirement to any dynamic programming based algorithm.

This work presents two algorithms, Single Pass Lookahead Search and Nearest-Neighbor Martingale, that were designed with this pricing challenge in mind. The algorithms combine methods from artificial intelligence, option pricing theory, and numerical optimization. In particular, convergence guarantees for the algorithms are provided through a Vapnik-Chernovenkis analysis. Further, the algorithms work on a mathematical programming formulation of an option pricing problem: an intuitive formulation that is easy to obtain for any option. The efficiency of the algorithms is demonstrated on options that are beyond the capabilities of existing pricing algorithms. A swing option with four constraints, and a passport option with 16 constraints.

The presented algorithms are fast, accurate and versatile. Single Pass Lookahead Search provides a



lower estimate of the option price through the lookahead search technique of artificial intelligence, while Nearest-Neighbor Martingale solves the dual pricing problem with a nearest-neighbor basis and a Frank-Wolfe iteration. To further illustrate the versatility of the algorithms, the algorithms are applied to the pricing of equity-linked product offering a reinvestment option.

From an option pricing standpoint, the algorithms are presented in the context of a complete market, where market participants are rationals and options prices are established by no-arbitrage under the given risk-neutral measure. This context provides a basis for extension to other contexts. For example, in an incomplete market context, an option price is determined by considering both the optimal exercise strategy and the cost of a self-financing hedging strategy. Indeed, market incompleteness may create a disparity between the option payoffs and how replicable these payoffs are, so that both the option holder and the option issuer perspectives need to be considered. In such a context, the presented algorithms can be applied twice, once for the holder perspective, and a second time for the issuer perspective. An other example is the context of model calibration. In this context, market prices are given and one tries to recover the underlying risk measure. In such a context, the algorithms can be used to test if a given measure gives back the observed market price. Depending on the family of tested measure, a grid search or a bisection combined with the algorithms can pinpoint the exact measure that gives the market prices.

## Chapter 2

Fast Lower and Upper Estimates for the  
Price of Constrained Multiple Exercise  
American Options by Single Pass  
Lookahead Search and Nearest-Neighbor  
Martingale

# Fast Lower and Upper Estimates for the Price of Constrained Multiple Exercise American Options by Single Pass Lookahead Search and Nearest-Neighbor Martingale

Nicolas Essis-Breton\*and Patrice Gaillardetz

Concordia University

March 2020

---

## Abstract

This article presents fast lower and upper estimates for a large class of options: the class of constrained multiple exercise American options. Typical options in this class are swing options with volume and timing constraints, and passport options with multiple lookback rights. Such options are widely traded on energy and financial markets, despite the fact that there is currently no method to price them. These options are intractable for exact pricing algorithms, and there is no approximate pricing algorithm that can incorporate all the complex features offered in these options. The main contribution of this article is to propose two algorithms that fill this gap. The first algorithm uses the artificial intelligence method of lookahead search with three novelties: 1) we use the lookahead search in a direct scheme, rather than in an iterative scheme: neither a value function, nor a policy function are learned; 2) we exhibit an approximation of the lookahead search problem in term of a conditional nearest-neighbor basis for the stock path, with convergence guarantees provided through a Vapnik-Chernovenkis

---

\*Address of correspondence to Nicolas Essis-Breton, Department of Mathematics and Statistics at Concordia University, Montreal, Quebec H3G 1M8, Canada, or e-mail: nicolasessisbreton@gmail.com.

dimension analysis and a convergence test based on an energy distance for the filtration. 3) we solve the lookahead search problem by mixed-integer programming rather than by Q-learning. We found that these three novelties are keys for fast lower estimate of the considered option class. We call the resulting algorithm *Single Pass Lookahead Search*. The second algorithm uses the dual approach to option pricing with three novelties: 1) we exhibit a martingale basis formed by a nearest-neighbor basis for the stock path, for which enforcing the martingality condition is practical, and with a convergence test provided by an energy distance for the filtration; 2) through a Vapnik-Chernovenkis dimension analysis, we show that obtaining a solution of the dual problem is much easier than obtaining a dual martingale that generalizes to any out-of-sample test; 3) we solve the resulting continuous non-linear dual problem with a Frank-Wolfe method. We found that these three novelties are key for fast upper estimate of the considered option class. We call the resulting algorithm *Nearest-Neighbor Martingale*. Several numerical examples illustrate the approaches including a swing option with four constraints, and a passport option with 16 constraints. The examples show that the proposed approaches are versatile, fast, and require no adjustment when applied to different options. The algorithms can thus simplify the risk management of financial derivatives when multiple pricing methods are used, or enable it when no other pricing method is applicable.

---

Much progress toward general option pricing method was made by extending dynamic programming algorithm initially designed for single exercise option. An early example of such extension is the *forest of tree* method (Thompson 1995). In this method, each tree represents the asset state under a particular constraint state. A change in the asset state induces a move in the asset tree, while a change in the constraint state induces a jump from one tree to another. The method was later improved by several authors (Lari-Lavassani, Simchi, and Ware 2001, Jaillet, Ronn, and Tompaidis 2004, Marshall and Reesor 2011, Bardou, Bouthemy, and Pages 2009, Wilhelm and Winter 2008, Dahlgren and Korn 2005, Dahlgren 2005, Carmona and Touzi 2008, Carmona and Dayanik 2008, Jain and Oosterlee 2012). Another important extension of single exercise pricing method is the extension of the least-square Monte Carlo approach (Longstaff and Schwartz 2001) to multiple exercise option (Meinshausen and Hambly 2004). The extension consists in superposing several sets

of sample paths, each sample path set corresponding to a particular state of the constraints. In essence, this approach could be named *forest of path*. The approach also provides an upper bound on the option price by martingale duality (Rogers 2002, Haugh and Kogan 2004): the optimal martingale is defined from the optimal exercise strategy found for the lower bound. The approach was later improved by several authors (Barrera-Esteve et al. 2006, Bender and Schoenmakers 2006, Aleksandrov and Hambly 2010, Bender 2011a, Bender 2011b, Schoenmakers 2012, Nadarajah, Margot, and Secomandi 2013, and Bender, Schoenmakers, and Zhang 2015). However, the runtime of dynamic programming based algorithm degrades quickly with the option complexity, in particular the constraint complexity. For example, the two previous families of approaches demand one tree (or one set of paths) for each possible constraint state. Further, enumerating the constraint state is often tedious, and these approaches do not address this difficulty. A fast fully general option pricing method that requires no adjustment from one option to another has not yet appeared in the option pricing literature.

Our algorithms, Single Pass Lookahead Search (SPLS) and Nearest-Neighbor Martingale (NNM), differ from the previous approaches in several aspects. First and foremost, they do not use dynamic programming. SPLS uses lookahead searches (Silver et al. 2017, Browne et al. 2012) in a direct scheme where no value function and no policy function are learned; NNM solves the dual martingale problem directly without reusing any information from the lower estimate. Second, they are fast and embarrassingly parallel algorithms. SPLS finds an optimal exercise strategy along a sample path with a single pass of lookahead search, independently of any other sample path. NNM finds an optimal martingale with a fast converging Frank-Wolfe scheme (Frank and Wolfe 1956), and each iteration of the scheme can be speed-up with a parallel computation of the dual option value. Third, they are fully general and applicable with no adjustments to a large class of options. Both SPLS and NNM work on a mathematical programming formulation of an option pricing problem. Such a formulation is intuitive and easy to obtain. To achieve these results, SPLS constructs the decision as step-functions on a nearest-neighbor basis for the stock path. This basis dispenses of building a search tree and makes the lookahead search solvable by mixed-integer programming. NNM constructs the optimal martingale as a conditional nearest-neighbor basis for the stock path.

This basis allows the martingality condition to be directly enforced in the Frank-Wolfe scheme and turns each iteration of the scheme into the solution of a continuous linear program. A conditional nearest-neighbor basis is a nearest-neighbor basis with two levels, one level for the filtration information, and one level for the decision given the filtration information.

In the literature on single exercise American option, some recent articles use the deep learning approach to dynamic programming (Becker, Cheridito, and Jentzen 2019), and solve the dual problem directly with a Wiener expansion basis (Lelong 2016). In contradistinction to these articles, we use simpler tools to provide fast estimates for the harder problem of constrained multiple exercise American options: SPLS uses only lookahead search, and NNM uses only a Frank-Wolfe scheme. Further, our algorithms have greater practical implications. They are fully general and we demonstrate their effectiveness on examples of unmatched complexity in the literature on multiple exercise options. First, a swing option with two multiple exercise American rights, a local and a global volume constraint, an exercise limit constraint, and a refraction constraint. Then, a passport option on three assets with constrained trading rights, several multiple exercise rights to waive the trading constraints, the right to choose between a call payoff or a put payoff, a barrier value that cancel the option, a multiple exercise reset right, and an American lookback right. All these features can be formulated in a mathematical program for our algorithms with five stopping times and 16 constraints.

## 1 Constrained Multiple Exercise American Options

A constrained multiple exercise American option is a stochastic control problem of the form

$$\max_{X, Y} E \left[ \sum_{t=0}^T e^{-rt} f_t(Y_{0:t}, S_{0:t}) X_t \right] \tag{1.1}$$

$$\text{s.t.} \quad g(X, Y, S) = 0 \tag{1.1.1}$$

where  $X$  is a multiple exercise stopping time adapted to the filtration generated by the stock  $S$ ,  $Y$  is an adapted control process,  $f_t$  is the option payoff at time  $t$ ,  $g$  is a vector constraint that holds almost-surely, the expectation is taken under the risk-neutral measure (Black and Scholes 1973, Merton 1973, Harrison and Kreps 1979),  $r$  is the risk-free rate, and  $T$  the maturity. The notation  $X_{0:t} = (X_0, X_1, \dots, X_t)$  denotes the path of a process up to time  $t$ , the option payoff  $f_t$  is in  $\mathbb{R}$ , the stock  $S_t \in \mathbb{R}^d$  is multi-dimensional,  $F_t$  is the filtration generated by the stock at time  $t$ , the multiple stopping time  $X_t : F_t \rightarrow \{0, 1\}$  is a sequence of 0-1 stopping decisions, and the control process  $Y_t : F_t \rightarrow \mathbb{R}^a$  is a sequence of vector-valued decisions. We refer to the stopping time and the control as the *decisions* or the *exercise strategy*.

A typical option in this class is a vanilla swing option that allows a maximum of 10 swings  $X$ , each swing ordering a quantity  $Y$  of the underlying  $S$  up to a global ordering limit of 100. This option can be formulated with the following stochastic control problem

$$\max_{X, Y} E \left[ \sum_{t=0}^T e^{-rt} Y_t S_t X_t \right] \quad (1.2)$$

$$\text{s.t.} \quad \sum_{t=0}^T X_t \leq 10 \quad (1.2.1)$$

$$\sum_{t=0}^T Y_t \in [-100, 100] \quad (1.2.2)$$

$$X_t \in \{0, 1\} \quad t = 0, 1, \dots, T \quad (1.2.3)$$

$$Y_t \in [-1, 1] \quad t = 0, 1, \dots, T \quad (1.2.4)$$

In Methods, we show that the dual problem for this class of options can be formulated with

$$\min_M E \left[ \max_{x, y, g(x, y, S)=0} \sum_{t=0}^T e^{-rt} f_t(y_{0:t}, S_{0:t}) x_t - M_t x_t \right] \quad (1.3)$$

where  $M$  is an adapted martingale,  $x$  is a non-adapted multiple stopping time, and  $y$  is a non-adapted control. The proof is similar to the dual derivation in Rogers (2002). The key ingredient

of our dual formulation is that the option payoff  $f_t$  is multiplied by a stopping time, so that the stopping time can be used as a weight for a penalizing martingale. This in turn motivates our choice of primal form (1.1).

For the previous swing option, the dual is given by

$$\min_M E \left[ \max_{x,y} \sum_{t=0}^T e^{-rt} y_t S_t x_t - M_t x_t \right] \quad (1.4)$$

where the maximum is taken over the non-adapted decisions satisfying the constraints in (1.2).

Since the stopping time in our option class is arbitrary, the following primal forms belong to the class

$$\begin{aligned} & \sum_{t=0}^T f_t(Y_{0:t}, S_{0:t}) \prod_i X_t^i, \\ & \sum_{t=0}^T h_t(X_{0:t}, Y_{0:t}, S_{0:t}), \end{aligned}$$

where  $X_t = (X_t^1, X_t^2, \dots)$  is a vector-valued multiple stopping time, and  $h_t$  is a payoff that depends on both the stopping time and the control. Indeed, for the first form take the weighting stopping as  $\prod_i X_t^i$ . For the second form, redefine the control as the pair  $\tilde{Y} = (X, Y)$ , redefine the payoff as  $\tilde{h}_t = h_t / \prod_i X_t^i$  and take the weighting stopping time as the logical OR  $\bigvee_i X_t^i$ . The dual forms can then be written with

$$\sum_{t=0}^T f_t(Y_{0:t}, S_{0:t}) \prod_i X_t^i - \prod_i X_t^i M_t, \quad (1.5)$$

$$\sum_{t=0}^T h_t(X_{0:t}, Y_{0:t}, S_{0:t}) - \bigvee_i X_t^i M_t. \quad (1.6)$$

The last dual form is particularly useful for option where a weighting stopping time cannot easily be factored from the payoff. The last dual form also shows that our option class is fully general.



## 2 Single Pass Lookahead Search

### 2.1 Algorithm

To present SPLS, we simplify the notation and write the residual payoff of a particular exercise strategy at time  $t$  with

$$J(t, X_{t:T}, S_{t:T}) = \sum_{s=t}^T e^{-rs} f_s(X_{0:s}, S_{0:s}) \quad (2.1)$$

s.t.  $g(X_{t:T}, S) = 0$

where the exercise strategy  $X_{t:T}$  is a vector of stopping times and control processes, and both the payoff and the constraint are conditional on the filtration  $F_t$ . In the payoff  $f_t$  and the constraint  $g$ , the past decisions  $X_{0:t-1}$  and the past stock path  $S_{0:t}$  are given by the filtration  $F_t$ , the future decision  $X_{t:T}$  is the proposed exercise strategy, and the future stock path  $S_{t:T}$  is the sample path given as argument. We simply write  $J(X_{t:T}, S_{t:T})$  when no confusion is possible on the anchoring time  $t$ . The key construction of SPLS is the lookahead operator  $L(X_{t:T}, F_t)$  that finds the optimal strategy  $X_{t:T}^*$  and returns the decision  $X_t^*$  at the anchoring time. This operator can be written with

$$L(X_{t:T}, F_t) = \left( \operatorname{argmax}_{X_{t:T}} E[J(X_{t:T}, S_{t:T} | F_t)] \right)_t \quad (2.2)$$

where  $(X)_t = X_t$  is the selection operator for time  $t$ . Given a sample path  $S$ , SPLS extracts an optimal exercise strategy for the sample path by repetitive application of the lookahead operator. A first decision  $X_0^*$  is extracted, then a second decision  $X_1^*$  is extracted conditional on the previous

decision  $X_0^*$  and the observed path  $S_{0:1}$ , and so on. This repetitive scheme can be written with

$$X_0^* = L(X_{0:T}, \sigma(S_0)), \quad (2.3)$$

$$X_1^* = L(X_{1:T}, \sigma(X_0^*, S_{0:1})), \quad (2.4)$$

$$X_2^* = L(X_{2:T}, \sigma(X_{0:1}^*, S_{0:2})), \quad (2.5)$$

...

$$X_T^* = L(X_T, \sigma(X_{0:T-1}^*, S_{0:T})), \quad (2.6)$$

where  $\sigma(A)$  is the  $\sigma$ -algebra generated by  $A$ . If we denote by  $L_t$  the  $t$ -th application  $L(X_{t:T}, \sigma(X_{0:t-1}^*, S_{0:t}))$ , the option value  $V$  can be written with

$$V = E \left[ J(L_{0:T}, S_{0:T}) \right]. \quad (2.7)$$

To estimate  $V$ , we use several Monte Carlo projections. To present these projections, consider the lookahead operator  $L(X_{t:T}, F_t)$  at time  $t$ , and let  $s$  be anytime between  $t$  and  $T$ . First, we project the decision  $X_s$  onto the space of step-function of the stock path  $S_{0:s}$ . The step-function is constructed with a nearest-neighbor basis for the stock path, where the centroids are sampled randomly. This step-function can be written with

$$\bar{X}_s(S_{0:s}) = \sum_{i=0}^{m_{s-t}} x^{s,i} \llbracket S_{t:s} \sim \bar{S}_{t:s}^{s,i} \rrbracket, \quad (2.8)$$

where  $\bar{S}^{s,i}$  is a sample path used for the basis at time  $s$ ,  $m_{s-t}$  is the number of sample path used at time  $s$ ,  $\llbracket \cdot \rrbracket$  is the indicator function,  $a \sim b$  is the event that  $a$  belongs to the Voronoi cell of  $b$ , and  $x^{s,i} \in \mathbb{R}$  are the basis weights. We write simply  $\bar{X}_s$  when no confusion is possible. Second, we project the lookahead operator with a Monte Carlo average of  $N$  sample. This projected operator can be written with

$$\bar{L}(\bar{X}_{t:T}, F_t) = \left( \operatorname{argmax}_{x^{s,i}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n) \right)_t \quad (2.9)$$

where  $S^n$  is a sample path,  $\bar{X}_{0:t-1}$  and  $S_{0:t}^i$  comes from the filtration  $F_t$ , and the maximum runs over the basis weights  $x^{s,i}$  for time  $s = t, t + 1, \dots, T$  and  $i = 1, 2, \dots, m_{s-t}$ . Note that the projected operator is a deterministic program over the nearest-neighbor basis weight  $x^{s,i}$ . This deterministic program can be solved by any optimization methods, and in particular by mixed-integer programming. The projected optimal exercise strategy can be extracted with

$$\bar{X}_0^* = \bar{L}(\bar{X}_{0:T}, \sigma(S_0^n)), \quad (2.10)$$

$$\bar{X}_1^* = \bar{L}(\bar{X}_{1:T}, \sigma(\bar{X}_0^*, S_{0:1}^n)), \quad (2.11)$$

$$\bar{X}_2^* = \bar{L}(\bar{X}_{2:T}, \sigma(\bar{X}_{0:1}^*, S_{0:2}^n)), \quad (2.12)$$

...

$$\bar{X}_T^* = \bar{L}(\bar{X}_T, \sigma(\bar{X}_{0:T-1}^*, S_{0:T}^n)). \quad (2.13)$$

Third, we project the option value with a Monte Carlo average of  $\tilde{N}$  sample. This projected option value can be written with

$$\bar{V} = \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} J(\bar{L}_{0:T}, S_{0:T}^{\tilde{n}}). \quad (2.14)$$

## 2.2 Convergence

The number of sample  $\tilde{N}$  in the option value, the number of sample  $N$  in the lookahead operator, and the nearest-neighbor basis size  $m = (m_t)_{t=0}^T$  are hyperparameters that need to be tuned so as to maximize the projected option value  $\bar{V}$ . To understand the result of such tuning, we analyze the convergence of the projection scheme. First, we complete a uniform convergence analysis with the Vapnik-Chervonenkis dimension as main tool (Vapnik and Chervonenkis 1971). Then, we complete an energy distance analysis (Székely, Rizzo, et al. 2004, Baringhaus and Franz 2004, Ramdas, Trillos, and Cuturi 2017) The uniform convergence analysis helps to understand the interaction between the hyperparameters, while the energy convergence analysis provides a practical score for any choice of hyperparameters. As the score demands little computational effort, hyperparameters tuning can be done quickly by finding the hyperparameters with the best score.

The uniform convergence of the scheme can be studied from two angles. A first angle look at convergence of the measure implied by the scheme, while a second angle look at convergence of the scheme to the Bayes-value. Convergence to the Bayes-value refers to the convergence to the best possible option value accessible by the strategy class used in the scheme. Convergence in measure provides many insights for parameters tuning, but does not quantify the rate of convergence to the option value. The benefits and drawbacks of the other mode of convergence are reversed. The first mode of convergences shows that SPLS is a consistent estimator for the option value, in that the scheme converges in probability to the option value. The second mode of convergence shows that SPLS is universally Bayes-value consistent, in that, for any stochastic dynamic of the underlying, any payoffs, and any constraints, there exists a value of the hyperparameters that approximate arbitrarily well the best possible option value accessible by the projected strategy class. If we assume that the optimal exercise strategy can be approximated arbitrarily well by a projected strategy, this last result implies that SPLS converges uniformly to the option value.

Completing an energy distance analysis has its roots in the stability theory of stochastic optimal control. Within this theory, a projection scheme is judged good if the scheme preserves the optimum of the control problem. Such a stability property can be guaranteed when the projected stochastic space is close to the true stochastic space in some variant of the Wasserstein distance (Heitsch, Römisch, and Strugarek 2006, Pflug and Pichler 2012). Such a convergence certificate has two important advantages. First, when tuning hyperparameters, the option price should be computed only for hyperparameters that can be certified. Second, such a certificate allows to use low dimensional projection while maintaining accuracy. When the certificate is fast to compute, these advantages accelerate both hyperparameters tuning and the projection scheme computation. Unfortunately, Wasserstein-based distances are not fast to compute, especially in high dimension (Ramdas, Trillos, and Cuturi, 2017). To correct this shortcoming, we use the energy distance instead. We show that SPLS converges in measure when the energy distance is small, and we propose a practical hyperparameters tuning algorithm based on the energy distance.

### 2.2.1 Uniform Convergence in Measure

Convergence in measure can be decomposed into three key drivers. First, the projected strategy space needs to be dense in the space of strategy. This driver is related to the nearest-neighbor basis size  $m$  and the unknown size  $\alpha$  of the smallest nearest-neighbor basis needed to replicate accurately the optimal strategy  $X^*$ . Second, once a dense projected strategy space is found, the projected lookahead operator needs to be accurate. This driver is related to the interaction between the number of sample in the lookahead operator  $N$ , and the nearest-neighbor basis size  $m$ . Third, once an accurate projected lookahead operator is found, the projected option value needs to be accurate. This driver is related to the number of sample  $\tilde{N}$  in the option value  $\bar{V}$ . With these three drivers, we can show that the projected option value  $\bar{V}$  is a consistent estimator for the option value  $V$ .

#### Projected Strategy

Consider the lookahead operator  $L(X_{t:T}, F_t)$  at time  $t$ , and let  $s$  be anytime between  $t$  and  $T$ . For the projected strategy convergence, let  $\alpha_{s-t}$  be the smallest number of Voronoi cell needed to tessellate perfectly the set  $\{(S_{0:s}, X_s^*)\}$ . For example, for a vanilla American option let  $B_i$  be any Voronoi cell of a perfect tessellation, then  $X_s^*(B_i) \in \{\{0\}, \{1\}\}$ , and  $X_s^*(B_i) \in \{\{0, 1\}\}$  is not possible. In other words, a perfect tessellation separates the optimal decision into cell with unambiguous decision. Cells with unambiguous decision are crucial for the projected strategy space to contain the optimal strategy. To quantify this crucial requirement, we look at how big the nearest-neighbor basis size  $m_{s-t}$  needs to be in order to approximate uniformly well any tessellation of size  $\alpha_{s-t}$ .

To write our convergence rate, let  $(\bar{B}_i)_{i=1}^{m_{s-t}}$  be the tessellation generated by the nearest-neighbor basis at time  $s$ . Also, for a given tessellation of size  $\alpha_{s-t}$ , let

$$A \in 2^{\prod_{i=1}^{\alpha_{s-t}} B_i} \setminus \{\{B_1\}, \{B_2\}, \dots, \{B_{\alpha_{s-t}}\}, \{\cup_{i=1}^{\alpha_{s-t}} B_i\}\},$$

be the power set of all permutation of the cells in the tessellation, excluding the permutation that contains a single cell of the tessellation, and excluding the union of the cells. In Methods, we show

that the uniform convergence rate of the projected strategy can then be written with

$$P\left(\sup_A \frac{1}{m_{s-t}} \sum_{i=1}^{m_{s-t}} \mathbb{I}[\bar{B}_i \in A] > \epsilon\right) \leq 2m_{s-t}^{3d(s-t)\alpha_{s-t}^2} e^{-2\epsilon^2 m_{s-t}}, \quad (2.15)$$

where  $\epsilon > 0$ . This convergence rate says that the empirical probability that a projected Voronoi cell  $\bar{B}_i$  overlaps several cells of any tessellation of size  $\alpha_{s-t}$  decreases exponentially fast with  $m_{s-t}$ . The key ingredients for the proof are the shatter coefficient of the family of set  $A$  (Vapnik and Chervonenkis 1971), and Hoeffding inequality. In regards to hyperparameters tuning, this convergence rate says that the projected strategy space is dense as soon as  $m_{s-t}$  is higher than  $3d(s-t)\alpha_{s-t}^2$ . In particular, a good projected strategy space is accessible well before  $m_{s-t}$  is taken as infinity.

### Projected Lookahead Operator

Assuming that the projected strategy space is dense, the projected lookahead operator will be close to the true lookahead operator if the distribution of the projected residual payoff  $J(\bar{X}_{t:T}, S^n)$  is uniformly accurate. This distribution is defined by the joint distribution of the projected strategy and the projected stock path. Since the projected strategy is a function of several nearest-neighbor basis, and since the stock path takes value in a real plane, this joint distribution is characterized by set of the form  $A \in \left(\prod_{s=t}^T \{\bar{B}^{s,i}\}_{i=1}^{m_{s-t}}, A'\right)$  where  $\bar{B}^{s,i}$  is the Voronoi cell generated at time  $s$  by the centroid  $\bar{S}_{t:s}^{s,i}$ , and  $A'$  is an interval in  $\mathbb{R}^{d(T-t)}$ . To write the convergence rate, let  $\nu(A) = P(S_{t:T} \in A)$  be the probability that a sample path  $S_{t:T}$  is in both components of  $A$ , and let  $\nu_N(A) = \frac{1}{N} \sum_{n=1}^N \mathbb{I}[S_{t:T}^n \in A]$  be the empirical probability. A direct application of the Vapnik-Chervonenkis inequality (Vapnik and Chervonenkis 1971, Devroye, Györfi, and Lugosi 1996) gives the following rate of convergence

$$P\left(\sup_A |\nu_N(A) - \nu(A)| > \epsilon\right) \leq 8N^{3d^2(T-t)^2 \sum_{s=t}^T m_{s-t}^2} e^{-\epsilon^2 N/32}, \quad (2.16)$$

with  $\epsilon > 0$ . See Methods for more details. This rate of convergence says that the square of the nearest-neighbor basis size affects the residual payoff distribution. For hyperparameters tuning, this implies that the basis sizes should be chosen as small as possible, so that an accurate distribution is

accessible with a moderate number of sample  $N$ . Indeed, recall that to compute the projected option value several samples of  $J(\bar{L}_{0:T}, S_{0:T})$  are needed: the larger  $N$ , the heavier the computation of each projected lookahead operator  $\bar{L}_t$ . A good way to satisfy this requirement, is to use a monotonically increasing basis size, for example  $m = (1, \alpha_1, \alpha_2, \dots, \alpha_T)$ , rather than a constant basis size such as  $m = (1, \alpha_T, \alpha_T, \dots, \alpha_T)$ . Of course, since the optimal nearest-neighbor basis size  $\alpha$  is not known, hyperparameters tuning should be performed by enforcing a monotone constraint on the basis size.

## Projected Option Value

Since the projected option value is simply a Monte Carlo expectation, the uniform convergence of the projected option value is given by the uniform law of large numbers of Vapnik and Chervonenkis (1982). Since the goal of this section is to understand the hyperparameters impact, we make the simplifying assumption that the projected lookahead operator is a uniform Lipschitz function of the filtration. This assumption will be removed in the next section. Assume further that the residual payoff  $J(X, S)$  is a Lipschitz function of the decision  $X$ . The convergence rate can then be written with

$$P\left(\sup_{\bar{L}} |\bar{V} - E(J(\bar{L}, S))| > \epsilon\right) \leq 8 \left(\frac{CK}{\epsilon}\right)^{(a+d)T^2} e^{-\epsilon^2 \tilde{N}/(128B^2)}, \quad (2.17)$$

where  $C$  is the Lipschitz constant of the projected lookahead operator,  $B$  is an upper bound on the residual payoff,  $K$  is the Lipschitz constant of the residual payoff, and  $a$  is the dimension of a decision, so that  $X \in \mathbb{R}^{a(T+1)}$ . See Methods for more details. For a particular option and a particular stochastic dynamic of the stock, we can assume that the Lipschitz constant  $C$  is universal. The rate of convergence is then uniform with respect to the projected lookahead operator  $\bar{L}$ , and the hyperparameter  $\tilde{N}$  is independent, in term of convergence impact, of the number of sample in the lookahead operator  $N$  and the nearest-neighbor basis size  $m$ . For hyperparameters tuning, this implies that once a value of  $\tilde{N}$  is found such that the projected option value converges, the same value of  $\tilde{N}$  can be used to tune the two other hyperparameters  $N$  and  $m$ .

With the three previous convergence rates, we can prove that the projected option value  $\bar{V}$  converges in probability to the option value  $V$  when the hyperparameters are taken as very large. This result

can be written with

$$\lim P(|\bar{V} - V| > \epsilon) = 0, \quad (2.18)$$

where the limit is taken with  $\tilde{N}$ ,  $N$  and  $m$  going to infinity, with  $m_{\max}^2 < N$ , and  $m_{\max} = \|m\|_{\infty}$  the largest nearest-neighbor size. See Methods for more details. In the proof, we make the assumptions that the convergence rates (2.15) and (2.16) control the denseness of the projected strategy space and the accuracy of the projected lookahead operator. The next section presents a stronger result that implies consistency without these assumptions.

### 2.2.2 Uniform Convergence in Bayes-Value

Convergence in Bayes-value is obtained by assuming that the residual payoff is Lipschitz in the strategy. The first step towards this result is to obtain the rate of convergence of the average residual payoff to the expected residual payoff. This rate of convergence can be written with

$$\begin{aligned} & P(\sup_{\bar{L}} \left| \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} J(\bar{L}_{0:T}, S^{\tilde{n}}) - E(J(\bar{L}_{0:T}, S)) \right| > \epsilon) \\ & \leq 8 \exp \left( T(cdTm_{\max}^3)^T \left( \frac{KX_{\max}S_{\max}}{\epsilon} \right)^{c(a+d)T^2m_{\max}^3} e^{-\epsilon^2 N/(512B^2K^2)} - \epsilon^2 \tilde{N}/(128B^2) \right), \quad (2.19) \end{aligned}$$

where the supremum is taken over the class of strategy implied by the projected lookahead operator,  $B$  is an upper bound for the residual payoff,  $K$  is the Lipschitz constant of the residual payoff,  $c$  is a universal constant,  $X_{\max}$  is an upper bound for the decision, and  $S_{\max}$  is an upper bound for the stock. The key ingredients of the proof are the uniform law of large numbers, and a careful estimates of the covering number for the residual payoff. The Lipschitz assumption allows to express this covering number in terms of a covering number for the projected lookahead operator. The resulting covering number can then be estimated with another application of the uniform law of large numbers. Further, this rate of convergence can be used to prove consistency of SPLS. See Methods for more details.

With the previous rates of convergence, we can show the universal convergence of the projected



option value to the Bayes-value. This result can be written with

$$\lim \bar{V} = V, \tag{2.20}$$

where the lookahead operator in the option value  $V$  is restricted to the class of strategy implied by the projected lookahead operator, the limit is taken with  $\tilde{N}$ ,  $N$  and  $m$  going to infinity, and  $m_{\max}^3 < o(N)$ . By assuming that the class of projected strategy is dense in the space of strategy, this result implies that SPLS converges uniformly to the option value.

### 2.2.3 Energy Convergence

We start by reviewing the definition of energy distance and by applying this distance to a Voronoi tessellation. Then, we present an energy-based hyperparameters tuning algorithm for SPLS.

Let  $X$  and  $Y$  be two random variables. The energy distance  $ED(X, Y)$  between  $X$  and  $Y$  is defined as

$$ED(X, Y) = 2E(\|X - Y\|) - E(\|X - X'\|) - E(\|Y - Y'\|),$$

where  $X'$  and  $Y'$  have the same distribution than  $X$  and  $Y$ , all variables are independent, and the norm can be defined freely. We use the  $\ell_1$ -norm. Given a sample  $(X_i)_{i=1}^n$  and  $(Y_j)_{j=1}^m$ , the energy distance can be estimated with

$$\widehat{ED}(\{X_i\}, \{Y_j\}) = \frac{2}{nm} \sum_{i=1}^n \sum_{j=1}^m \|X_i - Y_j\| - \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \|X_i - X_j\| - \frac{1}{m^2} \sum_{i=1}^m \sum_{j=1}^m \|Y_i - Y_j\|.$$

The energy distance has the property that  $X$  and  $Y$  have the same distribution if and only if  $ED(X, Y) = 0$ .

For the projected lookahead operator, the convergence rate (2.16) shows that the distribution that is crucial to guarantee the accuracy of the projected lookahead operator is the joint distribution of the strategy and the stock path. Since this joint distribution is the distribution of a stochastic process,

the distribution can be characterized by its conditional distribution. As the projected lookahead operator is a Monte Carlo average on the stock path, any discrepancy in the conditional distribution originates from the projected strategy. In particular, if the number of sample path  $N$  used in the projected lookahead operator is not adequate with respect to the nearest-neighbor basis size  $m$ , the conditional distribution will concentrate in the wrong cell of the Voronoi tessellation induced by the projected strategy. One way to measure the accuracy of the projected lookahead operator is hence to verify that the conditional distribution concentrates in the appropriate Voronoi cells.

To measure the accuracy of the conditional distribution concentration, consider the projected lookahead operator  $L_t$  at time  $t$ , a sample path  $S$ , and any time  $s$  between time  $t$  and  $T - 1$ . Denote the conditional cell by  $\pi S_s = (i_s, i_{s+1})$ . The conditional cell is the index of the Voronoi cell at time  $s$  and  $s + 1$  for the stock path  $S$ . The index  $i_s$  denotes one of the Voronoi cell  $\{\bar{B}^{s,i}\}_{i=1}^{m_{s-t}}$ , and similarly for  $i_{s+1}$ . For an entire sample  $S_{t:T}^{(N)} = (S_{t:T}^n)_{n=1}^N$ , denote by  $\pi S_{t:T}^{(N)}$ , the resulting sample of conditional cells.

For a fix nearest-neighbor size  $m$ , a large sample  $N'$  of conditional cells can be considered as the reference conditional distribution. The adequacy of the number of sample paths  $N$  in the projected lookahead operator can then be measured by the energy distance between the reference sample and the conditional cells sample induced by the projected lookahead operator. This energy distance can be written with

$$\widehat{ED}(\bar{L}_t) = \widehat{ED}(\pi S_{t:T}^{(N')}, \pi S_{t:T}^{(N)}).$$

By extension, the total energy of the projected lookahead operator  $\bar{L}$  can be defined as the expected average energy. This total energy can be written with

$$\widehat{ED}(\bar{L}) = \frac{1}{\tilde{N}T} \sum_{n=1}^{\tilde{N}} \sum_{t=0}^T \widehat{ED}(\bar{L}_t). \quad (2.21)$$

For process with independent and identically distributed increments, the total energy can be esti-

mated with the energy of the lookahead at time zero, giving

$$\widehat{ED}(\bar{L}) = \frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} \widehat{ED}(\bar{L}_0).$$

Whenever applicable, the second form for the total energy is preferable as this form is faster to compute.

By construction, a total energy of zero is a certificate that the projected lookahead operator is accurate. A total energy of zero is hence an equivalent condition to the convergence rate (2.16) and can be used as a substitute to prove the consistency of SPLS (2.18).

Using the total energy, hyperparameters tuning can be done as follows. First, fix a sample size  $\tilde{N}$  and an energy threshold  $\delta$ . The sample size  $\tilde{N}$  can be small as the energy certificate guarantees the convergence. The energy threshold  $\delta$  is the maximum total energy that we are willing to tolerate in order to consider the projected lookahead operator accurate. In numerical experiments, we found that there is a clear shift in the total energy once the lookahead is no more accurate, and it is not necessary to develop a formal hypothesis test to define this threshold. Such a threshold can be found by computing the projected option value (2.14) on a small set of hyperparameters. The threshold is then given by the hyperparameters with the maximum projected option value. Second, the projected option value is computed conditionally on the energy threshold on a grid of the form

$$(N = 1, 2, \dots, N_{\max}) \times (m = 1, 2, \dots, m_{\max}),$$

The notation  $m = x$  means that the nearest-neighbor size is increasing with  $m_0 = 1$  and  $m_T = x$ . At each grid point, the total energy (2.21) is computed and the projected option value is computed only if the total energy is below the threshold  $\delta$ . Third, the maximum projected option value found through the grid search is taken as a lower estimate for the option price.

### 3 Nearest Neighbor Martingale

#### 3.1 Algorithm

To present NNM, assume that any martingale is defined as a random walk with increments that are conditionally zero-mean. Such a martingale  $M$  can be written with

$$\begin{aligned} M_0 &= 0, \\ M_t &= M_{t-1} + I_t, \quad t = 1, 2, \dots, T, \\ E(I_t | F_{t-1}) &= 0, \quad t = 1, 2, \dots, T, \end{aligned}$$

where  $I_t$  is the random walk increment at time  $t$ . The martingality condition  $E(M_t | F_{t-1}) = M_{t-1}$  is enforced by the increment being conditionally zero-mean. Denote by  $D(M, S)$  the dual payoff for a particular martingale  $M$  and a sample path  $S$ . This dual payoff can be written with

$$D(M, S) = \max_{x, y, g(x, y, S)=0} \sum_{t=0}^T e^{-rt} f_t(y_{0:t}, S_{0:t}) x_t - M_t x_t.$$

Given the probability measure  $P$  of the sample path, the Rogers operator gives the martingale that minimizes the expectation of the dual payoff. This operator can be written with

$$R(P) = \operatorname{argmin}_M E \left[ D(M, S) \right].$$

The option value is then given by the expectation of the dual payoff with the martingale given by the Rogers operator

$$V = E \left[ D(R(P), S) \right].$$

See Methods for the equivalence of the dual and primal problem.

To estimate  $V$ , we use several Monte Carlo projections and a relaxation of the Rogers operator. First, we project the martingale increment  $I_t$  onto the space of conditional nearest-neighbor basis

for the stock paths. The centroid for such a basis is a pair  $(\bar{S}_{0:t-1}^{t,i}, \bar{S}_t^{t,i,j})$  where the path  $\bar{S}_{0:t-1}^{t,i}$  is sampled randomly, and the next stock price  $\bar{S}_t^{t,i,j}$  is sampled conditional on the path  $\bar{S}_{0:t-1}^{t,i}$ . The resulting step-function can be written with

$$\bar{I}_t(S_{0:t}) = \sum_{i=1}^{p_t} \sum_{j=1}^{q_t} x^{t,i,j} \llbracket S_{0:t-1} \sim \bar{S}_{0:t-1}^{t,i}, S_t \sim \bar{S}_t^{t,i,j} \rrbracket,$$

where  $p_t$  is the number of stock path used,  $q_t$  is the number of conditional stock price used, and  $x^{t,i,j} \in \mathbb{R}$  are the basis weights. We write simply  $\bar{I}_t$  when no confusion is possible. Second, we relax the Rogers operator to a Frank-Wolfe iteration. To describe the iteration, let  $\bar{Z}$  be any projected martingale. The relaxed Rogers operator produces an improved projected martingale  $\bar{M}$  by solving the following linear program

$$\bar{R}(\bar{Z}) = \operatorname{argmin}_{x^{t,i,j}} \frac{1}{N} \sum_{n=1}^N D(\bar{M}, S^n) \quad (3.1)$$

$$\text{s.t. } \sum_{n=1}^N \sum_{j=1}^{q_t} x^{t,i,j} \llbracket S_{0:t-1}^n \sim \bar{S}_{0:t-1}^{t,i}, S_t^n \sim \bar{S}_t^{t,i,j} \rrbracket = 0 \quad t = 1, 2, \dots, T, \quad i = 1, 2, \dots, p_t \quad (3.2)$$

$$\bar{M}_0 = 0 \quad (3.3)$$

$$x^{t,i,j} - z^{t,i,j} \in [-\eta, \eta] \quad t = 1, 2, \dots, T, \quad i = 1, 2, \dots, p_t, \quad j = 1, 2, \dots, q_t \quad (3.4)$$

where  $\eta > 0$  is the learning rate,  $x^{t,i,j}$  are the basis weights for the increment of the martingale  $\bar{M}$ , and  $z^{t,i,j}$  are the basis weights for the increment of the martingale  $\bar{Z}$ . The constraints (3.2) and (3.3) ensure that the improved martingale satisfies the martingality condition, while the constraint (3.4) ensures that the improved martingale is a small change to the given martingale. The relaxed operator is iterated  $m$  times to obtain the projected optimal martingale  $\bar{R}^m(\bar{Z})$ . Third, we project the dual option value with another sample  $\{\tilde{S}^{\tilde{n}}\}_{\tilde{n}=1}^{\tilde{N}}$ . This projected dual value can be written with

$$\bar{V} = \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} D(\bar{R}^m(\bar{Z}), \tilde{S}^{\tilde{n}}). \quad (3.5)$$

As the next section will show, the computational effort required in the relaxed Rogers operator to obtain a good projected dual value is often prohibitive. For this reason, the projected dual value

is further relaxed by using the same sample  $\{S^n\}_{n=1}^N$  that is used for the relaxed operator. This relaxed dual value can be written with

$$\bar{V} = \frac{1}{N} \sum_{n=1}^N D(\bar{R}^m(\bar{Z}), S^n). \quad (3.6)$$

As the next section will show, estimating the dual option value with the relaxed dual value is not an heuristic. The relaxed dual value converges faster to the dual option value than the projected dual value. However, the optimal martingale associated to the relaxed dual value does not generalize well to out-of-sample test. In other words, the relaxed dual value approximates the distribution of the sample path well enough to obtain a good point estimate, but not well enough to obtain a good estimate of a stochastic process.

## 3.2 Convergence

For hyperparameters tuning insights, we first look at the uniform convergence in measure of NNM, which shows that the relaxed dual value is a consistent estimator of the dual value. Second, we look at the uniform convergence in Bayes-value, which shows that NNM is universally Bayes-value consistent. Third, we complete an energy convergence analysis and provide an energy-based algorithm for hyperparameters tuning.

### 3.2.1 Uniform Convergence in Measure

The convergence in measure of NNM is driven by three major factors. First, the projected martingale space needs to be dense, second, the relaxed Rogers operator needs to be accurate, and third, the projected dual value needs to be accurate. The relaxed operator accuracy depends on whether the operator is used for the projected dual value (a stochastic process estimate) or the relaxed dual value (a point estimate).

### Projected Martingale

Denote by  $(p, q) = ((p_t, q_t), t = 1, 2, \dots, T)$  the nearest-neighbor basis size. Without loss of

generality, we can assume that the optimal martingale is a nearest-neighbor martingale of size  $(\alpha, \beta)$ . For example, for a vanilla American put in a binomial world (Cox, Ross, and Rubinstein 1979)  $(\alpha, \beta) = ((2^{t-1}, 2), t = 1, 2, \dots, T)$ , and a good projected martingale can be obtained with  $(p, q) = ((\mathcal{O}(t), 2), t = 1, 2, \dots, T)$ , see Methods for more details. The projected martingale space can be considered dense in the space of martingale as soon as the projected space approximates uniformly well any martingale of size  $(\alpha, \beta)$ .

We consider separately the convergence of the conditioning part and the current part of the nearest-neighbor basis. For the conditioning part, let  $(\bar{B}_i)_{i=1}^{p_t}$  be the tessellation generated by the conditioning part of the nearest-neighbor basis at time  $t$ . For any conditioning tessellation of size  $\alpha_t$  let

$$A \in 2^{\prod_{i=1}^{\alpha_t} B_i} \setminus \{\{B_1\}, \{B_2\}, \dots, \{B_{\alpha_t}\}, \{\cup_{i=1}^{\alpha_t} B_i\}\},$$

be the power set of all permutation of the cell  $B_i$  in the tessellation, excluding the permutation that contains a single cell of the tessellation, and excluding the union of the cells. In Methods, we show that the uniform convergence rate of the conditioning part can be written with

$$P\left(\sup_A \frac{1}{p_t} \sum_{i=1}^{p_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon\right) \leq 2p_t^{3d\alpha_t^2} e^{-2\epsilon^2 p_t}, \quad (3.7)$$

where  $\epsilon > 0$ .

For the current part, let  $(\bar{B}_i)_{i=1}^{q_t}$  be the tessellation generated by the current part of the nearest-neighbor basis at time  $t$ . For any current part tessellation of size  $\beta_t$  let

$$A \in 2^{\prod_{i=1}^{\beta_t} B_i} \setminus \{\{B_1\}, \{B_2\}, \dots, \{B_{\beta_t}\}, \{\cup_{i=1}^{\beta_t} B_i\}\},$$

be the power set of all permutation of the cell  $B_i$  in the tessellation, excluding the permutation that contains a single cell of the tessellation, and excluding the union of the cells. In Methods, we show that the uniform convergence rate of the current part can be written with

$$P\left(\sup_A \frac{1}{q_t} \sum_{i=1}^{q_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon\right) \leq 2q_t^{3d\beta_t^2} e^{-2\epsilon^2 q_t}, \quad (3.8)$$

where  $\epsilon > 0$ .

These convergence rates say that the empirical probability that a conditional Voronoi cell  $\bar{B}_i$  overlaps several cells of any conditional tessellation of size  $(\alpha_t, \beta_t)$  decreases exponentially fast with  $p_t$  and  $q_t$ . In regard to hyperparameters tuning, this convergence rate says that the projected martingale space is dense as soon as  $p_t$  is higher than  $3dt\alpha_t^2$ , and as soon as  $\beta_t$  is higher than  $3d\beta_t^2$ .

## Relaxed Rogers Operator

Assuming that the projected martingale space is dense, the convergence of NNM lies in the accuracy of the relaxed Rogers operator. When the relaxed Rogers operator is used to estimate the projected dual value, the goal is to find a projected martingale that generalizes well to any out-of-sample computation of the dual value. Within a probabilistic setting, this goal is equivalent to estimate accurately the distribution of pair of the form  $(\bar{M}, M)$ , where  $\bar{M}$  is a projected martingale, and  $M$  is the optimal martingale. When the relaxed Rogers operator is used to estimate the relaxed dual value, the goal is to estimate the dual option value, and this goal is equivalent of estimating the distribution of pair of the form  $(\bar{M}, D)$  where  $\bar{M}$  is a projected martingale, and  $D$  is the dual payoff value. The first usage is a stochastic process estimate, while the second usage is a point estimate.

To write the convergence rates, let  $B = \prod_{t=0}^T \{\bar{B}_i\}_{i=0}^{p_t q_t}$  be the tessellation generated by the projected martingale. The optimal martingale  $M$  is a vector in  $\mathbb{R}^T$ , and the dual payoff  $D$  is a real scalar.

When the relaxed Rogers operator is used for a stochastic process estimate, denote by  $A$  set of the form  $B \times [a, b]$ , with  $[a, b]$  an interval in  $\mathbb{R}^T$ , and let  $\nu$  be the probability that a sample stock path and the optimal martingale are in  $A$ ,  $\nu(A) = P((S, M) \in A)$ . The corresponding empirical probability can be written with  $\nu_N(A) = \frac{1}{N} \sum_{i=1}^N \mathbb{I}[S \in B, M \in [a, b]]$ . A direct application of the Vapnik-Chernovenkis inequality (Vapnik and Chervonenkis 1971, Devroye, Györfi, and Lugosi 1996) gives the following rate of convergence

$$P\left(\sup_A |\nu_N(A) - \nu(A)| > \epsilon\right) \leq 8N^{3dT} \sum_{t=1}^T t p_t^2 q_t^2 e^{-\epsilon^2 N/32}, \quad (3.9)$$

with  $\epsilon > 0$ . See Methods for more details.



When the relaxed Rogers operator is used for a point estimate, denote by  $A$  set of the form  $B \times [a, b]$ , with  $[a, b]$  an interval in  $\mathbb{R}$ , and let  $\nu$  be the probability that a sample stock path and the dual payoff are in  $A$ ,  $\nu(A) = P((S, D) \in A)$ . The corresponding empirical probability can be written with  $\nu_n(A) = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[S \in B, D \in [a, b]]$ . A direct application of the Vapnik-Chervonenkis inequality (Vapnik and Chervonenkis 1971, Devroye, Györfi, and Lugosi 1996) gives the following rate of convergence

$$P\left(\sup_A |\nu_n(A) - \nu(A)| > \epsilon\right) \leq 8N^{3d} \sum_{t=1}^T t p_t^2 q_t^2 e^{-\epsilon^2 N/32}, \quad (3.10)$$

with  $\epsilon > 0$ . See Methods for more details.

The first observation that comes from these convergence rates is that to guarantee the accuracy of a stochastic process estimate the sample size  $N$  needs to be  $T$  times bigger than for a point estimate. As each iteration of the relaxed Rogers operator demands to compute the dual payoff for each sample stock path, and demands to solve a large linear program, this factor often makes a point estimate much faster to obtain. Second, these convergence rates show that the product of the nearest-neighbor basis size slowdown the convergence of the relaxed operator. The basis size should hence be chosen as small as possible. In particular, the conditioning size  $p_t$  and the current size  $q_t$  can be taken as monotonically increasing.

## Projected Dual Value

Once the relaxed Rogers operator is accurate, NNM accuracy lies in an accurate estimation of the projected dual value. Since the goal of this section is to understand the hyperparameters impact, we use a probabilistic perspective where the relaxed Rogers operator is viewed as a martingale, and we look at how well the dual value associated to any martingale can be estimated. With this assumption, analyzing the convergence of the projected dual value (3.5) or the relaxed dual value (3.6) is equivalent. We use the notation of the projected dual value.

Let  $\bar{M}$  be a projected martingale, the convergence rate of the projected dual value to the dual value follows by assuming that the dual payoff is a Lipschitz function of the martingale, and by using the

uniform law of large numbers. This rate can be written with

$$\begin{aligned}
& P(\sup_{\bar{M}} \left| \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} D(\bar{M}, S^{\tilde{n}}) - E(D(\bar{M}, S)) \right| > \epsilon) \\
& \leq 8(cdT p_{\max}^3 q_{\max}^3)^T \left( \frac{UK}{\epsilon} \right)^{cdT^2 p_{\max}^3 q_{\max}^3} e^{-\epsilon \tilde{N}/(128B^2)}, \tag{3.11}
\end{aligned}$$

where  $(p_{\max}, q_{\max}) = (\|p\|_{\infty}, \|q\|_{\infty})$  is the largest nearest-neighbor size,  $U$  is an upper bound on the projected martingale,  $K$  is the Lipschitz constant of the dual payoff,  $B$  is an upper bound on the dual payoff, and  $c$  is a universal constant. See Methods for more details. For hyperparameters tuning, this convergence rate says that the nearest-neighbor basis size  $(p, q)$  has a direct impact on the convergence of the projected dual value. An appropriate sample size  $\tilde{N}$  should hence be chosen by considering the biggest nearest-neighbor basis that will be used in hyperparameters tuning. Once such a sample size  $\tilde{N}$  is found, the projected dual value will be accurate for all the nearest-neighbor basis size considered.

With the previous convergence rates, we can prove that the projected dual value  $\bar{V}$  and the relaxed dual value  $\bar{\bar{V}}$  converge in probability to the option value  $V$  when the hyperparameters are taken as very large. These results can be written with

$$\lim P(|\bar{V} - V| > \epsilon) = 0, \tag{3.12}$$

$$\lim P(|\bar{\bar{V}} - V| > \epsilon) = 0, \tag{3.13}$$

where the limit is taken with  $\tilde{N}$ ,  $N$  and  $(p, q)$  going to infinity. The increasing rate for the nearest-neighbor basis size is  $p_{\max}^3 q_{\max}^3 < o(N)$  for the projected dual value, and  $p_{\max}^2 q_{\max}^2 < o(N)$  for the relaxed dual value. This difference in increasing rate further shows that the computational effort behind the projected dual value and the relaxed dual value are substantially different. See Methods for more details.

### 3.2.2 Uniform Convergence in Bayes-Value

For the projected dual value, convergence in Bayes-value is obtained by assuming that the dual payoff is Lipschitz in the martingale, and by considering a more general relaxed Rogers operator. To define this operator, denote by  $S^{(N)} = (S^n)_{n=1}^N$  a random sample of size  $N$  of the stock path. The general relaxed Rogers operator  $\bar{R}(S^{(N)})$  makes no assumption on the optimization method used and can be written with

$$\bar{R}(S^{(N)}) = \operatorname{argmin}_{\bar{M}} \frac{1}{N} \sum_{n=1}^N D(\bar{M}, S^n),$$

where the minimization is subject to the same constraint than in (3.1). The rate of convergence to the Bayes-value can then be written with

$$\begin{aligned} & P\left(\sup_{\bar{R}(S^{(N)})} \left| \frac{1}{N} \sum_{n=1}^{\tilde{N}} D(\bar{R}(S^{(N)}), S^n) - E(D(\bar{R}(S^{(N)}), S)) \right| > \epsilon\right) \\ & \leq 8(cdT p_{\max}^3 q_{\max}^3 N^3)^T \left(\frac{UK}{\epsilon}\right)^{cdT^2 p_{\max}^3 q_{\max}^3 N^3} e^{-\epsilon^2 \tilde{N}/(128B^2)}, \end{aligned} \quad (3.14)$$

where the supremum is taken over the class of martingale implied by the general relaxed Rogers operator,  $U$  is an upper bound on the projected martingale,  $K$  is the Lipschitz constant of the dual payoff,  $B$  is an upper bound on the dual payoff, and  $c$  is a universal constant.

For the relaxed dual value, there is no out-of-sample test and the convergence in Bayes-value can be analyzed directly with a probabilistic view of the relaxed Rogers operator. The convergence rate (3.11) can hence be taken as the Bayes-value rate of the relaxed dual value.

With the previous rates of convergence, we can show the universal convergence of the projected dual value and the relaxed dual value to the Bayes-value. These results can be written with

$$\lim \bar{V} = V, \quad (3.15)$$

$$\lim \bar{\bar{V}} = V, \quad (3.16)$$

with the Rogers operator restricted to the class of martingale implied by the relaxed Rogers operator. For the projected dual value  $\bar{V}$ , the limit is taken with  $\tilde{N}$ ,  $N$ , and  $(p, q)$  going to infinity, with an increasing rate of  $p_{\max}^3 q_{\max}^3 N^3 < o(\tilde{N})$ . For the relaxed dual value  $\bar{\bar{V}}$ , the limit is taken with  $N$  and  $(p, q)$  going to infinity, with an increasing rate of  $p_{\max}^3 q_{\max}^3 < o(N)$ . The difference in increasing rate between the projected dual value and the relaxed dual value supports the difference in computational effort between the two methods. By assuming that the class of projected martingale is dense in the space of strategy, these Bayes-value consistency results imply that NNM converges uniformly to the dual option value.

### 3.2.3 Energy Convergence

As the projected dual value needs a stochastic process estimate, an energy score for this method is difficult to design. Indeed, the measure that needs to be estimated is given in (3.9) and is for set of the form  $(\bar{M}, M)$ , where  $\bar{M}$  is a projected martingale, and  $M$  is the optimal martingale. The projected martingale is a step-function on a nearest-neighbor basis of the stock, and the quality of its distribution can be measured by looking at the energy of the tessellation. For the optimal martingale however there is no simple basis of comparison. Especially so, that the optimal martingale is unknown. This shortcoming does not apply to the relaxed dual value as from (3.10) the distribution that needs to be estimated is for set of the form  $(\bar{M})$ .

To define an energy score for the relaxed dual value, we adapt the energy score of SPLS (Section 2.2.3). To this end, fix a time  $t$ , and consider the conditional tessellation  $\cup_{i=1}^{p_t} \cup_{j=1}^{q_t} B_{i,j}$  induced by the projected martingale. A Voronoi cell in this tessellation is of the form  $B_{i,j} = U_i \times W_j$  where  $U_i$  is the conditioning part, and  $W_j$  is the current part. As the tessellation already has a conditional nature, the conditional cell at time  $t$  for a stock path  $S$  that falls in the cell  $B_{i,j}$  can be defined with  $\pi(S_{0:t-1}, S_t) = (i, j)$ , where  $i$  is the index of the conditioning part cell, and  $j$  is the index of the current part cell. For a sample of stock paths  $S^{(N)} = (S^n)_{n=1}^N$ , denote by  $\pi S^{(N)}$  the corresponding sample of conditional cells. The energy of the relaxed dual value can then be written with

$$\widehat{ED}(\bar{R}(\bar{Z})) = \widehat{ED}(\pi S^{(N)}, \pi S'^{(N')}),$$

where  $S^{(N')}$  is another independent sample with  $N'$  much bigger than  $N$ . As the energy distance detects any discrepancy in distribution, an energy of zero is an equivalent condition to (3.10) and can be used as substitute to prove the consistency of the relaxed dual value (3.13).

In numerical experiments, we found that when the nearest-neighbor size  $(p, q)$  is too high compared to the sample size  $N$ , the relaxed dual value does not converge. The Frank-Wolfe iteration in the relaxed Rogers operator is able to continually decrease the relaxed dual value. In contrast, when the nearest-neighbor size is appropriate for the sample size, the Frank-Wolfe iteration converges. Defining an appropriate energy threshold can hence be done by detecting where the Frank-Wolfe iteration diverges. A simple way to detect this divergence is to compare the Frank-Wolfe iteration to the lower bound found by SPLS. Alternatively, instead of defining an energy threshold, the SPLS lower bound can be used to define a barrier past which the Frank-Wolfe iteration is considered divergent. For example, with an SPLS value of  $v$ , the Frank-Wolfe iteration is considered divergent whenever the relaxed dual value falls below  $v$ . As SPLS is a lower estimate of the option price, the lower bound barrier may be setted slight higher than the actual SPLS value. For example, the lower bound barrier can be setted to  $\alpha v$ , with  $\alpha > 1$ . We call the factor  $\alpha$  the lower bound repulsion factor.

Using the filtration energy, hyperparameters tuning can be done as follows. First, fix a sample size  $N$  and an energy threshold  $\delta$ . The sample size  $N$  can be small as the energy certificate guarantees the convergence. The energy threshold  $\delta$  is the maximum energy that we are willing to tolerate to consider the relaxed dual value accurate. This threshold can be selected by detecting where the relaxed dual value (3.6) diverges. Second, the filtration energy is computed on a grid of the form

$$(p = 1, 2, \dots, p_{\max}) \times (q = 1, 2, \dots, q_{\max}),$$

The notation  $p = x$  means that the conditioning size is increasing with  $p_0 = 1$  and  $p_T = x$ , and similarly for the notation  $q = x$ . Third, the minimum relaxed dual value found through the grid search is taken as an upper estimate for the option price.

## 4 An American Option

This section and the following applies SPLS and NNM to options of increasing complexity. For each option, we present the price of the option under different pricing parameters. The first two examples can be valued by exact pricing algorithm, and we compare our algorithms to option prices that are available in the literature. See Methods for a description of our implementation of SPLS and NNM.

Consider a single exercise American put option on a stock  $S$  with a strike price of  $K$ . The option can be exercised 50 times per year, up to the maturity  $T$ . The stochastic program for this option can be written with

$$\max_X E \left[ \sum_{t=0}^T e^{-rt} (K - S_t)_+ X_t \right] \quad (4.1)$$

$$\text{s.t.} \quad \sum_{t=0}^T X_t \leq 1 \quad (4.1.1)$$

where  $X$  is the exercise decision, and  $r$  is the risk-free rate. The risk-neutral dynamic for the stock price  $S$  is a geometric Brownian motion and can be described by the following stochastic differential equation

$$dS = rSdt + \sigma SdW, \quad (4.2)$$

where  $\sigma$  is the stock volatility, and  $W$  is a standard Brownian motion. Our benchmark for this example are the finite difference prices computed in Longstaff and Schwartz (2001). These prices are computed by an implicit finite difference scheme, with 40,000 time steps per year and 1,000 steps for the stock price.

Table 4.1 presents the option price under different pricing parameters. Both SPLS and NNM uses a 1,000 simulations. A first observation from this table is that the price estimates are in average within 5% of the option price. This range of precision is adequate for an estimation algorithm, especially if this precision is maintained with much more complex options. A second observation

from this table is that there is some volatility in the estimates, and that there is no simple rule on the pricing parameters that allow to predict this volatility. This situation is made on purpose. Table 4.1 uses a very coarse hyperparameters grid, and uses a low number of simulation to keep the estimates fast. The table hence shows the performance that comes out-of-box with the algorithms. Later, we will see that with a finer hyperparameters tuning and more computing time, the volatility in the estimates disappears. A third observation from this table is that the energy level of the estimates is low. This low energy level indicates that the projection scheme behind the algorithms agrees in distribution with the distribution of the pricing problem. Even if a low number of simulations was used, this agreement in distribution gives confidence that the estimates are the best possible within the chosen projected space dimension.

$S_0$	$\sigma$	$T$	Finite Difference	SPLS		NNM	
36	0.20	1	4.478	4.251	[0.08] (-0.05)	4.337	[0.07] (-0.03)
		2	4.840	4.591	[0.00] (-0.05)	4.877	[0.05] (0.01)
	0.40	1	7.101	6.901	[0.00] (-0.03)	7.090	[0.11] (-0.00)
		2	8.508	8.462	[0.00] (-0.01)	8.814	[0.06] (0.04)
38	0.20	1	3.250	3.232	[0.00] (-0.01)	3.329	[0.05] (0.02)
		2	3.745	3.608	[0.00] (-0.04)	3.861	[0.05] (0.03)
	0.40	1	6.148	5.989	[0.09] (-0.03)	6.164	[0.15] (0.00)
		2	7.670	7.573	[0.00] (-0.01)	7.729	[0.05] (0.01)
40	0.20	1	2.314	2.325	[0.00] (0.00)	2.393	[0.06] (0.03)
		2	2.885	2.753	[0.00] (-0.05)	3.085	[0.05] (0.07)
	0.40	1	5.312	5.308	[0.09] (-0.00)	5.419	[0.13] (0.02)
		2	6.920	6.946	[0.00] (0.00)	7.392	[0.05] (0.07)
42	0.20	1	1.617	1.655	[0.08] (0.02)	1.705	[0.07] (0.05)
		2	2.212	2.055	[0.00] (-0.07)	2.131	[0.05] (-0.04)
	0.40	1	4.582	4.493	[0.08] (-0.02)	4.633	[0.07] (0.01)
		2	6.248	5.943	[0.00] (-0.05)	6.066	[0.08] (-0.03)
44	0.20	1	1.110	1.058	[0.09] (-0.05)	1.113	[0.07] (0.00)
		2	1.690	1.612	[0.00] (-0.05)	1.919	[0.05] (0.14)
	0.40	1	3.948	3.894	[0.08] (-0.01)	4.016	[0.08] (0.02)
		2	5.647	5.422	[0.00] (-0.04)	5.606	[0.07] (-0.01)

Table 4.1: Comparison of SPLS and NNM with a finite difference method for a single exercise American put option. The option can be exercised 50 times per year, the strike price is 40, and the risk-free rate is 0.06. The initial stock price  $S_0$ , the volatility  $\sigma$ , and the maturity  $T$  are as indicated. The finite difference numbers are from Longstaff and Schwartz (2001). The filtration energy is reported in bracket, and the relative difference to the finite difference price in parenthesis. SPLS uses  $\tilde{N} = 1,000$  simulations, an energy tolerance of 0.1, an energy validation sample of size  $N' = 1,000$ , and an hyperparameters grid size of 25 with  $(N_{\max}, m_{\max}) = (200, 200)$ . NNM uses  $N = 1,000$  simulations, an energy tolerance of 0.4, an energy validation sample of size  $N' = 10,000$ , a lower bound barrier of 1.02, and an hyperparameters grid size of 30 with  $(p_{\max}, q_{\max}) = (10, 500)$ .

## 4.1 SPLS Analysis

For an American put, an exercise strategy is optimal if the strategy captures the exercise boundary. Indeed, the optimal exercise strategy depends only on which side of the exercise boundary the last observed value of the stock fall (Carr, Jarrow, and Myneni 1992). See Figure 4.1 for an example of exercise boundary under different volatility. Looking at how SPLS approximates the exercise boundary should hence provide many insights on the algorithm, and will be the starting point of our analysis. Then, we look at the convergence of SPLS with respect to the projected strategy quality and the filtration energy. Finally, we look at the volatility of the SPLS estimates.

### 4.1.1 Exercise Boundary

Consider the projected lookahead operator at time  $t$  (2.9). This operator  $\bar{L}(\bar{X}_{t,T}, F_t)$  is a function of the strategy class and the filtration, and is hence a random variable. For this example, the operator is a random variable in  $\{0, 1\}$ . One way to understand the operator is to look at the distribution of the operator and to contrast this distribution with the exercise boundary. Indeed, even if the operator uses the filtration, the operator will be accurate only if it maps accurately the filtration  $F_t$  to the last observed stock value  $S_t$ , so that the distribution of the operator captures the exercise boundary.

Figure 4.2 shows the distribution of the projected lookahead operator at different time period. A first observation from this figure is that when the filtration energy is low, the projected operator accuracy increases, and the option is exercised with high probability when the stock crosses the exercise boundary. A second observation is that for some hyperparameters, the projected operator sometimes exercises the option before the stock crosses the exercise boundary. While with some others hyperparameters, such a too early exercise is not observed, but many exercise opportunities are missed. These distributions are two extremes, and the convergence theory of the previous section guarantees the existence of intermediate distributions. A third observation is that hyperparameters with similar filtration energy may perform differently. This is because the filtration energy certifies that the projected operator distribution has converged, but certifies nothing for the



projected strategy convergence. This situation is related to the difference in convergence impact between the projected operator convergence (2.16) and the projected strategy convergence (2.15). The next section analyses more the difference between these two convergence factors.

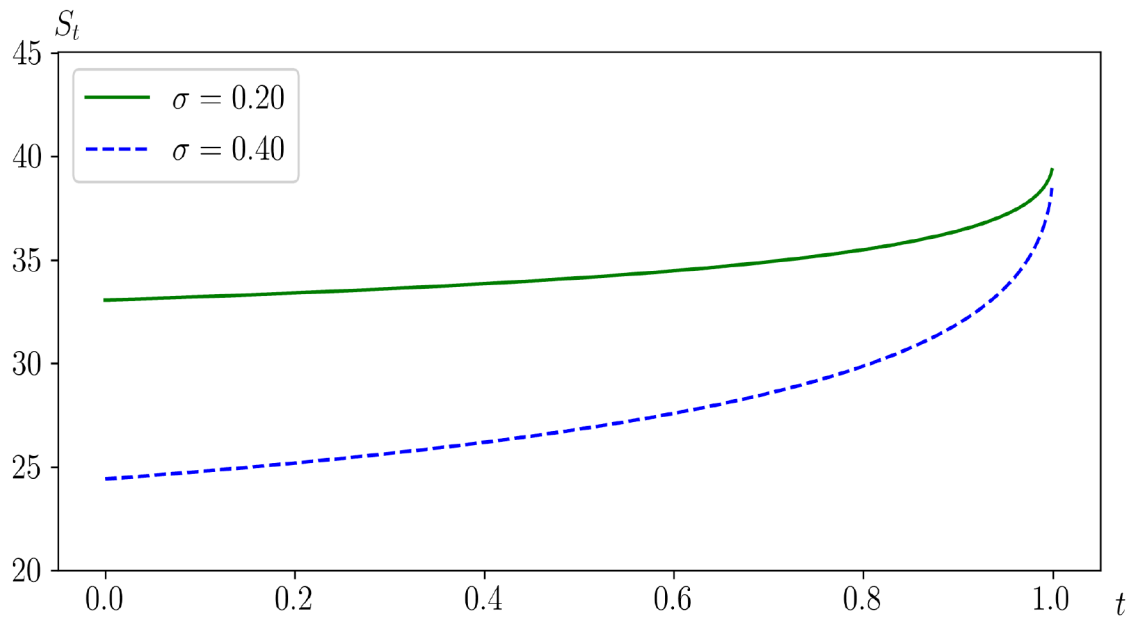
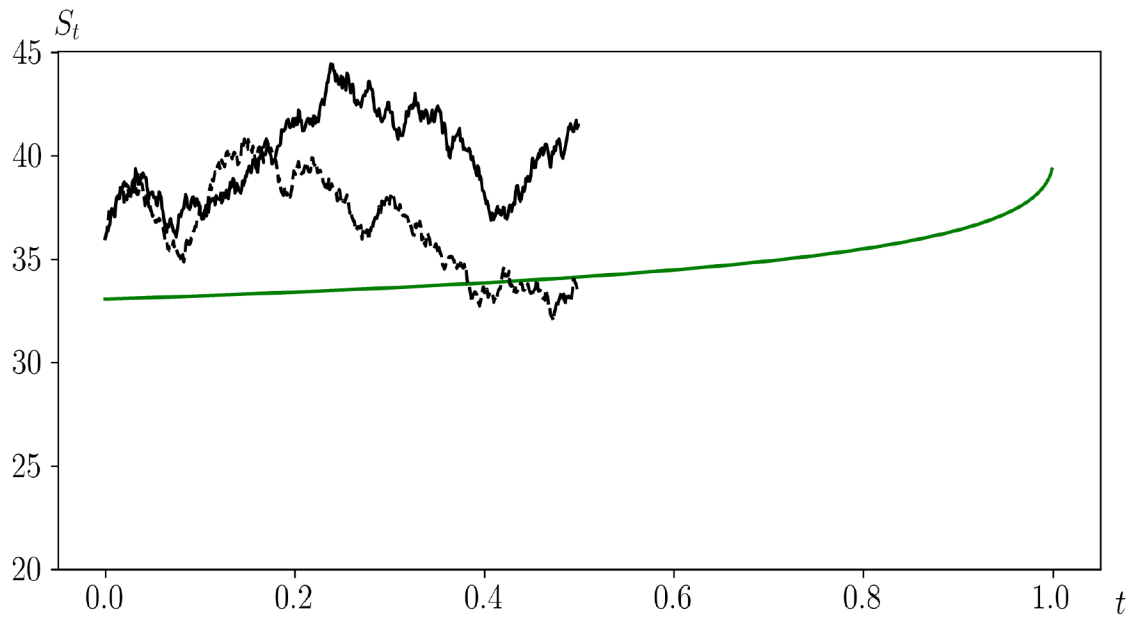


Figure 4.1: Exercise boundary for an American put under different volatility. The exercise boundary is a function of the time to maturity, and the last observed value of the stock. The x-axis is the time period  $t$ , and the y-axis is the stock price  $S_t$  at time  $t$ . In the top figure, two sample stock paths are shown: one that crosses the boundary, and for which the optimal strategy is to exercise the option at the crossing moment; a second one that never hits the boundary, and for which the optimal strategy is to never exercise the option. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, and the maturity is one year. The volatility  $\sigma$  is as indicated. The exercise boundary is found with an implicit finite difference scheme with 1,000 time steps per year and 10,000 steps for the stock price.

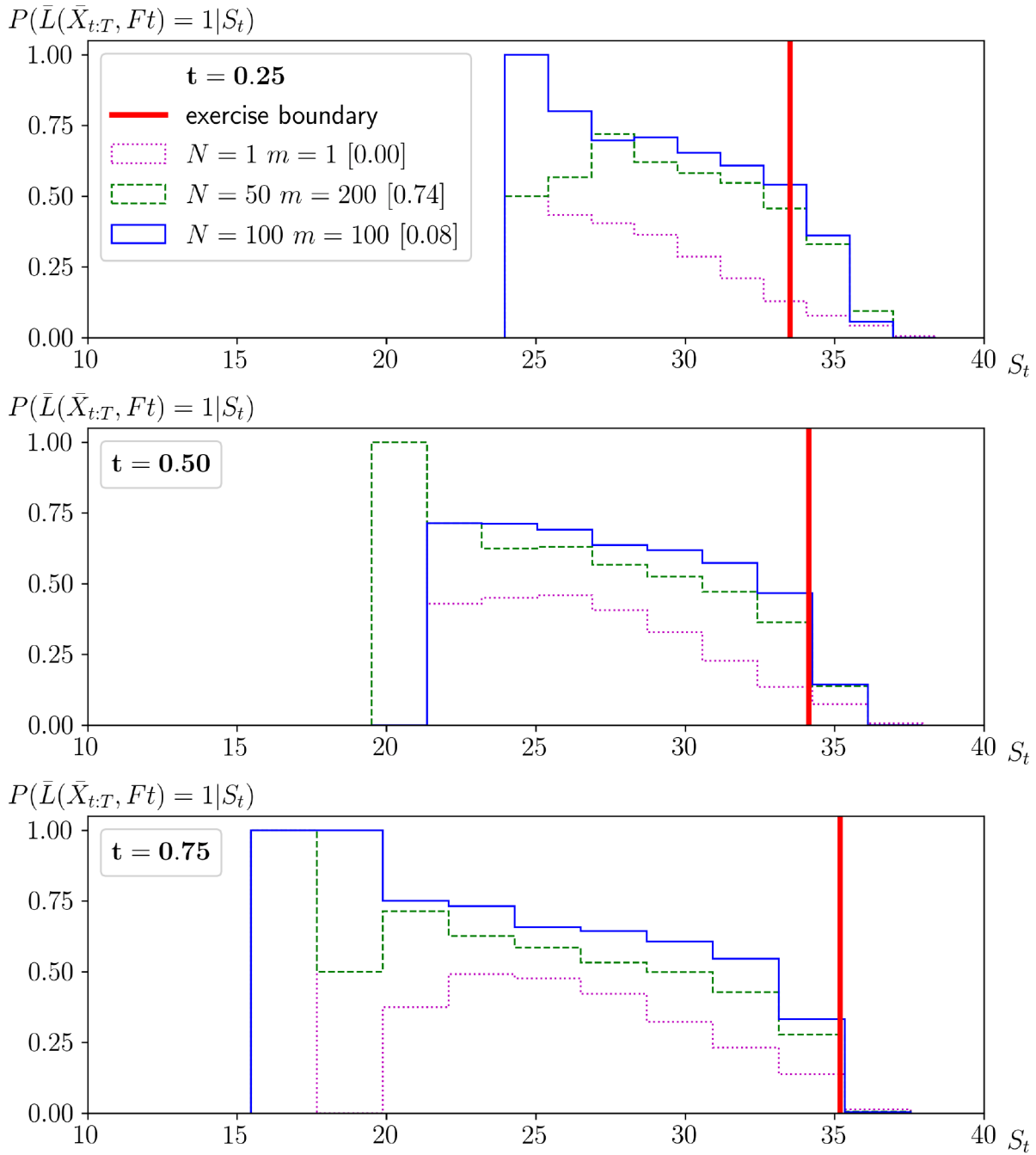


Figure 4.2: Distribution of the projected lookahead operator at different time points. The time  $t$ , the number of samples  $N$  in the projected operator, and the nearest-neighbor basis size  $m$  are as indicated. The energy of the hyperparameters is indicated in square bracket, the exercise boundary is indicated with a vertical line. The x-axis is the stock price  $S_t$ , and the y-axis is the empirical conditional probability that the projected operator exercises the option. The empirical probability is estimated with 100,000 simulations, and with an initial stock price  $S_0$  drawn uniformly at random from the interval  $[35, 45]$ . The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the volatility is 0.20, and the maturity is one year.

### 4.1.2 Projected Strategy

The quality of a projected strategy depends on the granularity of the underlying nearest-neighbor basis. The basis should be fine enough to differentiate on which side of the exercise boundary the last observed stock price falls. As SPLS enforces the exercises constraints almost surely, we need in fact a stronger granularity: the basis should be fine enough to differentiate on which side of the exercise boundary every previous stock prices fall. Indeed, for a lookahead at time  $t = 0.1$ , consider three projected decisions  $(\bar{X}_{0.25}^2, \bar{X}_{0.5}^1, \bar{X}_{0.75}^3)$ , where  $\bar{X}_{0.25}^1$  means that the Voronoi cell underlying the decision is the cell generated by the first centroid  $\bar{S}_{0.1:0.25}^{0.25,1}$ , see (2.8) for the notation. Consider also two sample paths  $(S_{0.25}^*, S_{0.5}, S_{0.75})$  and  $(S'_{0.25}, S'_{0.5}, S'_{0.75})$ , where  $S_{0.5}^*$  means that the sample path hits the exercise boundary at time  $t = 0.5$ . Now, note that if the two sample paths fall in the same Voronoi cell, the following two constraints cannot be enforced if the projected decisions are optimal

$$\begin{aligned}\bar{X}_{0.25}^2(S_{0.25}^*) + \bar{X}_{0.5}^1(S_{0.5}) + \bar{X}_{0.75}^3(S_{0.75}) &\leq 1, \\ \bar{X}_{0.25}^2(S'_{0.25}) + \bar{X}_{0.5}^1(S'_{0.5}) + \bar{X}_{0.75}^3(S'_{0.75}) &\leq 1.\end{aligned}$$

By construction, the decisions  $\bar{X}_{0.25}^1$  and  $\bar{X}_{0.75}^1$  must be the same for every path that fall in the underlying Voronoi cell, and the two constraints will be of the form  $(2 \leq 1)$ , if the projected decisions are optimal. Figure 4.3 illustrates this situation.

The projected strategy convergence rate (2.15) guarantees that the previous ambiguous situation is rare in probability when the nearest-neighbor basis size  $m$  is large. Two questions then arise: how big a basis is needed to avoid ambiguity? and how much ambiguity can be tolerated in the projected lookahead operator? These questions are answered by the Bayes-value convergence rate (2.19). Here, we provide a different insight on these questions, by defining an ambiguity metric specific to single exercise American option.

For the option of this example, a specific measure of ambiguity is the number of ambiguous constraints. For a projected lookahead at time  $t$ , and a nearest-neighbor basis size  $m$ , this measure can

be written with

$$\widehat{AM}_t = \frac{1}{N_a^2} \sum_{i=1}^{N_a} \sum_{j=1}^{N_a} \llbracket c(\bar{X}, S^i) \text{ and } c(\bar{X}, S^j) \text{ are infeasible if } \bar{X} \text{ is optimal} \rrbracket,$$

where  $c(\bar{X}, S^i)$  is the exercise constraint  $\bar{X}_0(S^i) + \bar{X}_1(S^i) + \dots + \bar{X}_T(S^i) \leq 1$ , and  $S^i$  is a path in a sample of size  $N_a$ . The event in the metric can be checked by taking every projected decision as optimal. For example, the decision  $\bar{X}_1(S^i)$  is taken as exercising the option only if the sample path  $S^i$  hits the exercise boundary at time 1. Intuitively, the ambiguity metric will be low if the nearest-neighbor basis as many Voronoi cells around the exercise boundary, so that the decision  $\bar{X}_t$  can differentiate between a sample path for which the option has already been exercised, and a sample path for which the option is exercisable for the first time at time  $t$ . A total ambiguity metric can be defined as the average ambiguity

$$\widehat{AM} = \frac{1}{T} \sum_{t=0}^T \widehat{AM}_t.$$

The total ambiguity is a proxy score for the convergence of the projected strategy. This score can be used as a substitute to the projected strategy convergence rate (2.15) to prove the convergence in probability of SPLS (2.18).

Figure 4.4 shows the exercise distribution and the ambiguity metric for different hyperparameters. For a projected lookahead at time  $t$ , the exercise distribution is an histogram of the empirical conditional probability  $P(\bar{L}(\bar{X}_t, F_t) | S_t)$ , see Figure 4.1.1 for reference. For Figure 4.4, the histogram probabilities are represented with the point size: point with larger size have a higher conditional probability of exercise. This convention can be written with

$$\text{point size at time } t \propto P(\bar{L}_t(\bar{X}_t, F_t) = 1 | S_t),$$

where the conditional exercise probability is estimated with a sample of projected lookahead operator.

A first observation from Figure 4.4 is that when the stock is closed to be in-the-money, the projected

operator may exercise too early. However, the conditional exercise probability of a too early exercise is small, and the probability that the stock is close to be in-the-money without crossing the exercise boundary is also small. These two factors allow the SPLS prices in Table 4.1 to be good estimates of the option value. A second observation is that the ambiguity metric depends on the sample size  $N'$  used to compute the metric. For the sample size used in the projected lookahead operator, which is of the order of 100, the ambiguity metric of order 1,000 suggests that very few sample paths lead to ambiguous constraints. A third observation is that the ambiguity metric differentiates hyperparameter quality. This can be seen by the hyperparameter  $(N, m) = (100, 100)$  always exercising when the option is in-the-money, while the hyperparameter  $(1, 1)$  exercises less. When two hyperparameters have a low ambiguity metric such as  $(100, 100)$  and  $(50, 200)$ , the best hyperparameter has a lower filtration energy. These observations can be summarized as follows. The nearest-neighbor size  $m$  defines a discretization of the strategy space, while the sample size  $N$  defines a distribution on this discretization. A finer discretization is better only if the distribution quality is maintained with a higher sample size.

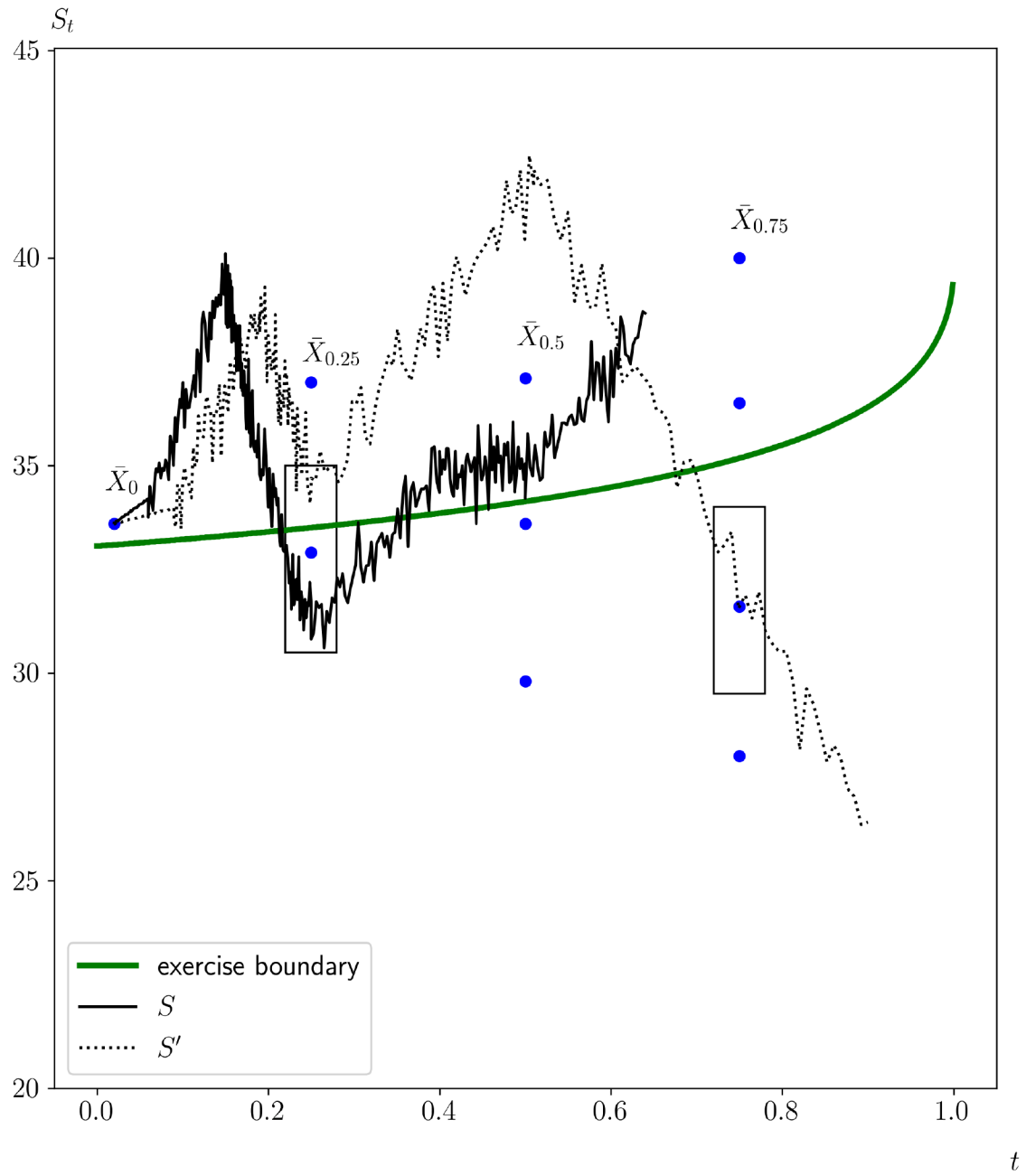


Figure 4.3: A projected strategy with ambiguous cell. For the sample path  $S$ , the optimal strategy is to exercise the option at time 0.25, while for the sample path  $S'$ , the optimal strategy is to exercise the option at time 0.75. However, both sample paths fall in the same Voronoi cell at time 0.25, and exercising the optimal decision for both sample paths is infeasible for the projected lookahead operator. The x-axis is the time period  $t$ , and the y-axis is the stock price  $S_t$  at time  $t$ . For the projected strategy  $\bar{X}$ , the Voronoi centroids are indicated by dots, and the Voronoi cells are indicated by rectangles. The exercise boundary is as indicated.

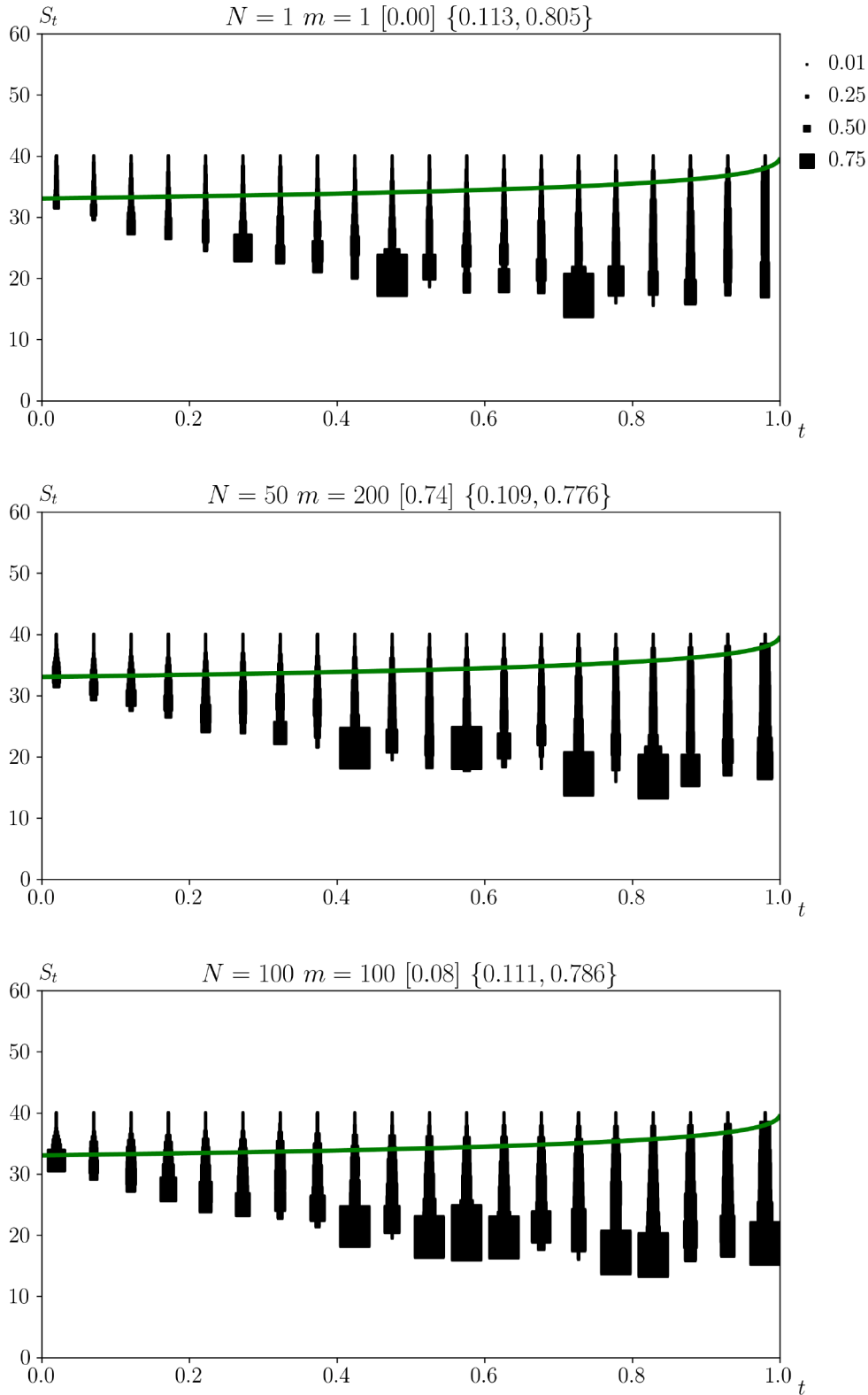


Figure 4.4: Distribution of the projected lookahead operator over time. The empirical conditional probability of exercise  $P(\bar{L}(\bar{X}_{t:T}, F_t)|S_t)$  is represented in the figure with a linewidth gradient. The exercise boundary is the bold curve. The sample size  $N$  in the projected lookahead operator, and the nearest-neighbor basis size  $m$  are as indicated. The filtration energy is indicated in square bracket. The ambiguity metric is indicated in braces. The first number in the brace is the ambiguity metric computed with a sample of 1,000 simulations, while the second number is computed with 10,000 simulations. The simulations are done with an initial stock price  $S_0$  draw uniformly at random from the interval  $[35, 45]$ . The x-axis is the time period  $t$ , and the y-axis is the stock price  $S_t$  at time  $t$ .



### 4.1.3 Estimates Volatility

With the projected lookahead operator being a random variable, SPLS uses a random strategy to approximate the optimal strategy: at each time period, the exercise decision is drawn at random from a certain probability distribution. As no value function and no policy function are learned in the course of sampling, the sampling distribution is fixed and should converge quickly to an estimate of the option value. Figure 4.5 confirms this intuition. When started from any random seed, any given set of hyperparameters converges quickly to the same value. This observation is supported by the projected option value convergence rate (2.17) and the Bayes-value convergence rate (2.19).

By using a random strategy that converges quickly, one approach in using SPLS is to use several small samples, instead of a single large sample. This a bootstrap procedure: use a small sample size  $\tilde{N} = 1,000$ , instead of a large sample size  $\tilde{N} = 10,000$ , and estimate the price with the highest estimate obtained. Figure 4.5 and Table 4.1 show that this approach works, especially if different hyperparameters are used in each small sample. This approach is massively parallel and can provide fast and accurate estimates.

The convergence theory of the previous section guarantees that there exists hyperparameters for SPLS that can approximate the price of any option to any degree of accuracy. Such hyperparameters can be found with a fine grained hyperparameters tuning. For instance, Table 4.1 could be obtained with a higher precision. To show that this is possible, Figure 4.6 shows the convergence of a high accuracy hyperparameter set for one of the prices in Table 4.1 for which the estimation error was 5%. The distribution of the projected lookahead operator is also displayed.

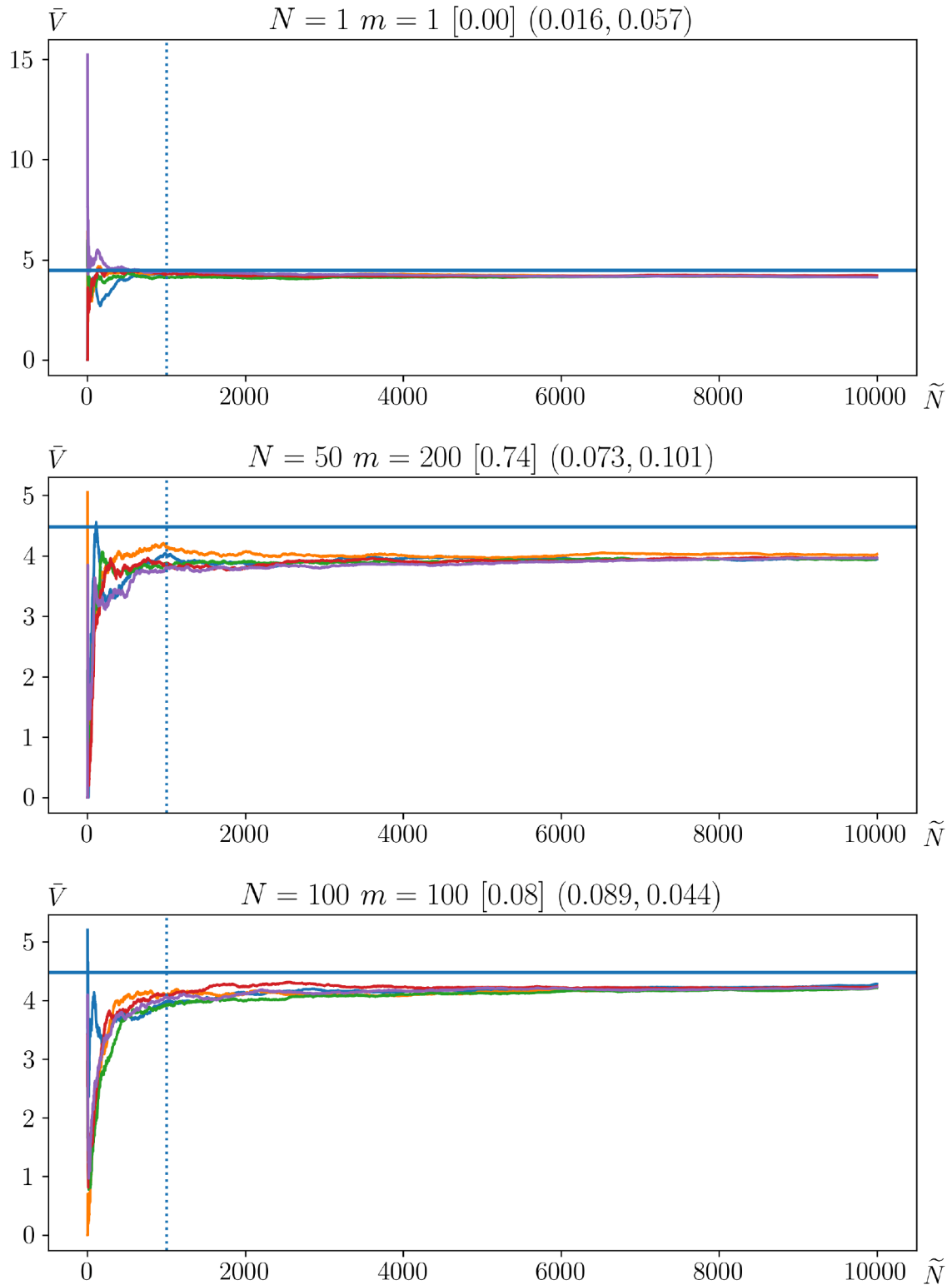


Figure 4.5: Convergence of the projected option value for different random seed. The number of sample  $\tilde{N}$  in the projected option value, the sample size  $N$  in the projected lookahead operator, and the nearest-neighbor basis size  $m$  are as indicated. The energy of the hyperparameters is indicated in square bracket, and the finite difference price is the horizontal line. The best relative difference between the project option value and the finite difference price is indicated in parenthesis. The first number in parenthesis is the best difference after a 1,000 sample, and the second number is after 10,000. The vertical line indicates the 1,000 sample level. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the volatility is 0.20, the maturity is one year, and the initial stock price is 36.

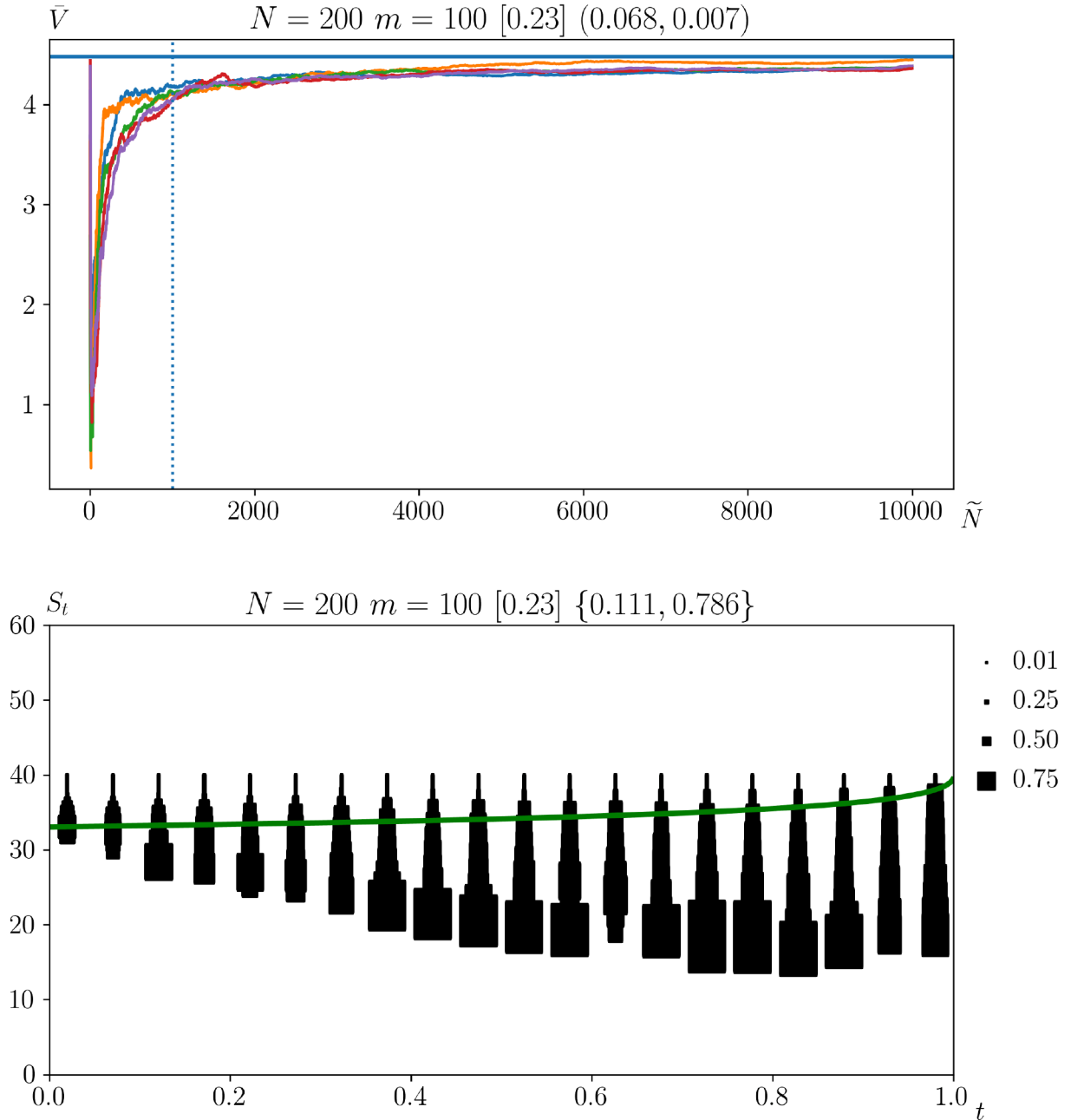


Figure 4.6: **Top.** Convergence of the projected option value for different random seeds with a high accuracy projection scheme. The number of samples  $\tilde{N}$  in the projected option value, the sample size  $N$  in the projected lookahead operator, and the nearest-neighbor basis size  $m$  are as indicated. The energy of the hyperparameters is indicated in square bracket, and the finite difference price is the horizontal line. The best relative difference between the project option value and the finite difference price is indicated in parenthesis. The first number in parenthesis is the best difference after a 1,000 sample, and the second number is after 10,000. The vertical line indicates the 1,000 sample level. The energy validation sample is of size  $N' = 2,000$ . The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the volatility is 0.20, the maturity is one year, and the initial stock price is 36. **Bottom.** Distribution of the projected lookahead operator over time for the hyperparameters set in the top figure. The empirical conditional probability of exercised  $P(\bar{L}(\bar{X}_{t:T}, F_t)|S_t)$  is represented in the figure with a linewidth gradient. The exercise boundary is the bold curve. The sample size  $N$  in the projected lookahead operator, and the nearest-neighbor basis size  $m$  are as indicated. The filtration energy is indicated in square bracket. The ambiguity metric is indicated in braces. The first number in the brace is the ambiguity metric computed with a sample of 1,000 simulations, while the second number is computed with 10,000 simulations. The simulations are done with an initial stock price  $S_0$  drawn uniformly at random from the interval  $[35, 45]$ . The x-axis is the time period  $t$ , and the y-axis is the stock price  $S_t$  at time  $t$ .

## 4.2 NNM Analysis

For any option, a dual martingale is optimal if the martingale is close to the martingale part of the value function (Rogers 2007, Rogers 2002, Haugh and Kogan 2004, Andersen and Broadie 2004). There are many senses in which two stochastic processes can be close. For an American put, we can look at the behavior of a dual martingale across the exercise boundary by quantizing the martingale. In such a perspective, two martingales are close if the distribution of their behavior across the exercise boundary is similar. This sense of closeness provides many insights on NNM and will be the starting point of our analysis. Then, we look at the volatility of the NNM estimates.

### 4.2.1 NNM Martingale

An NNM martingale is the martingale produced by the projected Rogers operator for some choice of hyperparameters. We will compare the behavior of several NNM martingales to the optimal martingale across the exercise boundary. For an American put, the optimal martingale can be obtained as follows. With an implicit finite difference method, obtain the value function. Then, obtain the martingale part of the value function by nested simulation along a sample stock path. The resulting martingale is optimal. See Methods for more details. Figure 4.7 shows an example of sample stock path, the corresponding value function path, and the corresponding martingale part path.

To study the distribution of a martingale across the exercise boundary, we quantize a sample of martingale path with a small number of centroids. As each martingale path is generated by a stock path, the martingale centroids define an implicit tessellation on the sample stock path. This implicit tessellation can be used to map any martingale metric to the regions of the exercise boundary. We call such a mapped martingale metric a *dual metric*. See Methods for more details. Figure 4.8 shows the distribution of three dual metrics: the typical stock trajectory, the dual exercise decision and the  $\ell_1$ -average of the optimal martingale. The typical stock trajectory within a cell of the implicit tessellation, is the average of the stock path that falls within the cell. In the top figure, the linewidth gradient is the distribution of the dual exercise over time. To define the dual exercise, recall that

to each stock path is associated the dual payoff

$$\max_x \sum_{t=0}^T e^{-rt} (K - S_t^i)_+ x_t - x_t M_t^i, \quad (4.3)$$

where  $S^i$  is a sample path, and  $M^i$  is the corresponding martingale path. Let  $t^i$  be the optimal exercise time in (4.3), and let  $S_{t^i}^i$  be the stock price at the dual exercise time  $t^i$ . The distribution of the dual exercise time is an histogram of a sample  $\{(S_{t^i}^i, t^i)\}$  of such dual exercise. The figure presents this distribution along the Voronoi cell. In the bottom figure, the linewidth gradient is the  $\ell_1$ -average value of the optimal martingale over time. To define this  $\ell_1$ -average, consider a particular Voronoi cell, and let  $\{M^i\}$  be the set of martingale path that falls within the cell. The  $\ell_1$ -average at time  $t$  can be written with

$$\frac{1}{n} \sum_{i=1}^n |M_t^i|, \quad (4.4)$$

where  $n$  is the number of martingale sample path that falls within the cell.

Figure 4.8 shows that the optimal martingale forces the dual exercise to be in the out-of-the-money region. To achieve this, the martingale takes large positive value in the in-the-money region, and small negative value in the out-of-the-money region. The graph shows only the average absolute value of the optimal martingale, but the martingale value sign can be inferred by whether the stock path centroid is in-the-money or out-of-the-money.

To measure how close a candidate martingale is to the optimal martingale we can look at the dual metrics. For the comparison to be meaningful, the reference Voronoi tessellation should be fixed. We use the optimal martingale tessellation as the reference tessellation. See Methods for more details. Figure 4.9 shows the dual metrics for the NNM martingale with nearest-neighbor basis size  $(p, q) = (1, 1)$ . By comparing the dual exercise distribution in Figure 4.9 and Figure 4.8, we see that the  $(1, 1)$ -martingale does not prevent dual exercise in the in-the-money region. The dual exercise distribution for the martingale is uniform over the timeline. This difference in dual exercise distribution is explained by the  $\ell_1$ -average of the  $(1, 1)$ -martingale: the martingale is zero on every stock centroid. Dual exercise is hence possible everywhere. Note that Figure 4.9 shows that the

option is dually exercised when the stock path centroid is out-of-the-money. This is because the Voronoi cell is an aggregation of several stock paths and many stock path in the cell can be in-the-money. In fact, for the  $(1, 1)$ -martingale, every martingale path is identically zero, so that all stock paths fall in the same Voronoi cell.

Figure 4.10 shows the dual metric for a high precision NNM martingale. A first observation from this figure is that both the dual exercise time and the  $\ell_1$ -average metrics of the martingale are very close to the optimal martingale. They are some small difference as to where the NNM pushes the dual exercise time, and in the  $\ell_1$ -magnitude of the martingale, but the relaxed dual value accuracy is preserved. This is expected from the convergence theory of the previous section. NNM is an algorithm with probabilistic guarantees, and such guarantees tolerate small difference in distribution.

Comparing the  $\ell_1$ -average of an NNM martingale to the optimal martingale gives an idea of the NNM projected martingale quality. Indeed, if the two  $\ell_1$ -averages are close, we can infer that the nearest-neighbor basis of the projected martingale is sufficiently fine for the projected martingale to match in average the optimal martingale. This proxy for the projected martingale quality allows to draw some conclusions on the relation between the filtration energy and the projected martingale quality. To draw these conclusions, Figure 4.11 shows the dual metrics for a low precision NNM martingale. By comparing this figure to Figure 4.10, we see that the projected martingale quality is more important than the filtration energy for the relaxed dual value accuracy. Indeed, the filtration energy is a metric for the accuracy of the projected Rogers operator, and the operator can only be as much accurate as the projected martingale. In particular, the  $(10, 100)$ -martingale nearest-neighbor basis does not capture one of the optimal martingale centroid, and the dual exercise time for the stock path in this centroid is uncontrolled. This can be seen by the distribution of the dual exercise time being uniform on the middle stock centroid.

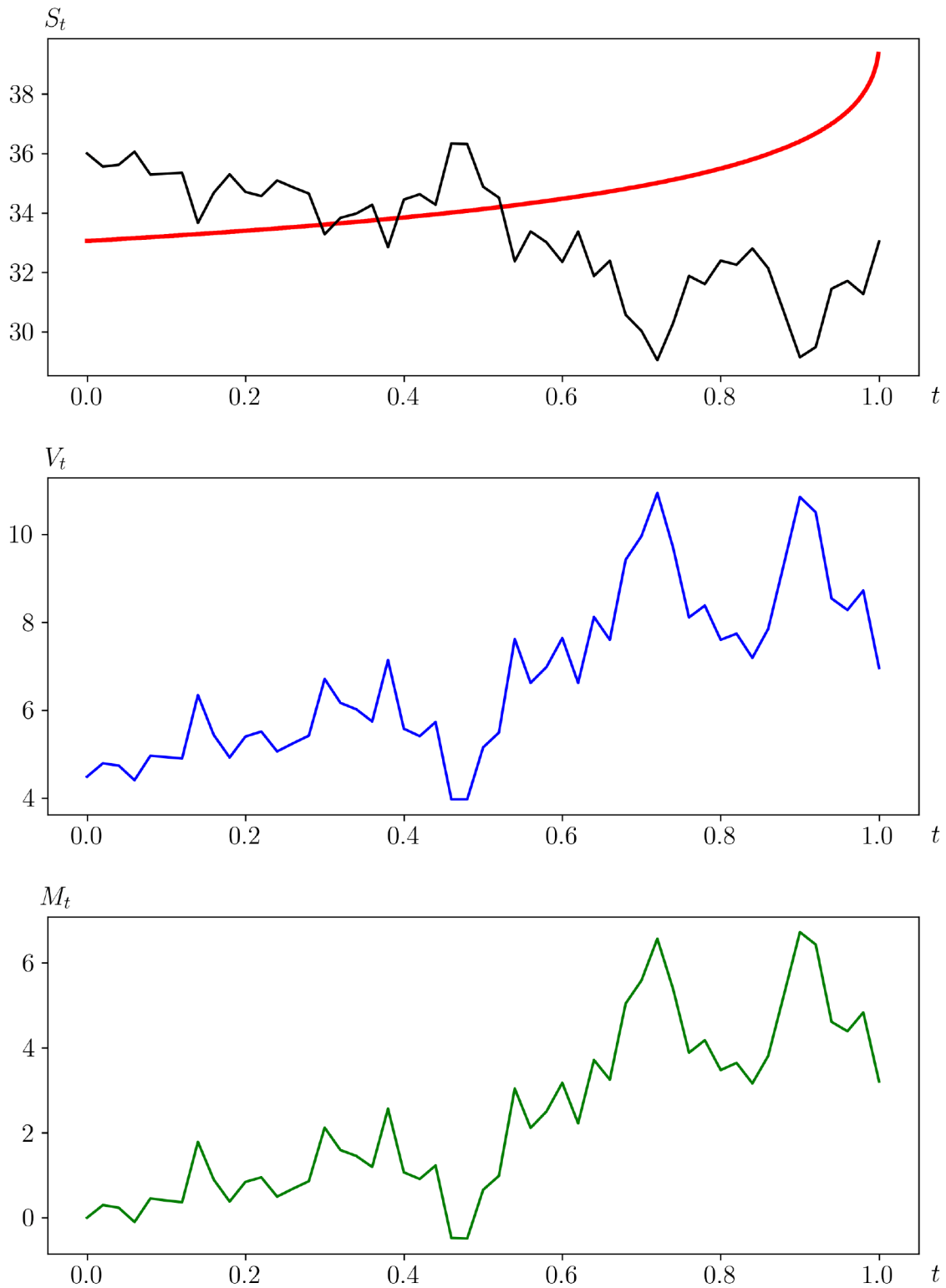


Figure 4.7: A sample stock path, the corresponding value function path, and the corresponding martingale part increment path. In the top graph, the exercise boundary is the bold line. The x-axis is the time period  $t$ , and the y-axis is either the stock price  $S_t$ , the value function  $V_t$ , or the martingale part  $M_t$  of the value function. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the maturity is one year, and the volatility is 0.20. The exercise boundary is found with an implicit finite difference scheme with 1,000 time steps per year and 10,000 steps for the stock price. The martingale part is found by nested simulation at each time period. Each nested simulation uses 10,000 sample of the next period stock price. See Methods for more details.

### Optimal Martingale {1.617}

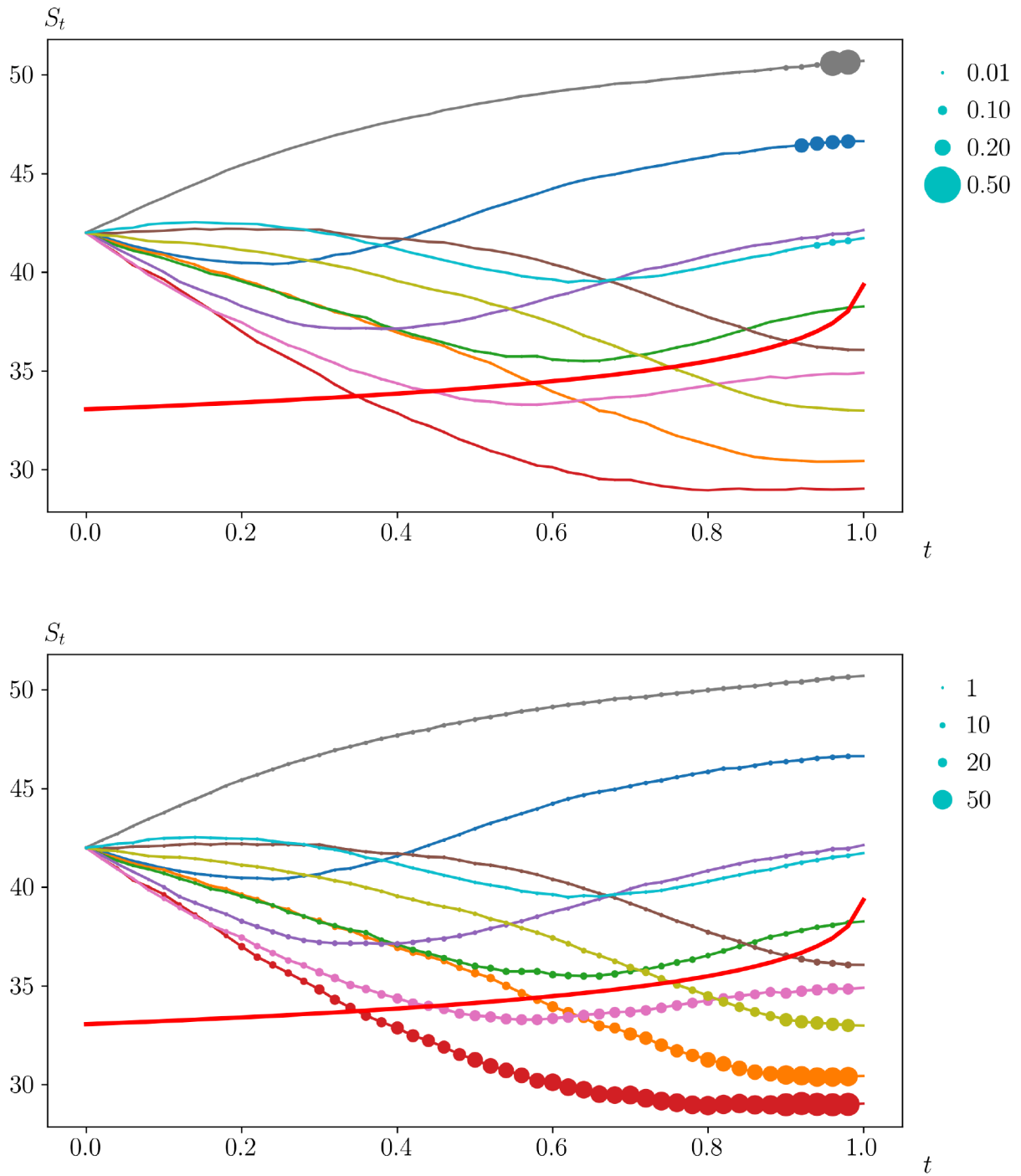


Figure 4.8: The dual exercise distribution and the  $\ell_1$ -average of the optimal martingale over a quantization of the optimal martingale. The sample stock paths shown are the implicit stock path centroids implied by the Voronoi tessellation of the optimal martingale. The Voronoi tessellation is obtained with a sample of 10,000 sample martingale path by finding 10 centroids with Lloyd's method. In the figure title, the price of the option is in brace. The exercise boundary is the bold line. The x-axis is the time period  $t$ , and the y-axis is the stock price  $S_t$ . In the top graph, the distribution of the dual exercise is represented with a linewidth gradient. In the bottom graph, the  $\ell_1$ -average of the optimal martingale is represented with a linewidth gradient. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the maturity is one year, the volatility is 0.20, and the initial stock price is 42. The exercise boundary is found with an implicit finite difference scheme with 1,000 time steps per year and 10,000 steps for the stock price. The martingale part is found by nested simulation at each time period. Each nested simulation uses 10,000 sample of the next period stock price. See Methods for more details.



NNM (1, 1) [0.00] {2.88}

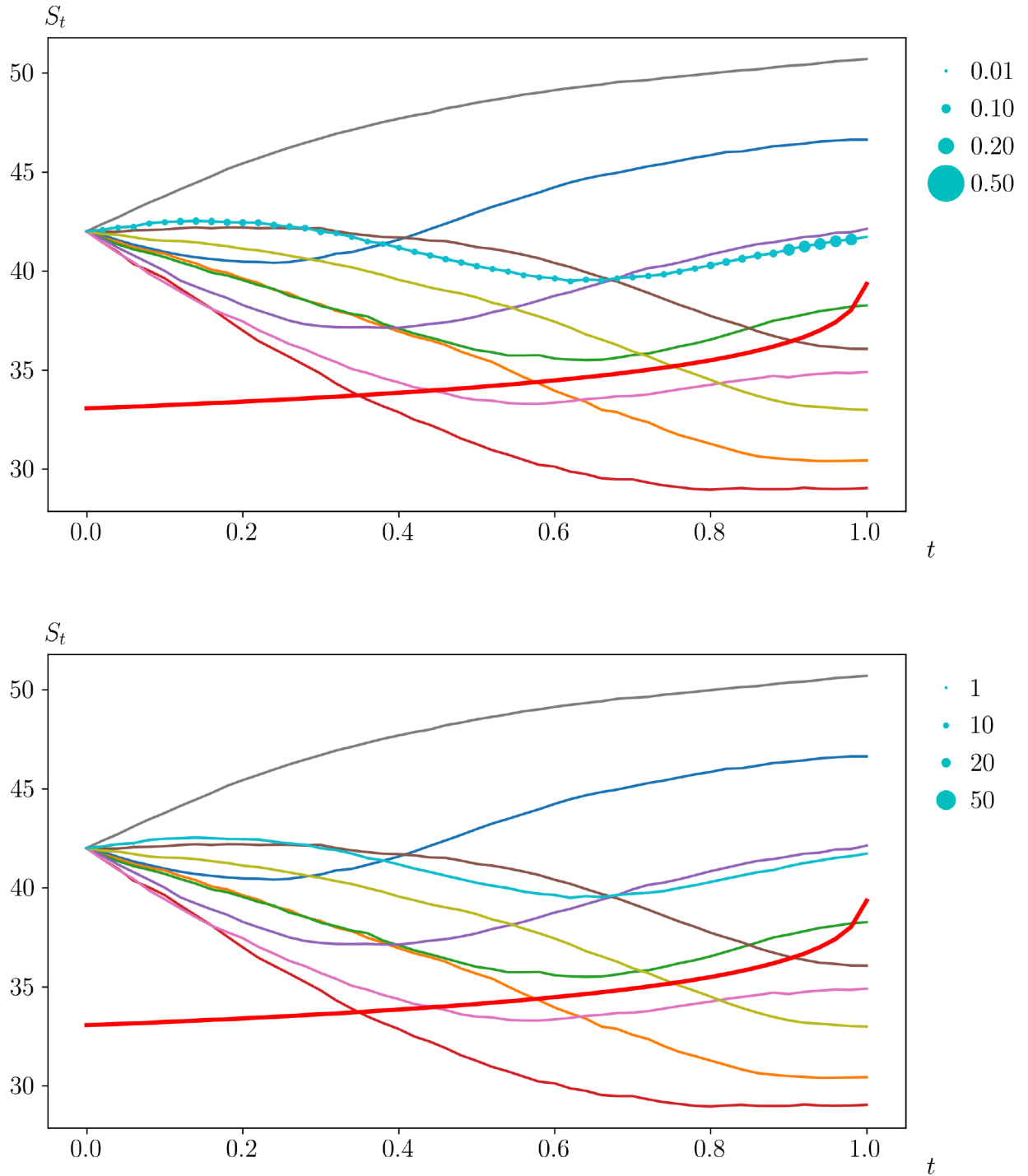


Figure 4.9: The dual exercise distribution and the  $\ell_1$ -average of the NNM martingale  $(p, q) = (1, 1)$  over a quantization of the optimal martingale. See Figure 4.8 for the quantization and notation details. In the figure title, the number in bracket is the filtration energy, and the number in brace is the relaxed dual value of the NNM martingale. NNM uses  $N = 1,000$  simulations, and an energy validation sample of size  $N' = 10,000$ .

NNM (10, 350) [0.20] {1.65}

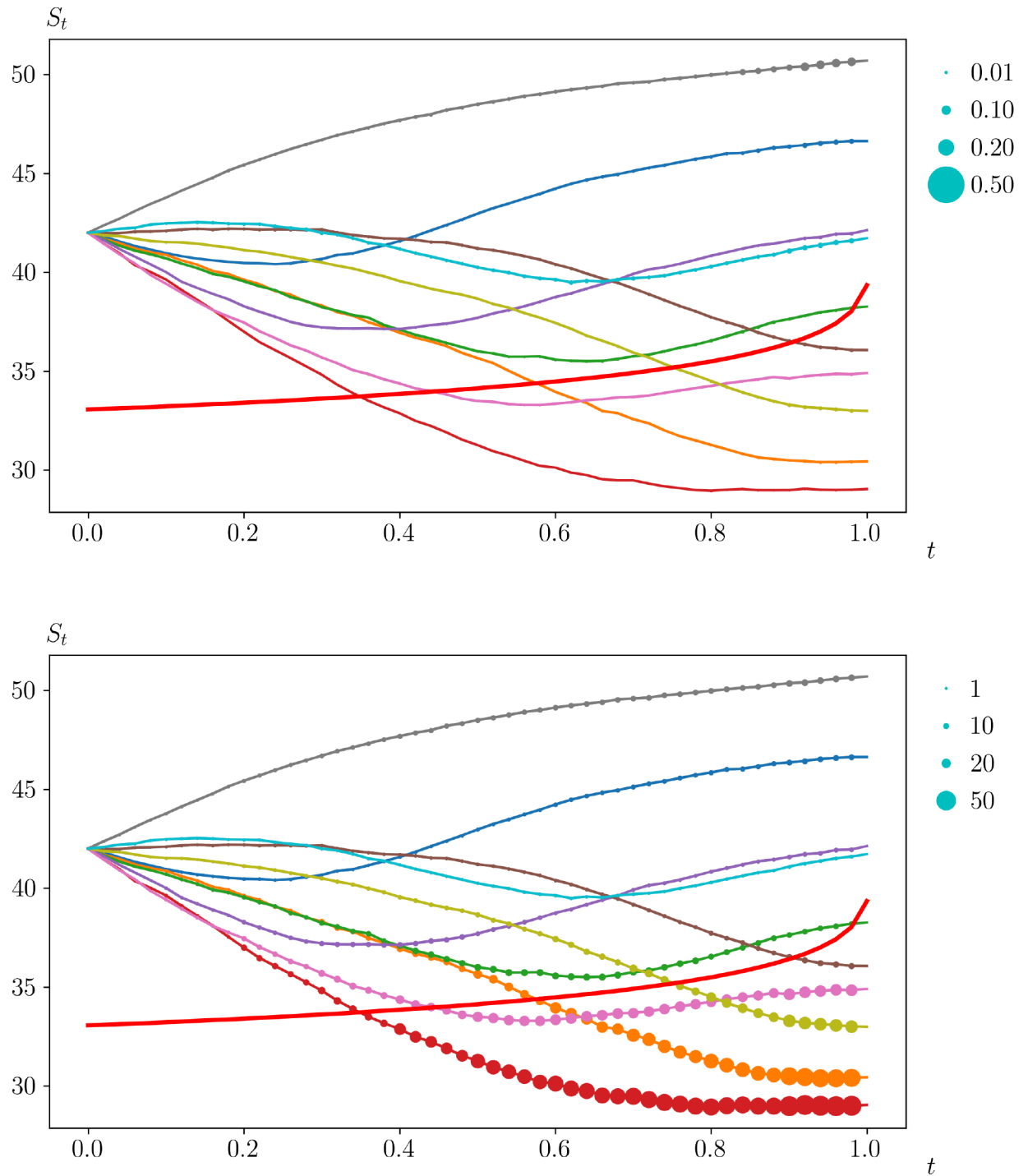


Figure 4.10: The dual exercise distribution and the  $\ell_1$ -average of the NNM martingale  $(p, q) = (10, 350)$  over a quantization of the optimal martingale. See Figure 4.8 for the quantization and notation details. In the figure title, the number in bracket is the filtration energy of the NNM martingale, and the number in brace is the relaxed dual value. NNM uses  $N = 1,000$  simulations, and an energy validation sample of size  $N' = 10,000$ .

NNM (10, 100) [0.03] {2.25}

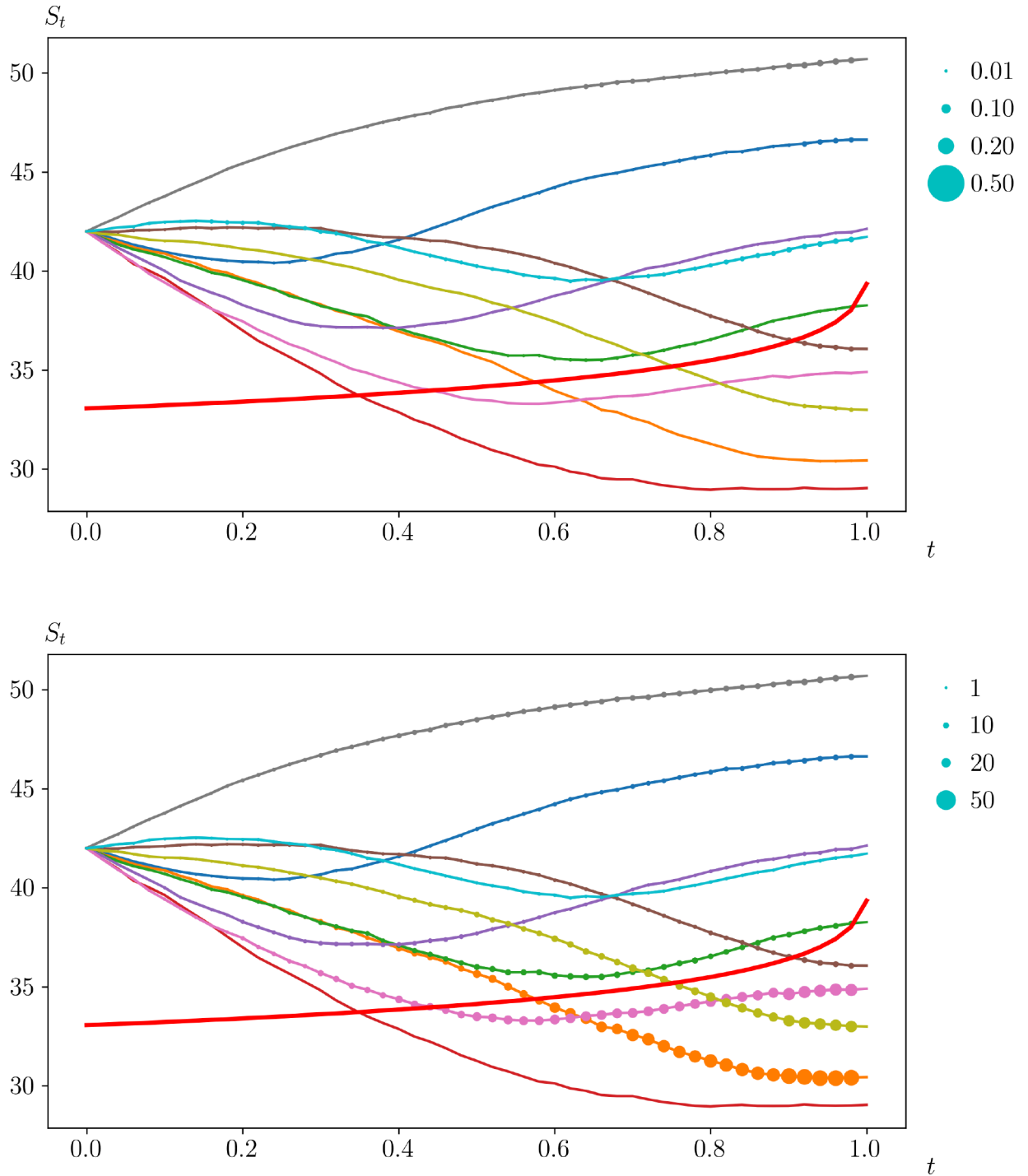


Figure 4.11: The dual exercise distribution and the  $\ell_1$ -average of the NNM martingale  $(p, q) = (10, 100)$  over a quantization of the optimal martingale. See Figure 4.8 for the quantization and notation details. In the figure title, the number in bracket is the filtration energy of the NNM martingale, and the number in brace is the relaxed dual value. NNM uses  $N = 1,000$  simulations, and an energy validation sample of size  $N' = 10,000$ .

### 4.2.2 Estimates Volatility

Figure 4.12 shows the convergence of NNM for various random seeds and for various hyperparameters. A first observation from this figure is that when the filtration energy is too low, the relaxed dual value is far above the dual value. From the previous section, we can infer that this is due to the projected martingale quality that is not sufficient to capture the optimal martingale behavior. A second observation is that when the nearest-neighbor size is large such as  $(p, q) = (10, 5000)$ , an equally large sample size  $N$  is needed for the convergence of the relaxed dual value. This observation can be contrasted with the observation that a good accuracy can be obtained with a small sample size  $N = 1,000$  when the filtration energy is adequate.

Figure 4.12 is informative for hyperparameters tuning, but the figure misses the key convergence behavior of NNM. Indeed, the figure approaches NNM as a probabilistic algorithm and the figure may give the impression that NNM does not converge with the sample size  $N$ . However, unlike SPLS, NNM is not a probabilistic algorithm, even if the convergence proof of NNM rests on probabilistic arguments. The Rogers operator is deterministic and the adequate perspective to study the convergence of NNM is by fixing the sample size  $N$ . From this perspective, the correct plane to analyse the convergence of NNM is not the sample size and dual value plane  $(N, V)$ . Instead, fix the sample size  $N$  and look at the nearest-neighbor size, energy and dual plane  $((p, q), \widehat{ED}, V)$ . A pattern then emerges and shows that the convergence of NNM rests mainly on the filtration energy. Figure 4.13 displays such a plane.

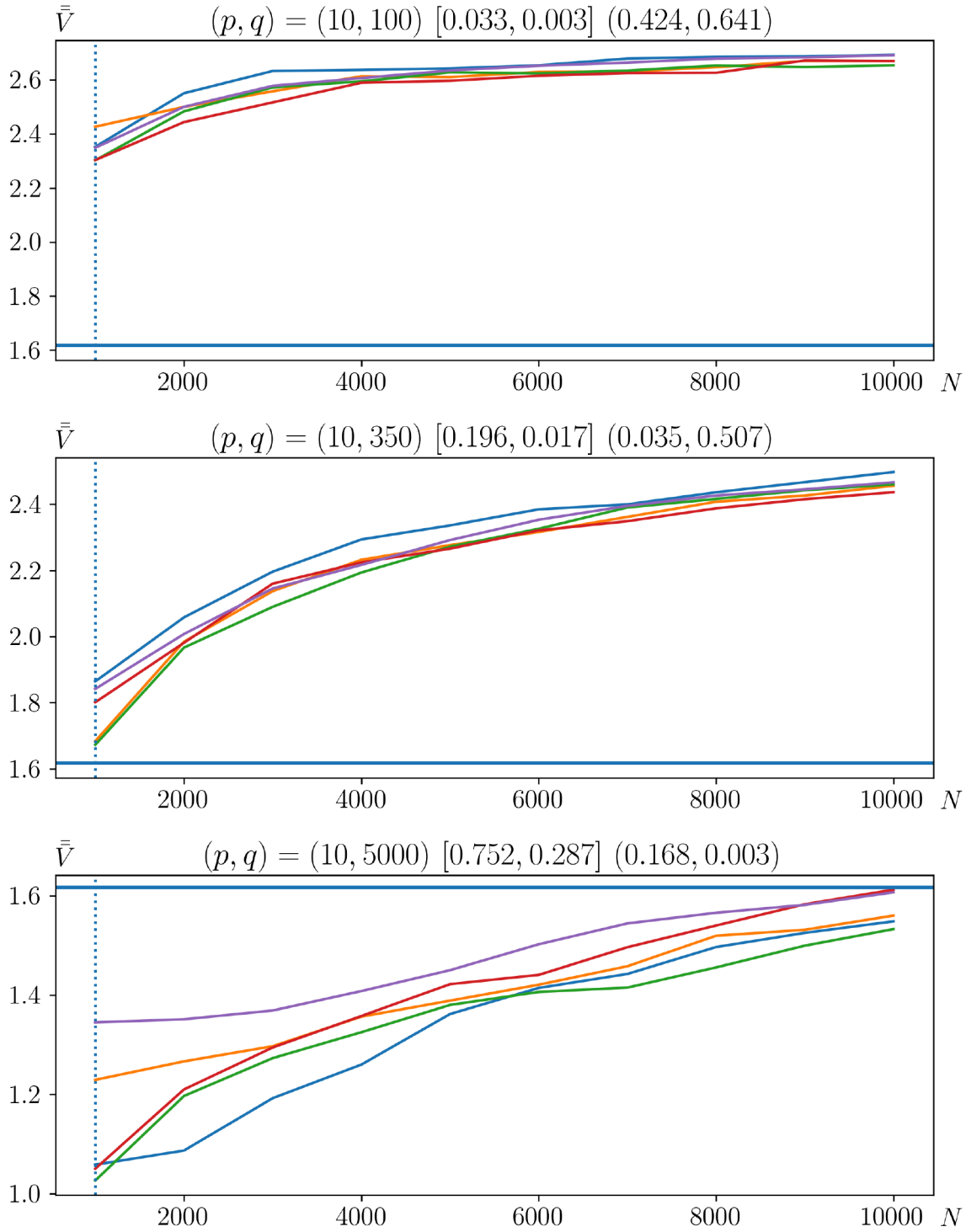


Figure 4.12: Convergence of the relaxed dual value for different random seed. The sample size  $N$  in the projected Rogers operator, and the nearest-neighbor basis size  $(p, q)$  are as indicated. The energy of the hyperparameters is indicated in square bracket. The first number in bracket is the energy with a sample size of  $N = 1,000$ , and an energy validation sample of  $N' = 10,000$ . The second number in bracket is the energy with a sample size of  $N = 10,000$ , and an energy validation sample of  $N' = 100,000$ . The finite difference price is the horizontal line. The best relative difference between the relaxed dual value and the finite difference price is indicated in parenthesis. The first number in parenthesis is the best difference for a sample size of  $N = 1,000$ , and the second number is for a sample size of  $N = 10,000$ . The vertical line indicates the  $N = 1,000$  sample size level. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the volatility is 0.20, the maturity is one year, and the initial stock price is 42.

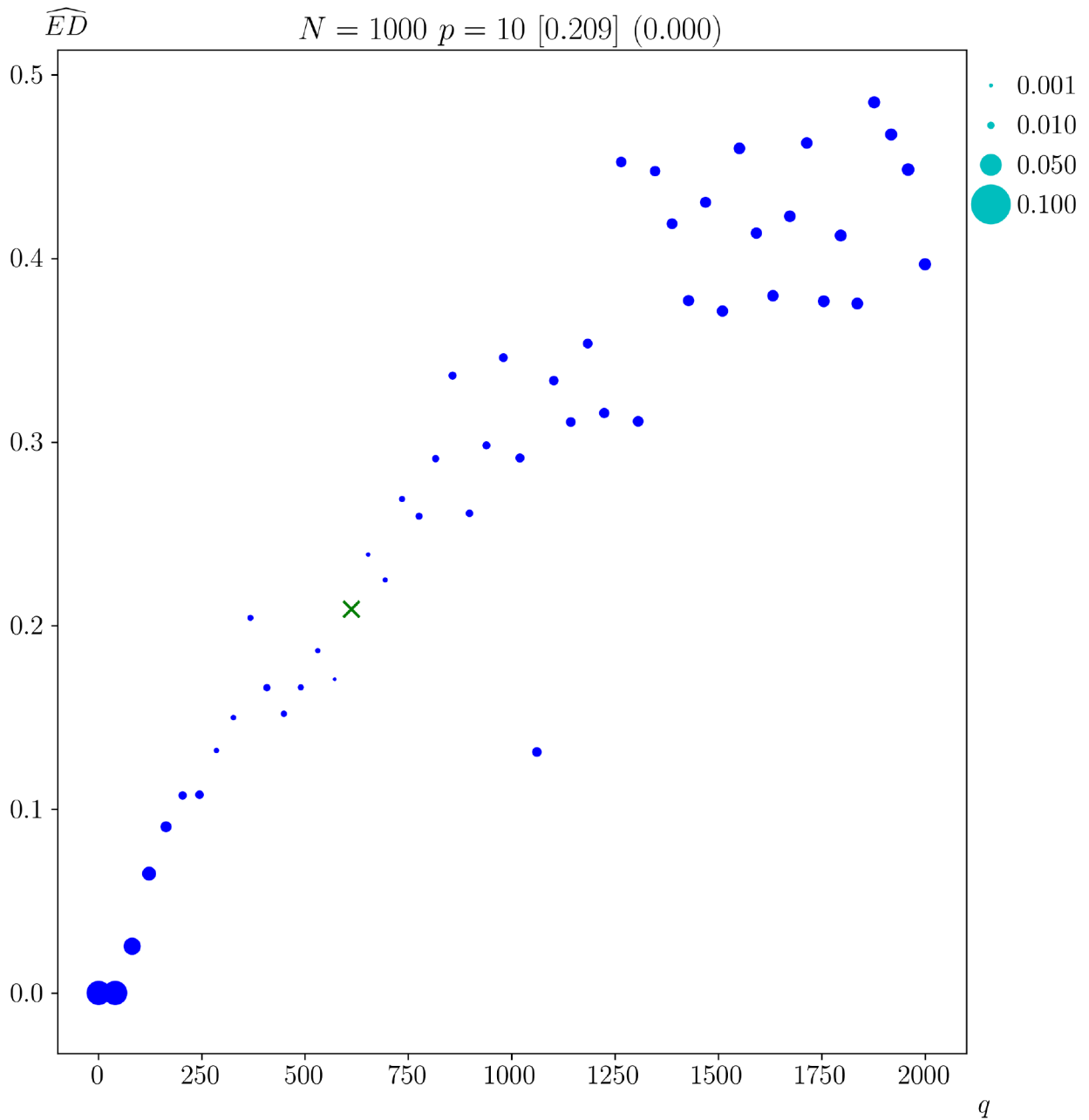


Figure 4.13: Convergence of NNM in the nearest-neighbor size and energy plane. The sample size  $N$  in the projected Rogers operator, and the conditioning nearest-neighbor size  $p$  are as indicated. The x-axis is the next nearest-neighbor size  $q$ , and the y-axis is the filtration energy  $\widehat{ED}$ . The energy validation sample is of size  $N' = 10,000$ . The point size gradient indicates the relative difference between the relaxed dual value and the finite difference price. The best relative difference is indicated by a cross, and, in the title, this best relative difference is indicated in parenthesis, along with the corresponding filtration energy in bracket. The NNM relaxed dual values are obtained by taking the best relative difference of five different random seeds. The option can be exercised 50 times per year, the strike price is 40, the risk-free rate is 0.06, the volatility is 0.20, the maturity is one year, and the initial stock price is 42.

## 5 A Swing Option

This section applies SPLS and NNM to a simple multiple exercise American option. The option is an American call on the price of electricity, with a constraint on the total number of exercises. This example is taken from Meinshausen and Hambly (2004).

Consider a multiple exercise American call option on the electricity price  $S$  with a strike price of zero. The option can be exercised at most on  $n$  days till the maturity of 50 days. With  $X$  the exercise decision, the stochastic program for the option can be written with

$$\max_X E \left[ \sum_{t=0}^{50} S_t X_t \right] \tag{5.1}$$

$$\text{s.t.} \quad \sum_{t=0}^{50} X_t \leq n \tag{5.1.1}$$

Assume further that the risk-neutral dynamics of the electricity price is given by the following AR(1) model

$$\log S_t = 0.1 \log S_{t-1} + \frac{1}{2} W_t, \tag{5.2}$$

with  $S_0 = 1$  and  $W$  a standard Brownian process.

In Meinshausen and Hambly (2004), the lower bound is obtained with the least-square Monte Carlo method of Longstaff and Schwartz (2001), and the upper bound is obtained by specializing the dual approach of Rogers (2002) and Haugh and Kogan (2004) to the option. The approach of Meinshausen and Hambly (2004) enables the pricing of this example through a theoretical extension of single exercise methods to the multiple exercise case. In contrast, SPLS and NNM are fully general algorithms and can be applied as is to this example. To further show the versatility of our algorithms, we will use a different hyperparameters tuning method than previously presented.

For SPLS, we tune the hyperparameters as follows. Fix a sample size  $N$  for the lookahead operator. Then, increase the nearest-neighbor size  $m$  until the projected dual value doesn't increase. This tuning approach aims at finding a better lower bound by increasing solely the projected strategy

quality. For NNM, we proceed as follows. Fix a conditioning nearest-neighbor size  $p$ , and increase the next nearest-neighbor size  $q$  until the first drop in relaxed dual value is recorded. When this drop is recorded, if the relaxed dual value is lower than the SPLS lower estimate, increase the sample size  $N$ , and repeat the procedure. This approach aims at finding the typing point in the accuracy of NNM, as suggested by Figure 4.13. We run this tuning procedure once, for the case when only a single exercise is allowed. Then, we keep the same hyperparameters for cases with multiple exercises. Table 5 shows the results of this approach.

$n$	Least-Square Monte Carlo	SPLS	NNM
1	2.750	2.748 (-0.00)	2.818 (0.02)
2	5.156	5.157 (0.00)	5.176 (0.00)
3	7.306	7.305 (-0.00)	7.301 (-0.00)
4	9.336	9.348 (0.00)	9.277 (-0.01)
5	11.230	11.255 (0.00)	11.121 (-0.01)
10	19.556	19.704 (0.01)	19.183 (-0.02)
15	26.488	26.860 (0.01)	25.892 (-0.02)
20	32.326	33.044 (0.02)	31.763 (-0.02)
25	37.697	38.458 (0.02)	37.007 (-0.02)
30	42.420	43.289 (0.02)	41.829 (-0.01)
35	46.337	47.627 (0.03)	46.118 (-0.00)
40	50.299	51.435 (0.02)	50.051 (-0.00)
45	53.335	54.755 (0.03)	53.439 (0.00)
50	56.765	57.439 (0.01)	55.898 (-0.02)

Table 5.1: Comparison of SPLS and NNM with a least-square Monte Carlo method for a swing option with a constraint on the total number of exercises. The option is a zero-strike American call on the electricity price that can be exercised at most  $n$  times till the maturity of 50 days. The least-square Monte Carlo price is obtained with the method of Meinshausen and Hambly (2004) and 1,000 simulations. The relative difference to the least-square Monte Carlo price is reported in parenthesis. SPLS uses  $\tilde{N} = 1,000$  simulations, a sample size of  $N = 100$ , a nearest-neighbor size of  $m = 250$ , and an energy validation sample of size  $N' = 1,000$ . NNM uses  $N = 5,000$  simulations, a nearest-neighbor size of  $(p, q) = (5, 60)$ , and an energy validation sample of size  $N' = 10,000$ . The energy for all SPLS estimates is on average 0.666, and the energy for all NNM estimates is on average 0.007

Table 5 shows that the SPLS and NNM estimates are very accurate, despite the simple hyperparameters tuning procedure used. This shows that adapting SPLS and NNM to a particular option rests solely on finding appropriate hyperparameters. Further, hyperparameters tuning can be done in many ways allowing to meet any computational, accuracy or speed constraints.



## 6 An Asian Option

This section applies SPLS and NNM to a multiple exercise constrained window Asian option driven by a Levy diffusion. The multiple exercise right is subject to a local limit, a global limit, and a refraction period. An option of this complexity has never been priced in the literature.

Consider a multiple exercise Asian call option on the energy price  $S$  with a strike price of  $K$ . The option can be exercised  $T$  times per year, up to the maturity of one year. The average of the stock price is calculated over the last five periods. The option can be exercised at most  $n$  times, and between each exercise a minimum waiting time of  $R$  period is required. At each exercise, the option delivers at most  $q$  of the payoff, and the total amount delivered cannot exceed  $Q$ . With  $X$  the exercise decision, and  $Y$  the ordered quantity of energy, the stochastic program for the option can be written with

$$\max_{X, Y} E \left[ \sum_{t=4}^T e^{-rt} (A_t - K)_+ Y_t X_t \right] \quad (6.1)$$

$$\text{s.t.} \quad Y_t \in [0, q] \quad t = 0, 1, \dots, 50 \quad (6.1.1)$$

$$\sum_{t=0}^T Y_t < [0, Q] \quad (6.1.2)$$

$$\sum_{s=0}^R X_{t+s} \leq 1 \quad t = 0, 1, \dots, 50 \quad (6.1.3)$$

$$\sum_{t=0}^T X_t \leq n \quad (6.1.4)$$

$$X \in \{0, 1\} \quad (6.1.5)$$

where  $A_t = \frac{1}{5} \sum_{s=t-4}^t S_s$ , and  $r$  is the interest rate. The stochastic dynamic for the energy price follows the diffusion equation

$$dS = \theta(\mu - S)dt + \sigma dW + dJ, \quad (6.2)$$

where  $\theta$  is the reversion speed,  $\mu$  the long-term mean,  $\sigma$  the price volatility,  $W$  is a standard

Brownian motion, and  $J$  is an independent Poisson process with exponential jump. The Lévy measure of the compound Poisson process is  $\nu(dj) = \lambda\alpha e^{-\alpha j}$ .

The stochastic program of this option is non-linear as the objective contains a product of decision variables. However, the program can be linearized by defining a third decision variable, and adding constraints to force the third variable to be equal to the product of the variables. The procedure is as follows. Substitute every occurrence of the product  $yx$  by the variable  $z$ , and add the following three constraints

$$\begin{aligned} z &\leq y, \\ z &\leq ax, \\ z &\geq y + ax - a, \end{aligned}$$

where  $a$  is an upper bound on  $y$ ,  $x$  is binary and  $y$  is nonnegative. This and other *integer programming tricks* can be found in Bisschop (2016). Such tricks allow SPLS and NNM to handle complex option with a linear solver.

Table 6.1 displays the SPLS and NNM estimates for various pricing parameters. The estimates are obtained with a bootstrap procedure only: the hyperparameters of the algorithms are fixed, only multiple random starts are used, and no grid search is made. If the option price is taken as the midpoint between SPLS and NNM, the table shows that the estimates narrow the option price within 5% in average. As we used a simple hyperparameters tuning method, some of the estimates are less precise. The convergence theory of the algorithms and the previous examples guarantee that any less precise estimate can be made sharper with more computational effort.

$n$	$R$	$\sigma$	$\lambda = 0$		$\lambda = 0.1$			
			SPLS	NNM	SPLS	NNM		
1	0	5	1.09	(0.11)	1.21	1.14	(0.30)	1.47
		10	2.20	(0.21)	2.66	2.25	(0.24)	2.78
	2	5	1.10	(0.62)	1.78	1.13	(0.04)	1.17
		10	2.22	(0.18)	2.62	2.22	(0.55)	3.43
5	0	5	9.74	(0.06)	10.30	10.11	(0.11)	11.27
		10	19.48	(0.09)	21.17	19.72	(0.07)	21.07
	2	5	9.13	(0.14)	10.40	9.43	(0.03)	9.68
		10	18.21	(0.09)	19.85	18.59	(0.08)	20.14
10	0	5	19.60	(0.10)	21.65	20.39	(0.09)	22.19
		10	39.29	(0.06)	41.77	40.03	(0.02)	41.01
	2	5	16.67	(0.02)	17.04	17.24	(0.07)	18.48
		10	33.38	(0.04)	34.66	33.85	(0.03)	35.00
20	0	5	36.51	(0.05)	38.18	37.80	(0.28)	48.48
		10	72.99	(0.05)	76.89	74.23	(0.06)	78.49
	2	5	21.95	(0.07)	23.51	22.73	(0.11)	25.16
		10	43.89	(0.04)	45.64	44.54	(0.12)	50.01
25	0	5	43.63	(0.08)	47.00	45.23	(0.03)	46.41
		10	87.28	(0.10)	95.78	88.54	(0.03)	91.64
	2	5	21.95	(0.13)	24.90	22.73	(0.11)	25.13
		10	43.90	(0.05)	46.05	44.54	(0.10)	48.80

Table 6.1: SPLS and NNM estimates for a multiple exercise constrained window Asian option driven by a Levy diffusion. The option can be exercised 50 times per year, with a maturity of one year. The maximum number of exercise  $n$ , the refraction period  $R$ , the volatility  $\sigma$ , and the jump intensity  $\lambda$  is as indicated. The risk-free rate is 0.06, the reversion speed  $\theta$  is 0.02, and the jump severity is exponentially distributed with rate  $\alpha = 0.5$ . The strike price  $K$ , the initial stock price  $S_0$ , and the long-term mean  $\mu$  have the same value of 36. The global volume limit is  $Q = n - 0.5$ . The relative difference between the NNM and SPLS estimates is in parenthesis. SPLS uses  $\tilde{N} = 1,000$  simulations, a sample size in the projected lookahead operator of  $N = 100$ , a nearest-neighbor size of  $m = 50$ , and an energy validation sample of size  $N' = 1,000$ . NNM uses  $N = 1,500$  simulations, a nearest-neighbor size of  $(p, q) = (10, 60)$ , an energy validation sample of size  $N' = 10,000$ , and a lower bound barrier of 1.02. The SPLS estimate is the best estimate from 10 different random seeds, while the NNM estimate is the best estimate from 30 different random seeds. The filtration energy is 0 for SPLS, and 0.02 for NNM.

## 7 A Passport Option

This section applies SPLS and NNM to a passport option on three assets. The passport option is stylized so that it includes all the exotic rights presented in Penaud, Wilmott, and Ahn (1999), for a total of 8 rights. In Penaud, Wilmott, and Ahn (1999), these features were priced one at a time. The complexity of this example is beyond the available algorithm in the literature.

Consider a passport option on three stocks  $S^i, i = 1, 2, 3$ . The option can be exercised 12 times per year, up to the maturity of one year. At each time  $t$ , the investor can trade an amount  $Y_t^i \in [-Q, Q]$  in only one of the stock  $i$ . The number of switch from a position in one stock to a position in another stock is limited to  $n$ . The investor is allowed once to change the position limit to  $2Q$  for a duration of  $L$  periods. The change in position value is limited to  $|Y_t^i - Y_{t+1}^i| < q$ , so that change in position are smooth. The investor is allowed 3 exceptions to this smooth change constraint. Denote the trading account value by  $A_t$ . After 6 months, the investor needs to choose between a payoff at maturity of  $\max(A_T, 0)$  or  $\max(-A_T, 0)$ . If the trading account ever reaches the barrier value of  $B$ , the investor receives  $B$  at maturity and the option expires. The investor is allowed to reset the account value to zero 3 times. Finally, the investor is allowed once to mark the account value, and then to restore the account to the marked value at a later time.

To write the stochastic program for the option, let  $X$  be the exercise decision,  $Y$  the position amount,  $PL$  the position limit extension right,  $SL$  the smooth limit extension right,  $P_{long}$  the long or short decision,  $R$  the reset right,  $MP$  the magic potion mark right, and  $MPX$  the magic potion restore right. Also, for a binary decision  $Z$ , let  $\bar{Z} = 1 - Z$  be the complement. The stochastic program for the option can then be written with

$$\max_{X, Y, PL, SL, P_{long}} E \left[ e^{-rT} \left( \overline{P_B} P_{long} \max(A_T, 0) + \overline{P_B} \overline{P_{long}} \max(-A_T, 0) + P_B B \right) \right] \quad (7.1)$$

s.t.

Trading volume I

$$Y_t^i \in [-Q - QU_t, Q + QU_t] \quad i = 1, 2, 3 \quad t = 0, 1, \dots, T-1$$

$$U_t = \sum_{s=t-L}^t PL_s \quad t = 0, 1, \dots, T-1$$

$$\sum_{t=0}^{T-1} PL_t \leq 1$$

Trading volume II

$$|Y_t^i - Y_{t-1}^i| \leq q + 4QSL_t \quad i = 1, 2, 3 \quad t = 1, 2, \dots, T-1$$

$$\sum_{t=1}^{T-1} SL_t \leq 3$$

Trading timing

$$\sum_{i=1,2,3} X_t^i \leq 1 \quad t = 0, 1, \dots, T-1$$

$$\sum_{t=1}^{T-1} \sum_{i=1,2,3} |X_t^i - X_{t-1}^i| \leq n$$

Barrier

$$P_B = \bigvee_{t=1}^T [|A_t| \geq B]$$

Trading account

$$A_t = \overline{R_t} \overline{MPX_t} A_t^a + \overline{R_t} MPX_t A_t^b \quad t = 1, 2, \dots, T$$

$$A_t^a = A_{t-1} + \sum_{i=1,2,3} (S_t^i - S_{t-1}^i) Y_{t-1}^i X_{t-1}^i \quad t = 1, 2, \dots, T$$

$$A_t^b = \sum_{s=0}^{t-1} MP_s A_s \quad t = 1, 2, \dots, T$$

$$A_0 = 0$$

$$\sum_{t=0}^{T-1} R_t \leq 3 ; \quad \sum_{t=0}^{T-1} MP_t \leq 1 ; \quad \sum_{t=1}^T MPX_t \leq 1$$

Binary Decisions

$$X_t^i \in \{0, 1\} \quad i = 1, 2, 3 \quad t = 0, 1, \dots, T-1$$

$$PL_t, SL_t, MP_t, MPX_t \in \{0, 1\} \quad t = 0, 1, \dots, T-1$$

$$P_{long} \in \{0, 1\} \quad t = T/2$$

The risk-neutral dynamic for the stock price follows the stochastic differential equation

$$dS^i = rS^i dt + \sigma_i S^i dW^i, \quad (7.2)$$

where  $r$  is the risk-free-rate,  $\sigma_i$  is the volatility, and  $W^i$  is an independent standard Brownian motion.

Table 7.1 displays the SPLS and NMM estimates for various pricing parameters. The weighting stopping time used is the logical OR of every binary decision (1.6). This weighting stopping time at time  $t$  can be written with

$$\left( \bigvee_{i=1}^3 X_t^i \right) \bigvee PL_t \bigvee SL_t \bigvee MP_t \bigvee MPX_t \bigvee \llbracket Plong, t = T/2 \rrbracket.$$

The runtime of one SPLS estimate with a single processor is 10 hours, we use 100 processors to bring the runtime down to 5 minutes. For NNM, the runtime of one estimate with a single processor is 5 hours, we use 100 processors to bring the runtime down to 3 minutes. With 100 processors, the runtime of the entire table is 20 hours. To maintain the total runtime small, we opt for coarse estimates of the option price. The previous sections guarantee that with a greater computational effort very precise estimates of the option price can be obtained. For example, with 1,000 processors, the runtime of the entire table would be 2 hours, and fine-grain estimates can be obtained.

$n$	$Q$	$q$	$B = \infty$			$B = 20$				
			$L = 0$		$L = 2$	$L = 0$		$L = 2$		
1	1	1	4.05 (0.04)	4.21	4.70 (0.37)	6.44	3.44 (0.12)	3.85	3.23 (0.45)	4.70
		2	5.28 (0.21)	6.37	5.68 (0.86)	10.56	3.90 (0.08)	4.21	4.15 (0.07)	4.46
10	1	1	41.60 (0.03)	42.94	44.48 (0.06)	47.34	11.46 (0.22)	13.93	12.15 (0.56)	19.01
		2	44.60 (0.08)	48.34	44.76 (0.06)	47.63	11.70 (0.61)	18.84	10.97 (0.08)	11.89
5	1	1	4.46 (0.06)	4.72	4.11 (0.35)	5.56	3.78 (1.97)	11.22	3.77 (1.89)	10.88
		2	5.35 (0.78)	9.50	5.68 (0.02)	5.80	4.30 (0.25)	5.36	4.28 (0.42)	6.07
10	1	1	31.59 (0.43)	45.06	32.62 (0.61)	52.67	17.03 (1.34)	39.89	18.09 (0.53)	27.72
		2	36.55 (0.24)	45.27	36.56 (0.39)	50.91	16.51 (0.33)	21.88	17.67 (0.03)	18.19

Table 7.1: SPLS and NNM estimates for an exotic passport option. The option can be exercised 12 times per year, up to the maturity of one year. The number of switches from a position in one stock to a position in another stock is limited to  $n$ . The position limit is  $Q$ , and the maximal change in position is  $q$ . The position limit can be changed once to  $2Q$  for a duration of  $L$  periods. If the option reaches the barrier value of  $B$ , the investor receives  $B$  and the option expires. The SPLS estimate is on the left of the parenthesis, the NNM estimate is on the right of the parenthesis, and the relative difference between the NNM and SPLS estimates is in parenthesis. The volatility is  $(\sigma_1, \sigma_2, \sigma_3) = (0.05, 0.25, 0.5)$ , the initial stock price is  $S_0^1 = S_0^2 = S_0^3 = 36$ , and the risk-free-rate is 0.06. SPLS uses  $\tilde{N} = 1,000$  simulations, a sample size in the projected lookahead operator of  $N = 1$ , a nearest-neighbor size of  $m = 1$ , and an energy validation sample of size  $N' = 1,000$ . NNM uses  $N = 1,000$  simulations, a nearest-neighbor size of  $(p, q) = (5, 10)$ , an energy validation sample of size  $N' = 10,000$ , and a lower bound barrier of 1.02. Both the SPLS and NNM estimates are the best estimates from 5 different random seeds. The filtration energy is 0 for both SPLS and NNM.

## 8 Conclusion

This article presents an alternative to dynamic programming for the pricing of multiple exercise American option with constrained rights. The algorithms provide lower and upper estimates of the option price with convergence guarantees provided through a Vapnik-Chernovenkis dimension analysis. The algorithms are fast, fully general, and applicable with no adjustment to a large class of options. We illustrate the algorithm with two realistic examples including a swing option with four constraints, and a passport option with 16 constraints.

The ability to value constrained multiple exercise American derivatives has many important advantages. In particular, the exercise constraints are at the heart of the negotiations of such derivative contract. A general valuation approach greatly facilitates those negotiations. Further, such a valuation approach simplifies risk management and can broaden the hedging instruments offering for those derivatives. The presented algorithms provide such a general valuation approach.

## 9 Methods

This section collects the technical details behind the various results presented. The name of each subsection contains in parenthesis the equation number being proved. The last subsection details our implementation of SPLS and NNM. In this section, the Vapnik-Chernovenkis dimension is denoted by *VC-dimension*, and the shattering coefficient of a class  $\mathcal{A}$  with a sample of size  $n$  is denoted by  $s(\mathcal{A}, n)$ . The  $\ell_1$ -covering  $\mathcal{N}(\epsilon, A)$  of a set  $A$  is the cardinality of the smallest  $\epsilon$ -net necessary to cover the set with the norm being the  $\ell_1$ -norm divided by the dimension of the element in the set. In covering number estimates, we use a universal constant  $c$  to collect every constants, so that  $cx$  is equivalent to the big-O notation  $\mathcal{O}(x)$ .



## 9.1 Dual Problem Derivation (1.3)

Define the value function

$$V_t = \max_{X_t, Y_t} e^{-rt} f_t(Y_{0:t}, S_{0:t}) X_t + E(V_{t+1} | F_t),$$

where the maximum is taken over the admissible decisions  $X_t$  and  $Y_t$ , and  $V_{T+1}$  is assumed to be zero. Since the decision  $X_t$  is in  $\{0, 1\}$ , the value function is a supermartingale and admits the following Doob decomposition

$$V_t = V_0 + M_t - A_t,$$

where  $M_t$  is a martingale vanishing at time zero, and  $A_t$  is a previsible increasing process vanishing at time zero. Now, we have

$$\begin{aligned} V_0 &= \max_{X, Y} E \left[ \sum_{t=0}^T e^{-rt} f_t(Y_{0:t}, S_{0:t}) X_t \right] \\ &= \max_{X, Y} E \left[ \sum_{t=0}^T e^{-rt} f_t X_t - M_t X_t \right] \end{aligned} \tag{9.1}$$

$$\leq E \left[ \max_{x, y} \sum_{t=0}^T e^{-rt} f_t x_t - M_t x_t \right] \tag{9.2}$$

$$\leq E \left[ \max_{x, y} \sum_{t=0}^T V_t x_t - M_t x_t \right] \tag{9.3}$$

$$= E \left[ \max_{x, y} \sum_{t=0}^T (V_t - M_t) x_t \right] \tag{9.4}$$

$$= E \left[ \max_{x, y} \sum_{t=0}^T (V_0 - A_t) x_t \right] \tag{9.5}$$

$$= V_0 \tag{9.6}$$

(9.1) holds by the optional sampling theorem. (9.2) inserts the maximum into the expectation. (9.3) holds by definition of the value function and by noting that  $a < b$  implies  $ay < by$ , whenever  $y$  is in  $\{0, 1\}$ . (9.4) is a simple rewrite. (9.5) uses the Doob decomposition of the value function.

(9.6) holds by monotonicity of the compensator  $A$ , and assumes that the decision  $x_0$  at time zero is always admissible.

The dual formulation (1.3) follows by using (9.2) and observing that

$$V_0 \leq \min_M E \left[ \max_{x,y} \sum_{t=0}^T e^{-rt} f_t x_t - M_t x_t \right] \leq E \left[ \max_{x,y} \sum_{t=0}^T e^{-rt} f_t x_t - M_t x_t \right] = V_0,$$

where the last equality holds when  $M$  is taken as the martingale part of the value function.

## 9.2 SPLS Projected Strategy Convergence (2.15)

We have

$$\begin{aligned} & P \left( \sup_A \frac{1}{m_{s-t}} \sum_{i=1}^{m_{s-t}} \llbracket \bar{B}_i \in A \rrbracket > \epsilon \right) \\ &= E \left[ P \left( \sup_A \frac{1}{m_{s-t}} \sum_{i=1}^{m_{s-t}} \llbracket \bar{B}_i \in A \rrbracket > \epsilon \left| \{\bar{B}_i\}_{i=1}^{m_{s-t}} \right. \right) \right] \end{aligned} \quad (9.7)$$

$$\leq s(\mathcal{A}, m_{s-t}) E \left[ \sup_A P \left( \frac{1}{m_{s-t}} \sum_{i=1}^{m_{s-t}} \llbracket \bar{B}_i \in A \rrbracket > \epsilon \left| \{\bar{B}_i\}_{i=1}^{m_{s-t}} \right. \right) \right] \quad (9.8)$$

$$\leq 2s(\mathcal{A}, m_{s-t}) e^{-2\epsilon^2 m_{s-t}}. \quad (9.9)$$

(9.7) is a simple conditioning on the Voronoi cell  $\{\bar{B}_i\}_{i=1}^{m_{s-t}}$ , (9.8) holds by definition of the shattering coefficient  $s(\mathcal{A}, m_{s-t})$ , and (9.9) holds by Hoeffding's inequality. To obtain the shatter coefficient, note that

$$\begin{aligned} s(\mathcal{A}, m_{s-t}) &\leq \prod_{i=1}^{\alpha_{s-t}} s(2^{B_i}, m_{s-t}) \\ &\leq \prod_{i=1}^{\alpha_{s-t}} m_{s-t}^{(d(s-t)+1)(\alpha_{s-t}-1)} \\ &\leq m_{s-t}^{3d(s-t)\alpha_{s-t}^2} \end{aligned}$$

as each set  $2^{B_i}$  is the intersection of at most  $\alpha_{s-t} - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d(s-t) + 1$ . The factor of 3 is used to bound the VC-dimension of each hyperplane so that the bound is greater than 2. See Theorem 13.5, 13.9, 13.8 and 13.3 in Devroye, Györfi, and Lugosi (1996). The result follows.

### 9.3 SPLS Projected Lookahead Operator Convergence (2.16)

The shatter coefficient of the first component in  $A$  can be bounded with

$$\begin{aligned} s\left(\prod_{s=t}^T \{\bar{B}^{s,i}\}_{i=1}^{m_{s-t}}, n\right) &\leq \prod_{s=t}^T \prod_{i=1}^{m_{s-t}} s(\bar{B}^{s,i}, n) \\ &\leq \prod_{s=t}^T \prod_{i=1}^{m_{s-t}} n^{(d(s-t)+1)(m_{s-t}-1)} \\ &\leq n^{d(T-t) \sum_{s=t}^T m_{s-t}^2} \end{aligned}$$

as each set  $\bar{B}^{s,i}$  is the intersection of at most  $m_{s-t} - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d(s-t) + 1$ . See Theorem 13.5, 13.9 and 13.8 in Devroye, Györfi, and Lugosi (1996). By Theorem 13.8 in Devroye, Györfi, and Lugosi (1996), the shatter coefficient for the second component in  $A$  is  $N^{2d(T-t)}$ , which we bound by  $N^{3d(T-t)}$  to make the VC-dimension greater than 2. The result then follows by Theorem 13.5, 13.3 and 12.5 in Devroye, Györfi, and Lugosi (1996).

### 9.4 SPLS Projected Option Value (2.17)

In this proof, the projected lookahead operator  $\bar{L}(\bar{X}_{t:T}, F_t)$  at time  $t$  is assumed to be a uniform Lipschitz function of the filtration, so that there exists a constant  $C$  for which the operator is Lipschitz in the filtration for any strategy  $\bar{X}_{t:T}$ . This assumption implies that the operator can be viewed as strategy, with the internal structure of the operator abstracted. The proof hence proceed by considering the covering number induced by any strategy that is a Lipschitz function of the filtration.

Denote by  $J(X, S^{(\tilde{N})})$  the vector  $(J(X, S^{\tilde{n}}), \tilde{n} = 1, 2, \dots, \tilde{N})$ , we bound the  $\ell_1$ -covering  $\mathcal{N}(\epsilon, J(X, S^{(\tilde{N})}))$

with an  $\ell_1$ -covering of the strategy  $\mathcal{N}(\epsilon, X)$ . Let  $X$  and  $Y$  be two strategies, by the Lipschitz assumption,

$$\begin{aligned} \|J(X, S^{(\tilde{N})}) - J(Y, S^{(\tilde{N})})\| &= \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \|J(X, S^{\tilde{n}}) - J(Y, S^{\tilde{n}})\| \\ &\leq \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} K \|X - Y\| \\ &= K \|X - Y\|. \end{aligned}$$

Proving that  $\mathcal{N}(\epsilon, J(X, S^{(\tilde{N})})) \leq \mathcal{N}(\frac{\epsilon}{K}, X)$ .

With  $F_t$  and  $F'_t$  two different filtrations, by the Lipschitz assumption,

$$\begin{aligned} \|X(F) - X(F')\| &= \frac{1}{T} \sum_{t=0}^T \|X_t(F_t) - X_t(F'_t)\| \\ &\leq \frac{C}{T} \sum_{t=0}^T \|F_t - F'_t\|, \end{aligned}$$

This proves that

$$\mathcal{N}(\frac{\epsilon}{K}, X) \leq \prod_{t=0}^T \mathcal{N}(\frac{\epsilon}{CK}, F_t),$$

The minimum information necessary in the filtration are the past decision  $X_{0:t-1}$  and the stock path history  $S_{0:t}$ . As each decision is a vector in  $\mathbb{R}^a$ , and each stock is a vector in  $\mathbb{R}^d$ , the filtration is at most a vector in  $\mathbb{R}^{(a+d)t}$ , giving that

$$\begin{aligned} \mathcal{N}(\frac{\epsilon}{K}, X) &\leq \prod_{t=0}^T \left(\frac{CK}{\epsilon}\right)^{(a+d)t} \\ &\leq \left(\frac{CK}{\epsilon}\right)^{(a+d)T^2}. \end{aligned}$$

The result then follows by Theorem 29.1 in Devroye, Györfi, and Lugosi (1996).

## 9.5 SPLS Consistency (2.18)

Consider the lookahead operator  $L(X_{t:T}, F_t)$  at time  $t$ , and define the estimation error for a projected strategy with

$$Z_t = \left| \frac{1}{N} \sum_{n=1}^N J(\bar{L}(\bar{X}_{t:T}, S^n) - E[J(L(X_{t:T}, F_t))] \right|,$$

where  $\bar{X}$  is the projection of  $X$ . Denote by  $X \sim \bar{X}$ , the event that the projected strategy has unambiguous cells. By using (2.15), this event can be written with

$$(X \sim \bar{X}) = \left( \sup_A \frac{1}{m_{s-t}} \sum_{i=1}^{m_{s-t}} [\bar{B}_i \in A] < \epsilon, t = 0, 1, \dots, T \right).$$

By conditioning, the probability of error becomes

$$P(Z_t > \epsilon) = P(Z_t > \epsilon | X \sim \bar{X}) P(X \sim \bar{X}) + P(Z_t > \epsilon | X \not\sim \bar{X}) P(X \not\sim \bar{X}).$$

By (2.15), the second term tends to zero when  $m$  is large. For the first term, the conditioning event is assumed to imply that any strategy  $X$  can be accurately projected to a strategy  $\bar{X}$ . The magnitude of the estimation error  $Z_t$  is then due solely to a discrepancy in distribution. By using (2.16), this discrepancy can be controlled with the expected estimation error

$$E[Z_t] \leq BE \left[ \sup_A |\nu_N(A) - \nu(A)| \right],$$

so that the estimation error  $Z_t$  converges in mean to zero. Markov inequality then implies that the first term tends to zero with  $N$  and with  $m_{\max}^2 < o(N)$ .

To complete the proof, let  $Z < \epsilon$  be the event that all the estimation error  $Z_t$  are less than  $\epsilon$ . The probability of an error in option value can then be written with

$$P(|\bar{V} - V| > \epsilon) = P(|\bar{V} - V| > \epsilon | Z < \epsilon) P(Z < \epsilon) + P(|\bar{V} - V| > \epsilon | Z > \epsilon) P(Z > \epsilon).$$

By the previous, the second term tends to zero. For the first term, the conditioning event is assumed to imply that the lookahead operator  $L(X_{t:T}, F_t)$  can be approximated arbitrarily well by a projected lookahead operator  $\bar{L}(\bar{X}_{t:T}, F_t)$ . When this is the case, the option value  $V$  can be approximated arbitrarily well by the expected projected residual payoff  $E[J(\bar{L}, S)]$ , so that  $V = E[J(\bar{L}, S)]$  in probability, for some projected lookahead operator  $\bar{L}$ . By (2.17), the first term then tends to zero with  $\tilde{N}$ .

## 9.6 SPLS Convergence in Bayes-Value (2.19)

Denote by  $J(\bar{L}_{0:T}, S^{(\tilde{N})})$  the vector  $(J(\bar{L}_{0:T}, S^{\tilde{n}}), \tilde{n} = 1, 2, \dots, \tilde{N})$ . The following four steps provide an estimate for the expected covering number  $E(\mathcal{N}(\epsilon, J(\bar{L}_{0:T}, S^{(\tilde{N})})))$ . The estimate is crude and can be used to bound  $E(\mathcal{N}(\frac{\epsilon}{8}, J(\bar{L}_{0:T}, S^{(\tilde{N})})))$ . Using this estimate, the result follows by Theorem 29.1 in Devroye, Györfi, and Lugosi (1996).

**Step 1.** We bound the covering number of the residual payoff  $\mathcal{N}(\epsilon, J(\bar{L}_{0:T}, S^{(\tilde{N})}))$  with the covering number of the projected lookahead process  $\mathcal{N}(\epsilon, \bar{L}_{0:T})$ . Let  $\bar{L}$  and  $\bar{L}'$  be two projected lookahead processes, by the Lipschitz assumption,

$$\begin{aligned} \|J(\bar{L}_{0:T}, S^{(\tilde{N})}) - J(\bar{L}'_{0:T}, S^{(\tilde{N})})\| &= \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \|J(\bar{L}_{0:T}, S^{\tilde{n}}) - J(\bar{L}'_{0:T}, S^{\tilde{n}})\| \\ &\leq \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} K \|\bar{L}_{0:T} - \bar{L}'_{0:T}\| \\ &= K \|\bar{L}_{0:T} - \bar{L}'_{0:T}\|. \end{aligned}$$

Proving that

$$\mathcal{N}(\epsilon, J(X, S^{(\tilde{N})})) \leq \mathcal{N}\left(\frac{\epsilon}{K}, \bar{L}_{0:T}\right).$$

**Step 2.** We bound the covering number of the projected lookahead process with the covering number of the projected lookahead operator. With  $\bar{L}$  and  $\bar{L}'$  two projected lookahead processes, we

have

$$\|\bar{L}_{0:t} - \bar{L}'_{0:t}\| \leq \sum_{t=0}^T \|\bar{L}_t - \bar{L}'_t\|.$$

Proving that

$$\mathcal{N}(\epsilon, \bar{L}_{0:T}) \leq \prod_{t=0}^T \mathcal{N}(\epsilon, \bar{L}_t).$$

**Step 3.** From (2.9), the projected lookahead operator  $\bar{L}_t$  can be written with

$$\bar{L}(\bar{X}_{t:T}, F_t) = \left( \operatorname{argmax}_{\bar{X}_{t:T}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n) \right)_t.$$

The minimum information necessary in the filtration are the past decision  $X_{0:t-1}$  and the stock path history  $S_{0:t}$ . The projected lookahead operator is hence a function of the form

$$(\bar{X}_{t:T}, X_{0:t-1}, S_{0:t}) \rightarrow \left( \operatorname{argmax}_{\bar{X}_{t:T}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n) \right)_t.$$

By viewing a function as a subset of the product of the function domain and codomain, we have

$$\mathcal{N}(\epsilon, \bar{L}_t) \leq \mathcal{N}\left(\frac{\epsilon}{2}, \bar{X}_{t:T}\right) \mathcal{N}\left(\frac{\epsilon}{2}, X_{0:t-1}\right) \mathcal{N}\left(\frac{\epsilon}{2}, S_{0:t}\right) \mathcal{N}\left(\frac{\epsilon}{2}, Z_t\right),$$

where  $Z_t$  is the codomain of the projected lookahead operator. The next three sub-steps derive the covering numbers for each element in the right-hand side above.

**Step 3.a,  $\bar{X}_{t:T}$ .** The covering number for a decision  $X_s$  in  $X_{t:T}$  is the covering number for a step-function constructed with a nearest-neighbor basis for the stock path (2.8). Such a function is of the form  $\cup_{i=1}^{m_{s-t}} B_i \times x^i$ , where  $B_i$  is a Voronoi cell in  $\mathbb{R}^{d(s-t)}$ , and  $x^i$  is in  $\mathbb{R}^a$ . The covering number of the decision can hence be bounded by the product of the covering numbers, giving

$$\mathcal{N}(\epsilon, X_s) \leq \left( \frac{X_{\max}}{\epsilon} \right)^a \mathcal{N}\left(\frac{\epsilon}{2}, \cup_{i=1}^{m_{s-t}} B_i\right),$$

where  $X_{\max}$  is an upper bound on a decision.

We bound the covering number of the Voronoi tessellation  $\cup_{i=1}^{m_{s-t}} B_i$  in term of its VC-dimension. Each cell  $B_i$  in the tessellation is the intersection of at most  $m_{s-t} - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d(s-t) + 1$ . Using Theorem 1.1 in Van Der Vaart and Wellner (2009), the VC-dimension of a cell is bounded by  $3d(s-t)m_{s-t} \log(4m_{s-t})$ . Another application of this theorem gives that the VC-dimension of the tessellation is bounded by  $9d(s-t)m_{s-t}^2 \log(4m_{s-t}) \log(4m_{s-t})$ , or simply  $cd(s-t)m_{s-t}^3$ . Using Theorem 2.6.4 in Van Der Vaart and Wellner (1996), the covering number for the Voronoi tessellation is then given by

$$\begin{aligned} \mathcal{N}(\epsilon, \cup_{i=1}^{m_{s-t}} B_i) &\leq cd(s-t)m_{s-t}^3 \left(\frac{4e}{\epsilon}\right)^{2cd(s-t)m_{s-t}^3} \\ &\leq cdTm_{s-t}^3 \left(\frac{c}{\epsilon}\right)^{cdTm_{s-t}^3}. \end{aligned} \quad (9.10)$$

We then have

$$\begin{aligned} \mathcal{N}(\epsilon, \bar{X}_{t:T}) &\leq \prod_{s=t}^T \mathcal{N}(\epsilon, X_s) \\ &\leq \prod_{s=t}^T \left(\frac{X_{\max}}{\epsilon}\right)^a \mathcal{N}\left(\frac{\epsilon}{2}, \cup_{i=1}^{m_{s-t}} B_i\right) \\ &\leq \left(\frac{X_{\max}}{\epsilon}\right)^{aT} \prod_{s=t}^T cdTm_{s-t}^3 \left(\frac{c}{\epsilon}\right)^{cdTm_{s-t}^3} \\ &\leq (cdTm_{\max}^3)^T \left(\frac{X_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3}, \end{aligned}$$

where  $m_{\max} = \|m\|_{\infty}$  is the largest nearest-neighbor size.

**Step 3.b,**  $X_{0:t-1}$  and  $S_{0:t}$ . The covering number for the past decision  $X_{0:t-1}$  is the covering number for a vector in  $\mathbb{R}^{a(t-1)}$  and can be written with

$$\mathcal{N}(\epsilon, X_{0:t-1}) = \left(\frac{X_{\max}}{\epsilon}\right)^{a(t-1)} \leq \left(\frac{X_{\max}}{\epsilon}\right)^{aT}.$$



Similarly, the observed stock path  $S_{0:t}$  is a vector in  $\mathbb{R}^{dt}$  with covering number

$$\mathcal{N}(\epsilon, S_{0:t}) = \left( \frac{S_{\max}}{\epsilon} \right)^{at} \leq \left( \frac{S_{\max}}{\epsilon} \right)^{aT},$$

where  $S_{\max}$  is an upper bound on the stock price.

**Step 3.c**,  $Z_t$ . Since the selection operator  $(\cdot)_t$  is an injection, and  $\operatorname{argmax}$  is an injection with an appropriate tie-breaking rule, we can bound the covering number of the codomain  $Z_t$  with a covering number for the best empirical average residual payoff. This best empirical average can be written with

$$A_N^* = \max_{\bar{X}_{t:T}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n).$$

Denote by  $A_{N,t}$  the empirical average

$$A_{N,t} = \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n),$$

and denote by  $A_t$  the expected average

$$A_t = E[J(\bar{X}_{t:T}, S_{t:T}^n) | F_t].$$

We bound the covering number of the best empirical average with a uniform deviation from the expected average (Devroye, Györfi, and Lugosi 1996, Lemma 8.2). Let  $A_{N,t}$  and  $A_{N,t}^*$  be two best averages computed from different samples, then

$$\begin{aligned} |A_{N,t}^* - A_{N,t}^*| &= \left| \max_{\bar{X}_{t:T}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n) - \max_{\bar{X}_{t:T}} \frac{1}{N} \sum_{n=0}^N J(\bar{X}_{t:T}, S_{t:T}^n) \right| \\ &= |A_{N,t}^* - A_t + A_t - A_{N,t}^*| \\ &\leq 2 \max_{\bar{X}_{t:T}, X_{0:t-1}} |A_{N,t} - A_t|. \end{aligned}$$

Using the uniform law of large numbers (Devroye, Györfi, and Lugosi 1996, Theorem 29.1), we can

now estimate the probability that two best averages are closed by  $\epsilon$

$$\begin{aligned}
P(|A_{N,t}^* - A'_{N,t}| \leq \epsilon) &\leq P\left(\max_{\bar{X}_{t:T}, X_{0:t-1}} |A_{N,t} - A_t| \leq \frac{\epsilon}{2}\right) \\
&= 1 - P\left(\max_{\bar{X}_{t:T}, X_{0:t-1}} |A_{N,t} - A_t| > \frac{\epsilon}{2}\right) \\
&\geq 1 - 8E\left(\mathcal{N}\left(\frac{\epsilon}{16}, (\bar{X}_{t:T}, X_{0:t-1})\right)\right)e^{-\epsilon^2 N/(512B^2)},
\end{aligned}$$

where  $B$  is an upper bound on the residual payoff. By viewing the expected covering number of the best empirical average as a geometric distribution, we have

$$\begin{aligned}
E(\mathcal{N}(\epsilon, A_{N,t}^*)) &= \left(P(|A_{N,t}^* - A'_{N,t}| \leq \epsilon)\right)^{-1} \\
&\leq \left(1 - 8E\left(\mathcal{N}\left(\frac{\epsilon}{16}, (\bar{X}_{t:T}, X_{0:t-1})\right)\right)e^{-\epsilon^2 N/(512B^2)}\right)^{-1} \\
&= \exp\left(cE\left(\mathcal{N}\left(\frac{\epsilon}{16}, (\bar{X}_{t:T}, X_{0:t-1})\right)\right)e^{-\epsilon^2 N/(512B^2)}\right),
\end{aligned}$$

where the identity  $1 - x \geq e^{-cx}$  was used.

By using Step 3.a and 3.b, the product bound gives

$$\begin{aligned}
\mathcal{N}\left(\frac{\epsilon}{16}, (\bar{X}_{t:T}, X_{0:t-1})\right) &\leq \mathcal{N}\left(\frac{\epsilon}{16}, \bar{X}_{t:T}\right)\mathcal{N}\left(\frac{\epsilon}{16}, X_{0:t-1}\right) \\
&\leq \left(\frac{16X_{\max}}{\epsilon}\right)^{aT} (cdTm_{\max}^3)^T \left(\frac{X_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3} \\
&\leq (cdTm_{\max}^3)^T \left(\frac{X_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3}
\end{aligned}$$

proving that

$$E(\mathcal{N}(\epsilon, A_{N,t}^*)) \leq \exp\left((cdTm_{\max}^3)^T \left(\frac{X_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3} e^{-\epsilon^2 N/(512B^2)}\right).$$

**Step 4.** By combining all the previous results, we have

$$\begin{aligned}
& E(\mathcal{N}(\epsilon, J(X, S^{(\tilde{N})}))) \\
& \leq E(\mathcal{N}(\frac{\epsilon}{K}, \bar{L}_{0:T})) \\
& \leq E\left(\prod_{t=0}^T \mathcal{N}(\frac{\epsilon}{K}, \bar{L}_t)\right) \\
& \leq E\left(\prod_{t=0}^T \mathcal{N}(\frac{\epsilon}{2K}, \bar{X}_{t:T}) \mathcal{N}(\frac{\epsilon}{2K}, X_{0:t-1}) \mathcal{N}(\frac{\epsilon}{2K}, S_{0:t}) \mathcal{N}(\frac{\epsilon}{2K}, Z_t)\right) \\
& \leq (cdTm_{\max}^3)^{T^2} \left(\frac{KX_{\max}}{\epsilon}\right)^{c(a+d)T^3m_{\max}^3} \left(\frac{KX_{\max}}{\epsilon}\right)^{aT^2} \\
& \left(\frac{KS_{\max}}{\epsilon}\right)^{aT^2} \exp\left(T(cdTm_{\max}^3)^T \left(\frac{KX_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3} e^{-\epsilon^2N/(512B^2K^2)}\right) \\
& \leq \exp\left(T(cdTm_{\max}^3)^T \left(\frac{KX_{\max}S_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3} e^{-\epsilon^2N/(512B^2K^2)}\right).
\end{aligned}$$

## 9.7 SPLS Universal Bayes-Value Consistency and Consistency (2.20)

To prove the universal Bayes-value efficiency, assume that the lookahead operator is restricted to the class of strategy implied by the projected lookahead operator. When this is the case, we can write

$$V = \sup_{\bar{L}} E(J(\bar{L}_{0:T}, S)).$$

Use Lemma 8.2 in Devroye, Györfi, and Lugosi (1996) to write

$$\begin{aligned}
\bar{V} - V & = \left(\sup_{\bar{L}} \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} J(\bar{L}_{0:T}, S^{\tilde{n}})\right) - \left(\sup_{\bar{L}} E(J(\bar{L}_{0:T}, S))\right) \\
& \leq 2 \sup_{\bar{L}} \left|\frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} J(\bar{L}_{0:T}, S^{\tilde{n}}) - E(J(\bar{L}_{0:T}, S))\right|.
\end{aligned}$$

By using (2.19), and Problem 12.1 in Devroye, Györfi, and Lugosi (1996), the expectation of the above right-hand side is bounded by

$$\sqrt{\frac{T(cdTm_{\max}^3)^T \left(\frac{KX_{\max}S_{\max}}{\epsilon}\right)^{c(a+d)T^2m_{\max}^3} e^{-\epsilon^2N/(512B^2K^2)}}{\tilde{N}/(128B^2)}}.$$

This bound implies that  $E(|\bar{V} - V|)$  converges to zero with  $\tilde{N}$ ,  $N$  and  $m$  going to infinity, and  $m_{\max}^3 < o(N)$ , so that the projected option value converges in mean to the option value. As convergence in mean implies convergence in  $\ell_1$ -norm, the universal consistency result (2.20) follows.

To prove consistency of SPLS, note that convergence in mean implies convergence in probability, and the consistency result follows.

## 9.8 NNM Projected Martingale in a Binomial World (Section 3.2.1)

For a vanilla American put in a binomial world, the optimal martingale has size  $(\alpha, \beta) = ((2^{t-1}, 2), t = 1, 2, \dots, T)$ , and a good projected martingale can be obtained with  $(p, q) = ((\mathcal{O}(t), 2), t = 1, 2, \dots, T)$ . The stated size for the optimal martingale is by construction of the binomial world. At time  $t$ , the maximum number of different stock paths leading to time  $t$  is  $2^{t-1}$ , and, given the path history up to time  $t - 1$ , the maximum number of different stock prices at time  $t$  is 2. The claimed size for the conditioning part of a good projected martingale can be verified numerically. To do so quantize with Lloyd's method all the  $2^{t-1}$  path leading to time  $t$ . Repeat this quantization over time, and note that a quantization with a small mean square error is obtained when the number of centroids grow linearly over time.

## 9.9 NNM Projected Martingale Convergence (3.7, 3.8)

For the conditioning part,

$$\begin{aligned} & P\left(\sup_A \frac{1}{p_t} \sum_{i=1}^{p_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon\right) \\ &= E\left[P\left(\sup_A \frac{1}{p_t} \sum_{i=1}^{p_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon \middle| \{\bar{B}_i\}_{i=1}^{p_t}\right)\right] \end{aligned} \quad (9.11)$$

$$\leq s(\mathcal{A}, p_t) E\left[\sup_A P\left(\frac{1}{p_t} \sum_{i=1}^{p_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon \middle| \{\bar{B}_i\}_{i=1}^{p_t}\right)\right] \quad (9.12)$$

$$\leq 2s(\mathcal{A}, p_t) e^{-2\epsilon^2 p_t} \quad (9.13)$$

(9.11) is a simple conditioning on the Voronoi cell  $\{\bar{B}_i\}_{i=1}^{p_t}$ , (9.12) holds by definition of the shattering coefficient  $s(\mathcal{A}, p_t)$ , and (9.13) holds by Hoeffding's inequality. To obtain the shatter coefficient, note that

$$\begin{aligned} s(\mathcal{A}, p_t) &\leq \prod_{i=1}^{\alpha_t} s(2^{B_i}, p_t) \\ &\leq \prod_{i=1}^{\alpha_t} p_t^{(dt+1)(\alpha_t-1)} \\ &\leq p_t^{3dt\alpha_t^2} \end{aligned}$$

as each set  $B_i$  is the intersection of at most  $\alpha_t - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $dt + 1$ . The factor of 3 is used to bound the VC-dimension of each hyperplane so that the bound is greater than 2. See Theorem 13.5, 13.9, 13.8 and 13.3 in Devroye, Györfi, and Lugosi (1996).

Similarly, for the current part,

$$\begin{aligned}
& P\left(\sup_A \frac{1}{q_t} \sum_{i=1}^{q_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon\right) \\
&= E\left[P\left(\sup_A \frac{1}{q_t} \sum_{i=1}^{q_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon \middle| \{\bar{B}_i\}_{i=1}^{q_t}\right)\right]
\end{aligned} \tag{9.14}$$

$$\leq s(\mathcal{A}, q_t) E\left[\sup_A P\left(\frac{1}{q_t} \sum_{i=1}^{q_t} \mathbb{I}[\bar{B}_i \in A] > \epsilon \middle| \{\bar{B}_i\}_{i=1}^{q_t}\right)\right] \tag{9.15}$$

$$\leq 2s(\mathcal{A}, q_t) e^{-2\epsilon^2 q_t}. \tag{9.16}$$

To obtain the shatter coefficient, note that

$$\begin{aligned}
s(\mathcal{A}, q_t) &\leq \prod_{i=1}^{\beta_t} s(2^{B_i}, q_t) \\
&\leq \prod_{i=1}^{\beta_t} q_t^{(d+1)(\beta_t-1)} \\
&\leq q_t^{3d\beta_t^2}
\end{aligned}$$

as each set  $B_i$  is the intersection of at most  $\beta_t - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d + 1$ . The factor of 3 is used to bound the VC-dimension of each hyperplane so that the bound is greater than 2. See Theorem 13.5, 13.9, 13.8 and 13.3 in Devroye, Györfi, and Lugosi (1996).

## 9.10 NNM Relaxed Rogers Operator (3.9, 3.10)

For the stochastic process estimate, the shatter coefficient of the class can be bounded with

$$\begin{aligned}
s(\{B \times [a, b]\}, N) &\leq \left(\prod_{t=1}^T \prod_{i=1}^{p_t q_t} s(\bar{B}_i^t, N)\right) s([a, b], N) \\
&\leq N^{2T} \prod_{t=1}^T \prod_{i=1}^{p_t q_t} N^{(d(t-1)+1)(p_t-1)+(d+1)(q_t-1)} \\
&\leq N^{3dT} \sum_{i=1}^T t p_i^2 q_i^2
\end{aligned}$$

Indeed, the VC-dimension of an interval in  $\mathbb{R}^T$  is  $2T$ . Each set  $\bar{B}_i^t$  can be written as  $U \times W$ . The conditioning part  $U$  is the intersection of at most  $p_t - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d(t - 1) + 1$ . The current part  $W$  is the intersection of at most  $q_t - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $d + 1$ . The factor of 3 is used to obtain a simple final bound that is greater than 2. See Theorem 13.5, 13.9, 13.8 and 13.3 in Devroye, Györfi, and Lugosi (1996).

For the point estimates, the same steps as above give the following shatter coefficient

$$s(\{B \times [a, b]\}, N) \leq N^{3d \sum_{t=1}^T tp_t^2 q_t^2}$$

as the interval  $[a, b]$  is now an interval in  $\mathbb{R}^d$  with a VC-dimension of  $2d$ .

### 9.11 NNM Projected Dual Value (3.11)

Denote by  $D(\bar{M}, S^{(\tilde{N})})$  the vector  $(D(\bar{M}, S^{\tilde{n}}, \tilde{n} = 1, 2, \dots, \tilde{N}))$ , and let  $\bar{M}$  and  $\bar{M}'$  be two projected martingale processes. By the Lipschitz assumption

$$\begin{aligned} \|D(\bar{M}, S^{(\tilde{N})}) - D(\bar{M}', S^{(\tilde{N})})\| &= \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \|D(\bar{M}, S^{\tilde{n}}) - D(\bar{M}', S^{\tilde{n}})\| \\ &\leq \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} K \|\bar{M} - \bar{M}'\| \\ &= K \|\bar{M} - \bar{M}'\|. \end{aligned}$$

Proving that  $\mathcal{N}(\epsilon, D(\bar{M}, S^{(\tilde{N})})) \leq \mathcal{N}(\frac{\epsilon}{K}, \bar{M})$ .

We bound the covering number of the projected martingale process with the covering number of the projected martingale. Indeed, with  $\bar{M}$  and  $\bar{M}'$  two projected martingales

$$\|\bar{M}_{0:T} - \bar{M}'_{0:T}\| = \frac{1}{T} \sum_{t=0}^T \|\bar{M}_t - \bar{M}'_t\|.$$

Proving that  $\mathcal{N}(\epsilon, \bar{M}_{0:T}) \leq \prod_{t=0}^T \mathcal{N}(\epsilon, \bar{M}_t)$ .

Each projected martingale  $M_t$  is a step-function on a nearest-neighbor basis for the stock path and can be written  $\cup_{i=1}^{p_t q_t} B_i \times m^i$  where  $m^i$  is bounded by  $U$ , and  $B_i$  is the intersection of two Voronoi cells, one in  $\mathbb{R}^{d(t-1)}$  and one in  $\mathbb{R}^d$ . The covering number of the decision can hence be bounded by the product of the covering numbers, giving

$$\mathcal{N}(\epsilon, \bar{M}_t) \leq \left(\frac{2U}{\epsilon}\right) \mathcal{N}\left(\frac{\epsilon}{2}, \cup_{i=1}^{p_t q_t} B_i\right).$$

We bound the covering number of the Voronoi tessellation  $\cup_{i=1}^{p_t q_t} B_i$  in term of its VC-dimension. Each cell  $B_i$  in the tessellation is the intersection of at most  $p_t q_t - 1$  hyperplanes, and the VC-dimension of each hyperplane is  $dt + 1$ . Using Theorem 1.1 in Van Der Vaart and Wellner (2009), the VC-dimension of each cell is bounded by  $3dtp_t q_t \log(4p_t q_t)$ . Another application of this theorem gives that the VC-dimension of the tessellation is bounded by  $9dtp_t^2 q_t^2 \log(4p_t q_t) \log(4p_t q_t)$ , or simply  $cdtp_t^3 q_t^3$ . By Theorem 2.6.4 in Van Der Vaart and Wellner (1996), the covering number for the Voronoi tessellation is then given by

$$\begin{aligned} \mathcal{N}(\epsilon, \cup_{i=1}^{p_t q_t} B_i) &\leq cdt p_t^3 q_t^3 \left(\frac{4e}{\epsilon}\right)^{cdtp_t^3 q_t^3} \\ &\leq cdT p_t^3 q_t^3 \left(\frac{c}{\epsilon}\right)^{cdT p_t^3 q_t^3}. \end{aligned}$$

Proving that

$$\begin{aligned} \mathcal{N}(\epsilon, \bar{M}_t) &\leq \left(\frac{2U}{\epsilon}\right) \mathcal{N}\left(\frac{\epsilon}{2}, \cup_{i=1}^{p_t q_t} B_i\right) \\ &\leq \left(\frac{2U}{\epsilon}\right) cdT p_t^3 q_t^3 \left(\frac{c}{\epsilon}\right)^{cdT p_t^3 q_t^3} \\ &\leq cdT p_t^3 q_t^3 \left(\frac{cU}{\epsilon}\right)^{cdT p_t^3 q_t^3}. \end{aligned}$$



By combining all the previous results, we have

$$\begin{aligned}
\mathcal{N}(\epsilon, D(\bar{M}, S^{(N)})) &\leq \mathcal{N}\left(\frac{\epsilon}{K}, \bar{M}\right) \\
&\leq \prod_{t=0}^T \mathcal{N}\left(\frac{\epsilon}{K}, \bar{M}_t\right) \\
&\leq \prod_{t=0}^T cdT p_t^3 q_t^3 \left(\frac{cUK}{\epsilon}\right)^{cdT p_t^3 q_t^3} \\
&\leq (cdT p_{\max}^3 q_{\max}^3)^T \left(\frac{UK}{\epsilon}\right)^{cdT^2 p_{\max}^3 q_{\max}^3}, \tag{9.17}
\end{aligned}$$

where  $(p_{\max}, q_{\max}) = (\|p\|_{\infty}, \|q\|_{\infty})$  is the largest nearest-neighbor size. Our estimate for the covering number  $\mathcal{N}(\epsilon, D(\bar{M}, S^{(N)}))$  is crude and can be used for  $\mathcal{N}(\frac{\epsilon}{8}, D(\bar{M}, S^{(N)}))$ . The result now follows by Theorem 29.1 in Devroye, Györfi, and Lugosi (1996).

## 9.12 NNM Consistency (3.12, 3.13)

We start with the consistency of the relaxed dual value (3.13). Let  $M$  be the optimal martingale and consider the estimation error in the relaxed dual value

$$|\bar{V} - V| = \left| \frac{1}{N} \sum_{n=1}^N D(\bar{R}^m(\bar{Z}), S^n) - E[D(M, S)] \right|.$$

Denote by  $U$  the event that the projected martingale has unambiguous cells. By using (3.7) and (3.8), this event can be written with

$$U = \left( \sup_A \frac{1}{pt} \sum_{i=1}^{pt} [\bar{B}_i \in A] > \epsilon, \sup_A \frac{1}{qt} \sum_{i=1}^{qt} [\bar{B}_i \in A] > \epsilon, t = 1, \dots, T \right).$$

Denote by  $W$  the event that the distribution of the relaxed Rogers operator is accurate. By using (3.10), this event can be written with

$$W = \left( \sup_A |\nu_n(A) - \nu(A)| > \epsilon \right).$$

By conditioning, the probability of error becomes

$$P(|\bar{V} - V| > \epsilon) = P\left(|\bar{V} - V| > \epsilon \mid U, W\right) P(U, W) + P\left(|\bar{V} - V| > \epsilon \mid \overline{(U, W)}\right) P(\overline{(U, W)}),$$

where the notation  $(U, W)$  means that both the event  $U$  and  $W$  hold, and the notation  $\overline{(U, W)}$  is the complementary event. By (3.7), (3.8), (3.10), and the union bound, the second term tends to zero when  $(p, q)$  and  $N$  are large, with  $p_{\max}^2 q_{\max}^2 < o(N)$ . For the first term, the event  $U$  is assumed to imply that the optimal martingale can be approximated accurately with a projected martingale. The estimation error can hence be written with

$$|\bar{V} - V| = \left| \frac{1}{N} \sum_{n=1}^N D(\bar{R}^m(\bar{Z}), S^n) - E[D(\bar{M}, S)] \right|.$$

where  $\bar{M}$  is the projection of the optimal martingale, and the equality is in probability. The event  $W$  is assumed to imply that the distribution underlying the relaxed Rogers operator is accurate, so that the following equality in distribution holds

$$\frac{1}{N} \sum_{n=1}^N D(\bar{R}^m(\bar{Z}), S^n) = E[D(\bar{M}, S)].$$

As the last equality is an equality in distribution for scalars, the equality also holds in probability, and  $P\left(|\bar{V} - V| > \epsilon \mid U, W\right)$  vanishes when  $(p, q)$  and  $N$  are large, with  $p_{\max}^2 q_{\max}^2 < o(N)$ . The consistency result (3.13) now follows.

For the consistency of the projected dual value (3.12), the steps are the same as above, except that an additional step is needed. Indeed, the estimation error is now evaluated with an out-of-sample test, and the estimation error can be written with

$$|\bar{V} - V| = \left| \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} D(\bar{R}^m(\bar{Z}), S^{\tilde{n}}) - E[D(M, S)] \right|.$$

Apply the above steps, with the modification that the event  $W$  now implies the following equality

in distribution

$$\bar{R}^m(\bar{Z}) = \bar{M}.$$

It hence remains to show that  $P(|\bar{V} - V| > \epsilon | U, W)$  vanishes. To this end, use (3.11) to show that the quantity vanishes when  $(p, q)$  and  $N$  are large, with  $p_{\max}^3 q_{\max}^3 < o(N)$ . The consistency result (3.12) now follows.

### 9.13 NNM Convergence in Bayes-Value (3.14)

Denote by  $D(\bar{R}(S^{(N)}), S^{(\tilde{N})})$  the vector  $(D(\bar{R}(S^{(N)}), S^n), n = 1, 2, \dots, \tilde{N})$ , we bound the covering number of the dual payoff  $\mathcal{N}(\epsilon, D(\bar{R}(S^{(N)}), S^{(\tilde{N})}))$  with the covering number of the general relaxed Rogers operator. Let  $S^{(N)}$  and  $S'^{(N)}$  be two random sample, by the Lipschitz assumption,

$$\begin{aligned} \|D(\bar{R}(S^{(N)}), S^{(\tilde{N})}) - D(\bar{R}(S'^{(N)}), S^{(\tilde{N})})\| &= \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \|D(\bar{R}(S^{(N)}), S^{\tilde{n}}) - D(\bar{R}(S'^{(N)}), S^{\tilde{n}})\| \\ &\leq \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} K \|\bar{R}(S^{(N)}) - \bar{R}(S'^{(N)})\| \\ &= K \|\bar{R}(S^{(N)}) - \bar{R}(S'^{(N)})\|. \end{aligned}$$

Proving that

$$\mathcal{N}(\epsilon, D(\bar{R}(S^{(N)}), S^{(\tilde{N})})) \leq \mathcal{N}\left(\frac{\epsilon}{K}, \bar{R}(S^{(N)})\right).$$

The general relaxed Rogers operator is a mapping from a random sample to a projected martingale. The operator is hence a function of the form  $S^{(N)} \times \bar{M}$ . The covering number of a function can be bound by the product of the covering number of the domain and the codomain, giving

$$\mathcal{N}(\epsilon, \bar{R}(S^{(N)})) \leq \mathcal{N}\left(\frac{\epsilon}{2}, S^{(N)}\right) \mathcal{N}\left(\frac{\epsilon}{2}, \bar{M}\right).$$

The covering number for a random sample  $S^{(N)}$  is equivalent to the covering number for a random Voronoi tessellation. By (9.10), we have

$$\mathcal{N}(\epsilon, S^{(N)}) \leq cdTN^3 \left(\frac{c}{\epsilon}\right)^{cdTN^3}.$$

The covering number for a projected martingale is given by (9.17) and is

$$\mathcal{N}(\epsilon, \bar{M}) \leq (cdTp_{\max}^3 q_{\max}^3)^T \left(\frac{U}{\epsilon}\right)^{cdT^2 p_{\max}^3 q_{\max}^3}.$$

By combining all the previous results, we have

$$\begin{aligned} \mathcal{N}(\epsilon, D(\bar{R}(S^{(N)}), S^{(\tilde{N})})) &\leq \mathcal{N}\left(\frac{\epsilon}{K}, \bar{R}(S^{(N)})\right) \\ &\leq \mathcal{N}\left(\frac{\epsilon}{2K}, S^{(N)}\right) \mathcal{N}\left(\frac{\epsilon}{2K}, \bar{M}\right) \\ &\leq cdTN^3 \left(\frac{cK}{\epsilon}\right)^{cdTN^3} (cdTp_{\max}^3 q_{\max}^3)^T \left(\frac{UK}{\epsilon}\right)^{cdT^2 p_{\max}^3 q_{\max}^3} \\ &\leq (cdTp_{\max}^3 q_{\max}^3 N^3)^T \left(\frac{UK}{\epsilon}\right)^{cdT^2 p_{\max}^3 q_{\max}^3 N^3}. \end{aligned}$$

The result now follows by Theorem 29.1 in Devroye, Györfi, and Lugosi (1996).

## 9.14 NNM Universal Bayes-Value Consistency (3.15, 3.16)

To prove the universal Bayes-value efficiency, assume that the Rogers operator is restricted to the class of martingale implied by the relaxed Rogers operator. For the projected dual value this assumption allows to write

$$V = \sup_{\bar{R}(S^{(N)})} E(D(\bar{R}(S^{(N)}), S)).$$

For the relaxed dual value, this assumption allows to write

$$V = \sup_{\bar{M}} E(D(\bar{M}, S)).$$

For the projected dual value, use Lemma 8.2 in Devroye, Györfi, and Lugosi (1996) to write

$$\begin{aligned} & \left( \sup_{\bar{R}(S^{(N)})} \frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} D(\bar{R}(S^{(N)}), S^n) \right) - \left( \sup_{\bar{L}} E(D(\bar{R}(S^{(N)}), S)) \right) \\ & \leq 2 \sup_{\bar{R}(S^{(N)})} \left| \frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} D(\bar{R}(S^{(N)}), S^n) - E(D(\bar{R}(S^{(N)}), S)) \right|. \end{aligned}$$

By using (3.14), and Problem 12.1 in Devroye, Györfi, and Lugosi (1996), the expectation of the right-hand side is bounded by

$$\sqrt{\frac{\log \left( 8e(cdTp_{\max}^3 q_{\max}^3 N^3)^T \left( \frac{UK}{\epsilon} \right)^{cdT^2 p_{\max}^3 q_{\max}^3 N^3} \right)}{\tilde{N}/(128B^2)}}.$$

This bound implies that  $E(|\bar{V} - V|)$  converges to zero. The universal consistency (3.15) follows.

Similarly, for the relaxed dual value, write

$$\left( \sup_{\bar{M}} \frac{1}{N} \sum_{n=1}^N D(\bar{M}, S^n) \right) - \left( \sup_{\bar{M}} E(D(\bar{M}, S)) \right) \leq 2 \sup_{\bar{M}} \left| \frac{1}{N} \sum_{n=1}^N D(\bar{M}, S^n) - E(D(\bar{M}, S)) \right|.$$

Then, using (3.11), the expectation of the right-hand side is bounded by

$$\sqrt{\frac{\log \left( 8e(cdTp_{\max}^3 q_{\max}^3)^T \left( \frac{UK}{\epsilon} \right)^{cdT^2 p_{\max}^3 q_{\max}^3} \right)}{N/(128B^2)}}.$$

The universal consistency (3.16) follows.

## 9.15 An American Option: Value Function Martingale Part (Section 4.2.1)

For an American put, the value function  $v_t$  is a function of the last observed stock price  $S_t$ . This value function can be written with

$$v_t(S_t) = \max_{X_{t:T}} E \left[ \sum_{s=t}^T e^{-rt} (K - S_s)_+ X_s \middle| S_t \right],$$

and can be found with an implicit finite difference scheme. To simulate a sample martingale path  $M$  of the value function martingale part, the procedure is as follows. First, sample a stock path  $S$  and obtain a sample value function path with

$$V = (v_0(S_0), v_1(S_1), \dots, v_T(S_T)),$$

Second, set  $M_0 = 0$ , and for each time  $t = 0, 1, \dots, T - 1$  obtain a sample of the next period stock price conditional on the current stock price. This sample can be written with

$$\{S_{t+1}^i | S_t\}_{i=1}^n.$$

For each next period price find the value function realization

$$\{V_{t+1}^i | S_t\}_{i=1}^n = \{v_t(S_{t+1}^i) | S_t\}.$$

Set the martingale realization in the next period as the current martingale plus the centered value function

$$M_{t+1} = M_t + V_{t+1} - \frac{1}{n} \sum_{i=1}^n V_{t+1}^i.$$

## 9.16 An American Option: Dual Metrics (4.3, 4.4)

To quantize a dual martingale use two samples: A sample of the stock path  $\{S^i\}_{i=1}^n$  and a corresponding sample of the martingale  $\{M^i\}_{i=1}^n$  generated by the stock path. Run Lloyd's algorithm (Lloyd 1982, Arthur and Vassilvitskii 2007) to split the martingale sample into 10 Voronoi cells. Denote these cells by  $\{B_1, B_2, \dots, B_{10}\}$ . As the martingale sample is generated by the stock sample, these Voronoi cells also define a tessellation on the stock path. This implicit tessellation can be written with

$$B'_i = \{S^i : M^i \in B_i\},$$

where  $B'_i$  is the  $i$ -th cell of the implicit tessellation, for  $i = 1, 2, \dots, 10$ . This implicit tessellation allows to map any metrics to the stock domain, and in turn, the exercise boundary domain. To visualize a metric in the exercise boundary domain, find the stock centroid in the implicit tessellation. This centroid can be written with

$$\hat{S}^i = \frac{1}{m} \sum_{j=1}^m S^j,$$

where the addition is vectorial,  $m$  is the size of the implicit cell  $B'_i$ , and  $S^j \in B'_i$  is a stock path in the implicit cell  $B'_i$ . Now, any metric on a martingale cell  $B_i$  can be visualized in the exercise boundary domain. It suffices to map the metric to the corresponding implicit cell  $B'_i$ , and to represent the metric along the stock centroid  $\hat{S}^i$ .

To compare the metrics of multiple dual martingales, fix the martingale tessellation and compute the metric conditional on the fixed tessellation. For example, let  $\{B_i\}$  be a tessellation of a sample  $\{M_i\}$  of optimal martingale path. Such a sample can be obtained with the method presented in the previous section. Consider the dual exercise distribution metric. This metric  $\mu_1(\{M^j\}, \{S^j\})$  is a function of a martingale path sample  $\{M^j\}$ , and its generating stock path sample  $\{S^j\}$ . This

metric can be written with

$$\mu_1(\{M^j\}, \{S^i\}) = \mathcal{H} \{ (S_{t^j}^j, t^j) : t^j \text{ is a dual exercise time} \},$$

where  $\mathcal{H}$  denotes the histogram operator, and the dual exercise time  $t^j$  of a stock path  $S^j$  is the time  $t$  that maximizes the dual payoff

$$\max_x \sum_{t=0}^T e^{-rt} (K - S_t^j)_+ x_t - x_t M_t^j.$$

The histogram operator gives a set  $\{(S_a, S_b, t_a, t_b, p)\}$  such that the probability that the dual exercise fall in the interval  $[S_a, S_b) \times [t_a, t_b)$  is  $p$ . The dual exercise metric is a set-valued estimate, and can be represented along the implicit stock centroid  $\hat{S}^i$  with a linewidth gradient propotional to the probability  $p$ . Similarly, the  $\ell_1$ -average metric  $\mu_2(t, \{M^j\}, B_i)$  can be written with

$$\mu_2(t, \{M^j\}, B_i) = \frac{1}{n} \sum_{j=1}^n |M^j|,$$

where the sum is over the martingale path  $M^j$  that falls in the reference cell  $B_i$ , and  $n$  is the number of such path. The  $\ell_1$ -average metric is a point estimate and can be represented with a linewidth gradient along the reference implicit stock centroid  $\hat{S}^i$ .

## 9.17 Implementation

To run SPLS and NNM, we use a 100 processors machine with 100 GB RAM in the Google Cloud (2020), and we use the linear programming solver Gurobi (2020). To find the nearest-neighbor of a path among a set of centroids we use the multiple random projection technique of Hyvönen et al. (2016).



## References

- Aleksandrov, N. and B. Hambly (2010) A dual approach to multiple exercise option problems under constraints. *Mathematical methods of operations research* **71** (3), 503–533.
- Andersen, L. and M. Broadie (2004) Primal-dual simulation algorithm for pricing multidimensional American options. *Management Science* **50** (9), 1222–1234.
- Arthur, D. and S. Vassilvitskii (2007) k-means++: The advantages of careful seeding. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1027–1035.
- Bardou, O., S. Bouthemy, and G. Pages (2009) Optimal quantization for the pricing of swing options. *Applied Mathematical Finance* **16** (2), 183–217.
- Baringhaus, L. and C. Franz (2004) On a new multivariate two-sample test. *Journal of multivariate analysis* **88** (1), 190–206.
- Barrera-Esteve, C. et al. (2006) Numerical methods for the pricing of swing options: a stochastic control approach. *Methodology and computing in applied probability* **8** (4), 517–540.
- Becker, S., P. Cheridito, and A. Jentzen (2019) Deep Optimal Stopping. *Journal of Machine Learning Research* **20** (74), 1–25.
- Bender, C. (2011a) Dual pricing of multi-exercise options under volume constraints. *Finance and Stochastics* **15** (1), 1–26.
- Bender, C. (2011b) Primal and dual pricing of multiple exercise options in continuous time. *SIAM Journal on Financial Mathematics* **2** (1), 562–586.
- Bender, C. and J. Schoenmakers (2006) An iterative method for multiple stopping: convergence and stability. *Advances in applied probability*, 729–749.
- Bender, C., J. Schoenmakers, and J. Zhang (2015) Dual representations for general multiple stopping problems. *Mathematical Finance* **25** (2), 339–370.
- Bisschop, J. (2016) *Aimms Optimization Modeling*. <https://aimms.com/english/developers/resources/manuals/optimization-modeling/>.
- Black, F. and M. Scholes (1973) The pricing of options and corporate liabilities. *The journal of political economy*, 637–654.

- Browne, C. B. et al. (2012) A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games* **4 (1)**, 1–43.
- Carmona, R. and S. Dayanik (2008) Optimal multiple stopping of linear diffusions. *Mathematics of Operations Research* **33 (2)**, 446–460.
- Carmona, R. and N. Touzi (2008) Optimal multiple stopping and valuation of swing options. *Mathematical Finance* **18 (2)**, 239–268.
- Carr, P., R. Jarrow, and R. Myneni (1992) Alternative characterizations of American put options. *Mathematical Finance* **2 (2)**, 87–106.
- Cox, J. C., S. A. Ross, and M. Rubinstein (1979) Option pricing: A simplified approach. *Journal of financial Economics* **7 (3)**, 229–263.
- Dahlgren, M. (2005) A continuous time model to price commodity-based swing options. *Review of derivatives research* **8 (1)**, 27–47.
- Dahlgren, M. and R. Korn (2005) The swing option on the stock market. *International Journal of Theoretical and Applied Finance* **8 (01)**, 123–139.
- Devroye, L., L. Györfi, and G. Lugosi (1996) *A probabilistic theory of pattern recognition*. Springer Science.
- Frank, M. and P. Wolfe (1956) An algorithm for quadratic programming. *Naval research logistics quarterly* **3 (1-2)**, 95–110.
- Google Cloud (2020) *Google Cloud Platform*. <http://cloud.google.com>.
- Gurobi (2020) *Gurobi Optimizer Reference Manual*. <http://www.gurobi.com>.
- Harrison, J. M. and D. M. Kreps (1979) Martingales and arbitrage in multiperiod securities markets. *Journal of Economic theory* **20 (3)**, 381–408.
- Haugh, M. B. and L. Kogan (2004) Pricing American options: a duality approach. *Operations Research* **52 (2)**, 258–270.
- Heitsch, H., W. Römisch, and C. Strugarek (2006) Stability of multistage stochastic programs. *SIAM Journal on Optimization* **17 (2)**, 511–525.
- Hyvönen, V. et al. (2016) Fast nearest neighbor search through sparse random projections and voting. *Big Data (Big Data), 2016 IEEE International Conference on*. IEEE, 881–888.

- Jaillet, P., E. I. Ronn, and S. Tompaidis (2004) Valuation of commodity-based swing options. *Management science* **50** (7), 909–921.
- Jain, S. and C. W. Oosterlee (2012) Pricing high-dimensional Bermudan options using the stochastic grid method. *International Journal of Computer Mathematics* **89** (9), 1186–1211.
- Lari-Lavassani, A., M. Simchi, and A. Ware (2001) A discrete valuation of swing options. *Canadian applied mathematics quarterly* **9** (1), 35–74.
- Lelong, J. (2016) Pricing American options using martingale bases. *arXiv preprint arXiv:1604.03317*.
- Lloyd, S. (1982) Least squares quantization in PCM. *IEEE transactions on information theory* **28** (2), 129–137.
- Longstaff, F. A. and E. S. Schwartz (2001) Valuing American options by simulation: a simple least-squares approach. *Review of Financial studies* **14** (1), 113–147.
- Marshall, T. J. and R. M. Reesor (2011) Forest of stochastic meshes: A new method for valuing high-dimensional swing options. *Operations Research Letters* **39** (1), 17–21.
- Meinshausen, N. and B. Hambly (2004) Monte Carlo methods for the valuation of multiple-exercise options. *Mathematical Finance* **14** (4), 557–583.
- Merton, R. C. (1973) Theory of rational option pricing. *The Bell Journal of economics and management science*, 141–183.
- Nadarajah, S., F. Margot, and N. Secomandi (2013) *Improved least squares Monte Carlo for term structure option valuation with energy applications*. Tech. rep. Technical report 2012-E54, Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA.
- Penaud, A., P. Wilmott, and H. Ahn (1999) Exotic passport options. *Asia-Pacific Financial Markets* **6** (2), 171–182.
- Pflug, G. C. and A. Pichler (2012) A distance for multistage stochastic optimization models. *SIAM Journal on Optimization* **22** (1), 1–23.
- Ramdas, A., N. Trillos, and M. Cuturi (2017) On wasserstein two-sample testing and related families of nonparametric tests. *Entropy* **19** (2), 47.
- Rogers, L. (2002) Monte Carlo valuation of American options. *Mathematical Finance* **12** (3), 271–286.

- Rogers, L. (2007) Pathwise stochastic optimal control. *SIAM Journal on Control and Optimization* **46 (3)**, 1116–1132.
- Schoenmakers, J. (2012) A pure martingale dual for multiple stopping. *Finance and Stochastics* **16 (2)**, 319–334.
- Silver, D. et al. (2017) Mastering the game of go without human knowledge. *Nature* **550 (7676)**, 354.
- Székely, G. J., M. L. Rizzo, et al. (2004) Testing for equal distributions in high dimension. *InterStat* **5 (16.10)**, 1249–1272.
- Thompson, A. C. (1995) Valuation of path-dependent contingent claims with multiple exercise decisions over time: The case of take-or-pay. *Journal of Financial and Quantitative Analysis* **30 (2)**, 271–293.
- Van Der Vaart, A. and J. A. Wellner (1996) *Weak convergence and empirical processes: with applications to statistics*. Springer.
- Van Der Vaart, A. and J. A. Wellner (2009) A note on bounds for VC dimensions. *Institute of Mathematical Statistics collections* **5**, 103.
- Vapnik, V. N. and A. Y. Chervonenkis (1982) Necessary and sufficient conditions for the uniform convergence of means to their expectations. *Theory of Probability & Its Applications* **26 (3)**, 532–553.
- Vapnik, V. and A. Y. Chervonenkis (1971) On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory of Probability and its Applications* **16 (2)**, 264.
- Wilhelm, M. and C. Winter (2008) Finite element valuation of swing options. *Journal of Computational Finance* **11 (3)**, 107.

## Chapter 3

# Valuation of the Reinvestment Option in Equity-Indexed Annuities

# Valuation of the Reinvestment Option in Equity-Indexed Annuities

Nicolas Essis-Breton and Patrice Gaillardetz

March 2020

---

## Abstract

This paper considers the valuation of equity-indexed annuity (EIA) with periodic premiums. Together with various mortality, surrender and financial options, these products are offered with a reinvestment option: the policyholder has the right to reinvest in the contract. This reinvestment option allows the policyholder to increase or slow down participation in the equity market, as each reinvestment accumulates interest according to the market indexing scheme. We consider two indexing schemes. A fund protection scheme that guarantees a minimal global investment return, and a premium protection scheme that guarantees a minimal return on each reinvestment made by the policyholder. The former is inspired by variable annuity product, while the latter is a new scheme. We propose the second scheme to help EIA issuers to differentiate themselves from alternative investments offered by banks. EIAs are generally priced through pricing parameters that define the proportion of return from the indexing scheme credited to the policyholder. We obtain the fair pricing parameters that makes the present value of the contract zero under the assumption that the policyholder is rational. Such pricing settings can be approached either exactly with a dynamic programming algorithm, or approximatively with the single pass lookahead search algorithm. In the numerical examples, we present fair pricing parameters for contracts of 30 years that include both an indexing scheme, a surrender option, and a reinvestment option. The gained rate used in the indexing scheme is either a point-to-point rate or an annual reset rate.

---

# 1 Introduction

An Equity-Indexed Annuity (EIA) provides limited participation in the performance of an equity index such as the S&P 500 while guaranteeing a minimum rate of return. The guaranteed minimum rate of return is contingent on both the return of the financial market and the mortality of the policyholder. For example, the guaranteed return is credited only if the equity market return is too low, or only if the policyholder dies. EIA also offers annuitization options at retirement and the right to completely or partially surrender the contract. EIA differs from investment products offered by banks in that EIA offers both financial and mortality guarantees, while bank products are limited to financial guarantees only. EIAs were introduced by Keyport Life Insurance in 1995 and have been the most innovative insurance product in recent years. They have become increasingly popular since their debut and EIA sales reached \$60 billion in 2016, and \$70 billion in 2018. The total assets in 2018 are \$591 billion. For variable annuity, a product similar to EIA and for which the tools presented here apply, the total sales in 2018 reached \$93 billion and the total assets were \$1.8 trillion. See the Insured Retirement Institute (2019) for more sales figures. Participating insurance product, and unit-linked product are also similar to EIA.

Equity-linked products have been intensively studied in previous decades. Brennan and Schwartz (1976) and Boyle and Schwartz (1977) were the first to extend the risk-neutral pricing framework (Black and Scholes 1973, Merton 1973) to insurance products linked to the financial market. In this extension, the financial risk and the biometric risk are considered independent, so that the financial guarantees can be priced by standard option pricing theory, while the biometric risk is hedged away by the insurer through diversification. This extension is since the reference framework to analyze the various guarantees offered in EIAs. See Hardy (2003) and Møller and Steffensen (2007) for a comprehensive study of the various guarantees. Tiong (2000) and Lee (2003) obtained closed-form formulas for several EIAs. Lin and Tan (2003) and Kijima and Wong (2007) use a general model, in which both the equity index and the interest rate satisfy general stochastic differential equations.

In discrete time, Bacinello (2003a, 2005) values the surrender option for participating insurance contracts under a binomial model for the equity index (Cox, Ross, and Rubinstein 1979), while Andreatta and Corradin (2003) value the surrender option by least squares Monte-Carlo (Carriere 1996, Longstaff and Schwartz 2001). Gaillardetz and Lin (2006), Yuen and Yang (2010) and Wei et al. (2013) use stochastic volatility and interest rate under a lattice-based model for the equity index. Consiglio and De Giovanni (2008) consider the valuation of EIA in incomplete financial models. Moore (2009) evaluates EIA under the principle of equivalent utility.

In all the previous articles, except in Brennan and Schwartz (1976), only single premium EIA contracts are considered. The policyholder pays a lump sum at contract inception and no further reinvestment is allowed. This assumption differs greatly from contracts commonly sold on the market. Many contracts allow the policyholder to reinvest in the contract. These reinvestments are not mandatory and their amounts and timing are decided by the policyholder. There have been few studies on contracts sold with a mandatory periodic premium. In such contracts, a fixed premium amount must be paid periodically, for example monthly, to keep the contract in force. See Bacinello (2003b), Costabile et al.(2008), Chi and Lin (2012), and Bernard et al. (2017). However, these periodic premium studies do not consider the amount and timing flexibility allowed in reinvestment. We call such a flexible reinvestment right a *reinvestment option*. The valuation of this reinvestment option is crucial for adequate risk management of EIA products. The goal of this article is to provide pricing algorithms applicable to a wide variety of contracts sold with reinvestment options.

The first step in studying the reinvestment option is to define an indexing scheme. The indexing scheme prescribes how the equity market return and the minimum guaranteed return are credited to the reinvestments. This step is necessary because there is currently no standard way of crediting return to reinvestments in EIA. We consider two indexing schemes. With the fund protection scheme, the policyholder receives the maximum between the reinvestments as if they were invested in the equity market only, and as if they were invested at the minimum guaranteed rate only. This scheme protects the global return of the policyholder. This scheme is commonly used in variable annuity offering a reinvestment option, but is not a standard scheme for EIA. A shortcoming of this scheme is that the return on individual reinvestment is not guaranteed, and this could make the product less appealing to some policyholders. To remediate this situation, we study a new premium



protection scheme where each reinvestment is credited the maximum between the equity market return and the minimum guaranteed return. This new scheme is of course more costly, but may appeal to some investors. However, in the context of EIA based on the indexing scheme, this new scheme can be hedged with the same ease than the fund protection scheme. In particular, when the indexing scheme is defined against a stock market index, rather than a portfolio of funds, as is done in variable annuity, an hedging strategy based on the Greeks is sufficient to hedge the guarantees associated with the reinvestment option.

With the indexing scheme defined, we can turn to valuation algorithms for the reinvestment option. For EIA offering American-style option, such as the surrender option, the valuation approach commonly encountered in the literature is dynamic programming (Bertsekas 2005). The problem of this approach for the reinvestment option is that the approach cannot scale to contract with long maturities. Especially when the indexing scheme is path-dependent. However, this approach is a good starting point to understand the major risk drivers in EIA and we present a valuation algorithm based on this optimization technic. This valuation algorithm is also specialized to the case where the index follows a binomial model. To price contract with long maturities, we resort to the Single Pass Lookahead Search (SPLS) algorithm (Essis-Breton and Gaillardetz 2020). A recently developed algorithm for the pricing of constrained multiple exercise American options, a class of options to which EIAs belong. SPLS allows to price American options by Monte Carlo simulation and linear programming: along a sample stock path, an adapted exercise strategy is extracted by linear programming, and the option price is the average value of several such extracted strategies. SPLS is a fast and fully general algorithm that applies to any option pricing problem expressed as a mathematical programming problem. As any EIA can be expressed as a mathematical programming problem, SPLS applies to any EIA currently sold in the market. In particular, SPLS allows to price contracts where reinvestment and surrender options are jointly offered.

This paper is organized as follows. Section 2 briefly recalls the risk-neutral pricing framework for EIAs, introduces the financial model, and presents the actuarial notation. Section 3 studies the indexing scheme for EIA offering a reinvestment option. Section 4 displays a dynamic programming valuation algorithm, while Section 5 introduces an SPLS algorithm. Section 6 contains numerical examples. Section 7 concludes.

## 2 Risk-Neutral Pricing Framework

Following the valuation paradigm of Brennan and Schwartz (1976) and Boyle and Schwartz (1977) the risk-neutral biometric measure is assumed to be identical to the physical biometric measure. This assumption is equivalent to a risk-minimizing strategy for the mortality risk (see Møller, 1998). The financial risk-neutral measure is assumed independent of the biometric risk-neutral measure, and the risk-neutral measure  $\mathbb{P}$  is the product of these two measures. Denote the index price process by  $S$  and the complete future lifetime of the policyholder by  $T_x$ . The filtration  $F$  of the contract is the filtration generated by the index process  $S$  and the lifetime of the policyholder  $T_x$ .

Denote by  $n$  the contract maturity and denote by  $T_s$  the surrender time. The contract duration  $T$  is the earliest between the contract maturity, the policyholder lifetime and the surrender time

$$T = \min(n, \lfloor T_x \rfloor + 1, T_s),$$

where  $\lfloor \cdot \rfloor$  is the floor function.

For the financial model, denote by  $r$  the constant risk-free rate, and denote by  $v$  the discounting factor  $\exp(-r)$ . The risk-neutral dynamic of the index is assumed to be

$$dS = rSdt + \sigma SdW,$$

where  $r$  is the risk-free rate,  $\sigma$  is the volatility, and  $W$  is a standard Brownian motion. The market is assumed frictionless, with no tax and no transaction cost. The reinvestment option and the surrender option are assumed exercised at discrete times  $0, 1, \dots, n$ . The reinvestment option and the surrender option can be exercised at the beginning of a period. When the surrender option is exercised, the exercise takes place before the reinvestment exercise and terminates the contract. Death is assumed to be recorded during a period, after the reinvestment option is exercised, and the mortality benefit is paid at the end of a period. Upon payment of the mortality benefit, the contract terminates. For the actuarial notation, denote by  $q_{x+k}$  the probability that a policyholder

of age  $(x + k)$  dies in the next year. This probability can be written with

$$q_{x+k} = \mathbb{P}(k < T_x \leq k + 1 | T_x > k),$$

and denote by  $p_{x+k}$  the complementary probability that the policyholder survives

$$p_{x+k} = 1 - q_{x+k}.$$

### 3 Indexing Schemes

The starting point to define indexing schemes for the reinvestment option is the gained rate used in single premium contracts. The gained rate is the return credited to the initial premium at maturity. To facilitate policyholder's adoption, the indexing scheme should be closed to the gained rate already offered in EIA. From Lin and Tan (2003) and Tiong (2000), there are two broad designs for the gained rate: the point-to-point design and the annual reset design. With a point-to-point gained rate, the credited return is based on the index return between the inception and the maturity of the contract. While with an annual reset gained rate, the credited return is based on the annual return index. For both designs, the effective credited return is a fraction  $\alpha$  of the credited return. This fraction is interpreted as a participation rate in the index return and is called the participation rate. The effective credited return may be subject to a limit  $\zeta$ , called the cap rate. Specifically, let  $R$  be the gained rate. For a point-to-point gained rate, the gained rate  $R_{t,T}$  between time  $t$  and the maturity  $T$  can be written with

$$R_{t,T} = \min \left( 1 + \alpha \left( \frac{S_T}{S_t} - 1 \right), (1 + \zeta)^{T-t} \right). \quad (3.1)$$

For a single premium contract, the anchor time  $t$  is always the inception time. A direct extension of this design to the reinvestment option allows the anchor time  $t$  to correspond to the time of reinvestment. For an annual reset design, the gained rate  $R_{t,T}$  between time  $t$  and the maturity  $T$

can be written with

$$R_{t,T} = \prod_{k=t}^{T-1} \min \left( \max \left( 1 + \alpha \left( \frac{S_{k+1}}{S_k} - 1 \right), 1 \right), 1 + \zeta \right). \quad (3.2)$$

Similarly to the point-to-point gained rate, the anchor time  $t$  corresponds to the time of reinvestment. The main difference between a point-to-point gained rate and an annual reset gained rate is that the annual reset gained rate offers an additional level of guarantee where the returns of the index during bad years are ignored. Other gained rate designs are sometimes offered. For example, in a highwater mark design for a single premium contract, the credited return is a function of the highest index price recorded during the contract. Similarly to the two previous gained rates, this gained rate can be extended to the reinvestment option by allowing the anchor time to correspond to the time of reinvestment.

With the gained rate, we can study the indexing scheme. To this end, denote by  $Y_t$  the reinvestment of the policyholder at time  $t$ . The reinvestment may be subject to several constraints. For example, the initial investment may be of at least  $y_0$ . Also, subsequent reinvestment may be limited to be in the interval  $[y_{min}, y_{max}]$ . If the policyholder can waive the reinvestment right,  $y_{min}$  is fixed to 0. The aggregate amount of reinvestment may also be limited.

We consider two indexing schemes. The fund protection scheme guarantees a minimum overall return of  $g$  to the policyholder, while the premium protection scheme guarantees a minimum return of  $g$  on each reinvestment. For both indexing schemes, the participation of the policyholder in the guaranteed return is fixed to a fraction  $\beta$ , and this fraction is called the guaranteed rate participation. To write these indexing schemes, denote by  $D_T$  the payoff of the contract at time  $T$ . The payoff of the fund protection scheme can be written with

$$D_T = \max \left( \sum_{t=0}^{T-1} R_{t,T} Y_t, \beta \sum_{t=0}^{T-1} (1+g)^{T-t} Y_t \right), \quad (3.3)$$

while the payoff of the premium protection scheme can be written with

$$D_T = \sum_{t=0}^{T-1} \max \left( R_{t,T}, \beta (1+g)^{T-t} \right) Y_t. \quad (3.4)$$

The main difference between these two schemes is that for the fund scheme each reinvestment increases the guaranteed amount without protecting individual premiums, while for the premium scheme each reinvestment increases directly the guaranteed amount.

## 4 Dynamic Programming Algorithm

In a contract offering reinvestment and surrender options, the decision processes are the reinvestment process and the surrender stopping time. To price a contract with a dynamic programming algorithm, these decision processes need to be turned into a Markov decision process. When a decision process is Markovian, the decision at time  $t$  depends only on the value of another adapted process called the state variable. The state variable process captures all the information necessary to make a decision at each time point. To find a state variable for a particular indexing scheme, the time  $t$ , the reinvestment decision  $Y_t$ , and the surrender decision at time  $t$  need to be fixed. Then, consider all the variables needed to compute the value of the contract with payoff  $D_t$ . These variables form the state variable. The state variable is usually not uniquely defined, and the definition with the fewest number of variables is preferred. For some indexing schemes, the state variable can be limited to a few variables, while for some other indexing schemes, the state variable is the entire index and reinvestment history. Examples of these extremes are a premium protection scheme with an annual reset gained rate, and a fund protection scheme with a point-to-point gained rate. Table 4.1 summarizes the possible state variable per indexing scheme and gained rate.

Indexing scheme	Gained rate	State variable at time $t$
fund protection	point-to-point	$((Y_k)_{k=0}^{t-1}, (S_k)_{k=0}^t)$
	annual reset	$(\sum_{k=0}^{t-1} R_{t,T} Y_t, \beta \sum_{k=0}^{t-1} (1+g)^{T-t} Y_t, S_t)$
premium protection	point-to-point	$((Y_k)_{k=0}^{t-1}, (S_k)_{k=0}^t)$
	annual reset	$(D_{t-1}, S_t)$

Table 4.1: State variable needed in the dynamic programming algorithm by indexing scheme and gained rate.

With the state variable defined, a dynamic programming algorithm will follow once we define the Bellman equation. To obtain the Bellman equation, define the value function  $V$  as the difference in the present value of the contract payoff and the remaining contract premiums given that the

policyholder is alive. The contract payoff is the payoff  $D_T$  at the contract termination time  $T$ , while the contract premiums are the reinvestments. To discourage the early exercise of the surrender option, the contract payoff is reduced by a surrender penalty  $\delta_T$ . The surrender penalty is usually substantial at the beginning of the contract, for example 80%, and negligible near the maturity of the contract, for example 100%. The value function is found under the assumption that the policyholder is rational, and that the policyholder follows the optimal reinvestment and surrender exercise strategy. With  $Y_{t:T} = (Y_t, Y_{t+1}, \dots, Y_T)$  the residual reinvestments, and  $H_t$  the state variable at time  $t$ , the value function at time  $t$  can be written with

$$V_t = \max_{T, Y_{t:T}} \mathbb{E} \left( v^{T-t} \delta_T D_T - \sum_{k=t}^{T-1} v^{k-t} Y_k \middle| H_t \right), \quad (4.1)$$

where the expectation is taken under the risk-neutral measure and  $t = 0, 1, \dots, T$ . The Bellman equation follows by separating the value function at time  $t$  into the payoff credited at time  $t$  and the future payoffs. The future payoffs are then simplified by identifying them with the value function at time  $t+1$ . In this simplification, the independence assumption between the financial risk-neutral measure and the biometric risk-neutral measure is used. This procedure leads to the following Bellman equation

$$V_t = \max_{Y_t} \mathbb{E} (\max (\delta_t D_t, v q_{x+t} \delta_{t+1} D_{t+1} + v p_{x+t} V_{t+1} - Y_t) | H_t), \quad (4.2)$$

where the first term in the inner maximum is the payoff received in case of immediate exercise of the surrender option. The second term contains the continuation value in case of death of the policyholder and the continuation value in case of survival of the policyholder.

By further simplifying the equity index process with a binomial model (Cox, Ross, and Rubinstein 1979), a recursive formula for the Bellman equation can be obtained. Recall that in a binomial model with  $N$  trading date per period, the return of the index from time  $t$  to  $t+1$  can take  $N+1$  possible values. The probability distribution for these values can be written with

$$w_i = \mathbb{P} \left( \frac{S_{t+1}}{S_t} = u^i d^{N-i} \right) = \binom{N}{i} p^i q^{N-i}, \quad (4.3)$$

where the up move  $u$  is  $\exp(\sigma/\sqrt{N})$ , the down move  $d$  is  $1/u$ , the up move probability  $p$  is  $(\exp(r/N) - d)/(u - d)$ , the down move probability  $q$  is  $1 - p$ , and  $i = 0, 1, \dots, N$ . With these assumptions, the Bellman equation simplifies to

$$V_t = \max_{Y_t} \max \left( \delta_t D_t, \sum_{i=0}^N w_i v(q_{x+t} \delta_{t+1} D_{t+1} + p_{x+t} V_{t+1} | u^i, H_t) - Y_t \right), \quad (4.4)$$

where the notation  $(Z_{t+1} | u^i, H_t)$  is the random variable  $Z_{t+1}$  given that  $i$  up moves occurred between time  $t$  and  $t+1$ , and given the state variable  $H_t$ . To use this recursive formula, it suffices to construct a grid for the state variable  $H$  from time  $t = 0$  to time  $t = n$ . Then apply the recursive formula backward, starting from the grid point at time  $t = n$ .

Under the equivalence premium principle (Young 2006), a contract is priced such that the value function at time zero is zero. Such a zero initial value is called the fair value of the contract. Pricing a contract hence amounts to finding a choice of pricing parameters that makes the contract value fair. The pricing parameters available are either the participation rate, the guaranteed rate participation, the guaranteed rate, or the cap rate. This article focusses on the participation rate.

## 5 Single Pass Lookahead Search Algorithm

The only step necessary to price an option by SPLS is to formulate the option pricing problem as a mathematical programming problem. We start with this formulation for a contract that offers both a reinvestment and a surrender option. Then, we briefly recall the SPLS algorithm.

To formulate the contract pricing problem as a mathematical programming problem, let  $X$  be the surrender decision process. The surrender decision process is a process that takes values in the set  $\{0, 1\}$ , and that takes the value 1 at time  $t$  only if the surrender option is exercised at time  $t$ . To ensure that the surrender option is only exercised once, the surrender decision process is constrained to sum to at most one. This constraint can be written with

$$\sum_{t=0}^n X_t \leq 1. \quad (5.1)$$

Since the reinvestment option can be exercised only if the contract is in force, it is necessary to track the event that the contract is in force. This event can be tracked with the complementary sum of the surrender decision process. This complementary sum can be written with

$$\bar{X}_{0:t} = 1 - \sum_{k=0}^{t-1} X_k. \quad (5.2)$$

The complementary sum is one when the contract is in force and zero at and after the exercise time of the surrender option. With these auxiliary variables, the stochastic control problem for the contract can be written with

$$\begin{aligned} & \max_{X,Y} \mathbb{E} \left( \sum_{t=1}^{T'} v^t \delta_t D_t X_t - \sum_{t=0}^{T'-1} v^t Y_t \bar{X}_{0:t} \right) \\ & \text{u.c. } \sum_{t=0}^n X_t \leq 1 \\ & X_t \in \{0, 1\}, \quad t = 0, 1, \dots, n \\ & Y_0 = y_0 \\ & Y_t \in [y_{min}, y_{max}] \quad t = 1, \dots, n \\ & T' = \min(n, T_x) \end{aligned} \quad (5.3)$$

and the SPLS algorithm can immediately be used to price the contract. When the contract is simpler, for example, when the contract is a single premium contract, or when the contract does not offer neither the premium option or surrender option, the stochastic control problem can be simplified. Table 5.1 presents the different cases. In particular, some cases reduce to a pure Monte Carlo simulation.

To recall the SPLS algorithm, it is easier to temporarily modify the notation and to consider the following general option pricing problem

$$\begin{aligned} & \max_X \mathbb{E} \left( \sum_{t=0}^n f(X_{0:t}, S_{0:t}) \right) \\ & \text{u.c. } g(X, S) = 0 \end{aligned} \quad (5.4)$$



where  $f$  is the option payoff at time  $t$ ,  $X$  is the exercise decision, and  $g$  is a constraint on the exercise decision. The payoff  $f$  depends on the previous exercise decision  $X_{0:t} = (X_k)_{k=0}^t$ , and the previous index price  $S_{0:t}$ . Given a sample path  $S$ , SPLS extracts an exercise strategy along the path by solving a Monte Carlo instance of the stochastic control problem (5.4) conditional on the filtration. In the Monte Carlo instance, the stochastic exercise strategy is turned into a deterministic strategy by approximating the exercise strategy with a step-function on a nearest-neighbor basis for the index price. The Monte Carlo instance is usually small, for example only 100 sample paths are used, and the resulting deterministic control problem is solved by linear programming. Specifically, the decision  $X_0$  at time 0 is extracted by solving a Monte Carlo instance of the control problem (5.4). The decision  $X_t$  at time  $t$  is extracted by solving a Monte Carlo instance of the following conditional control problem

$$\begin{aligned} \max_{X_{t:T}} \mathbb{E} \left( \sum_{s=t}^n f(X_{0:s}, S_{0:s}) \right) \\ \text{u.c. } g(X, S) = 0 \end{aligned} \tag{5.5}$$

where the objective and the constraint are conditional on the filtration. The filtration used here contains all the information generated by stock, and the previously extracted exercise decision  $X_{0:t-1}$ . The option price is then obtained by taking the average value of several such extracted strategies. The accuracy of the SPLS algorithm is hence characterized by three hyperparameters: the number of simulations used in the option price average value, the number of sample paths used in the Monte Carlo instantiations, and the size of the nearest-neighbor basis used to approximate the exercise strategy. Through a Vapnik-Chernovenkis analysis, it is possible to show that this algorithm converges uniformly to the option price. Uniform convergence means that the rate of convergence of the algorithm with respect to the hyperparameters is independent of both the payoff and the stochastic dynamic of the option. Through a metric called the filtration energy, it is also possible to show that the algorithm converges in probability to the option price. The key idea behind the filtration energy is to use an out-of-sample Monte Carlo simulation to measure how close in distribution the Monte Carlo instantiations are to the true control problem. See Essis-Breton and

Gaillardetz (2020) for more details.

Premium	Surrender Option	Stochastic Control Problem
single	no	$\mathbb{E} \left( v^{T'} \delta_{T'} D_{T'} - Y_0 \right)$
	yes	$\text{u.c. } Y_0 = y_0$ <hr/> $\max_X \mathbb{E} \left( \sum_{t=1}^{T'} v^t \delta_t D_t X_t - Y_0 \right)$ $\text{u.c. } \sum_{t=0}^n X_t \leq 1$ $X_t \in \{0, 1\} \quad t = 0, 1, \dots, n$ $Y_0 = y_0$
mandatory periodic	no	$\mathbb{E} \left( v^{T'} \delta_{T'} D_{T'} - \sum_{t=0}^{T'-1} v^t Y_t \right)$
	yes	$\text{u.c. } Y_t = y_0 \quad t = 0, 1, \dots, n$ <hr/> $\max_X \mathbb{E} \left( \sum_{t=1}^{T'} v^t \delta_t D_t X_t - \sum_{t=0}^{T'-1} v^t Y_t \bar{X}_{0:t} \right)$ $\text{u.c. } \sum_{t=0}^n X_t \leq 1$ $X_t \in \{0, 1\} \quad t = 0, 1, \dots, n$ $Y_t = y_0 \quad t = 0, 1, \dots, n$
reinvestment option	no	$\max_Y \mathbb{E} \left( v^{T'} \delta_{T'} D_{T'} - \sum_{t=0}^{T'-1} v^t Y_t \right)$
	yes	$\text{u.c. } Y_0 = y_0$ $Y_t \in [y_{min}, y_{max}] \quad t = 1, \dots, n$ <hr/> $\max_{X,Y} \mathbb{E} \left( \sum_{t=1}^{T'} v^t \delta_t D_t X_t - \sum_{t=0}^{T'-1} v^t Y_t \bar{X}_{0:t} \right)$ $\text{u.c. } \sum_{t=0}^n X_t \leq 1$ $X_t \in \{0, 1\} \quad t = 0, 1, \dots, n$ $Y_0 = y_0$ $Y_t \in [y_{min}, y_{max}] \quad t = 1, \dots, n$

Table 5.1: Stochastic control problem of different contracts

## 6 Numerical Examples

For the numerical examples, the initial stock price is  $S_0 = 1$ , the guaranteed rate participation  $\beta$  is 100%, the cap rate  $\zeta$  is  $\infty$ , The policyholder is 50 years old and the mortality follows the 1994 Variable Annuity MGDB Mortality Table (MGDB, 1994). The surrender penalty is a linear function defined by  $\delta_t = 1 - \max(5 - t, 0)\delta$ , and the penalty is applied only in case of early surrender. A contract sold without the surrender option is denoted with  $\delta = \infty$ . The risk-free rate  $r$  is 4% unless otherwise stated.

For the dynamic programming algorithm, the index dynamic is approximated by a binomial model with 6 trading dates, and the fair participation rate is found with the recursive formula (4.4). The dynamic programming algorithm converges quickly with the number of trading dates per year. In Table 6.1, all the fair participation rates presented differ by at most 2% to a fair participation rate computed with 2 trading dates per year (not included) and differ by at most 0.5% to a fair participation rate computed with 12 trading dates per year (not included). In the table, the fair participation rate is indicated in percent, and the dynamic programming fair participation rate is indicated by DP. For SPLS, the number of samples used to estimate the option price is 1,000, the number of samples in the Monte Carlo instantiation is 30, and the nearest-neighbor basis size is 15. With an out-of-sample validation of 1,000 paths, this choice of hyperparameters gives a maximum filtration energy of 0.2 for the different cases. As the contracts considered in the examples have a maximum maturity of 30 years, with the annual reinvestment exercise and surrender options, this choice of hyperparameters seems reasonable. Table 6.1 confirms this intuition as the SPLS estimates are very close to the dynamic programming estimates.

Table 6.2 presents the fair participation obtained by SPLS for contracts with long maturities. In both Table 6.1 and Table 6.2, the fair participation rate is found by finding the root of a spline fitted to a sample of participation rate (Press et al. 1992). For SPLS, for each sampled participation rate, the value of a participation rate is found with a single run of SPLS. The same random seed is used in each SPLS run. The computations are done in the Google Cloud (2020) on a 100 processors machine with 100 GB RAM, and with the linear programming solver Gurobi (2020). The nearest-neighbor of a path among a set of centroids is found with the multiple random projection technique

of Hyvonen et al. 2016. Depending on the premium option complexity, each run of SPLS takes between 1 second and 3 minutes, and obtaining a fair participation rate takes between 5 seconds and 30 minutes. The total computing time for Table 6.2 is 40 hours.

In Table 6.2, some SPLS estimates are not perfect. For example, the fair participation rate should increase with longer maturities, but the fair participation rate sometimes decreases when going from a maturity of 20 years to a maturity of 30 years. In numerical experiments, we found that this problem can be solved by using a higher number of simulations. In particular, when going from an estimate with 1,000 simulations to an estimate with 10,000 simulations, the difference in the fair participation rate is at most 5%. Further, there exists several ways to improve the accuracy of the SPLS estimates. For example, a multi-start approach can be used where estimates are computed from different random seeds, and the lowest fair participation rate found is retained. Another approach is hyperparameters tuning where the estimates are computed from different hyperparameters. The two previous approaches can also be mixed. See Essis-Breton and Gaillardetz (2020) for more details. The goal of this article is to shed light on the reinvestment options in EIAs, and to show that the fair participation for such EIAs can be estimated with an accuracy of 5%. We leave the quest for sharper estimates of the fair participation rate to future research.

## 7 Conclusions

With appropriate pricing algorithms, the risk associated to the reinvestment option in EIAs can be adequately priced and hedged. Such pricing algorithms are key to support innovative designs for EIAs and to help insurers distinguish themselves from alternative investment product manufacturers. The algorithms presented in this article provide such innovation support.

Premium	$\delta$	$g$	$\sigma$	Fund protection				Premium protection				
				Point-to-point		Annual reset		Point-to-point		Annual reset		
				DP	SPLS	DP	SPLS	DP	SPLS	DP	SPLS	
single	$\infty$	0.01	0.15	71.0	71.2	49.7	48.2	71.0	71.2	49.7	48.2	
			0.25	52.5	52.8	33.5	32.5	52.5	52.8	33.5	32.5	
			0.25 ( $r = 0.06$ )	69.8	69.9	46.3	44.9	69.8	69.9	46.3	44.9	
		0.03	0.15	49.9	49.8	42.8	41.9	49.9	49.8	42.8	41.9	
			0.25	33.5	33.3	28.0	27.5	33.5	33.3	28.0	27.5	
			0.25 ( $r = 0.06$ )	58.1	58.0	43.9	42.9	58.1	58.0	43.9	42.9	
		0.00	0.01	0.15	43.2	40.8	43.2	44.1	43.2	40.5	43.2	44.1
				0.25	28.9	26.8	28.9	28.8	28.9	27.3	28.9	28.8
				0.25 ( $r = 0.06$ )	42.2	39.6	42.2	43.0	42.2	40.5	42.2	43.0
	0.03	0.15	0.15	28.9	27.0	28.9	26.9	28.9	26.8	28.9	26.9	
			0.25	18.1	17.6	18.1	17.4	18.1	17.9	18.1	17.4	
			0.25 ( $r = 0.06$ )	32.9	32.1	32.9	31.8	32.9	31.9	32.9	31.8	
	0.01	0.01	0.15	66.8	68.2	49.6	49.9	66.8	72.5	49.6	49.9	
			0.25	47.4	50.9	33.5	34.5	47.4	48.1	33.5	34.5	
			0.25 ( $r = 0.06$ )	61.0	63.2	46.3	47.3	61.0	61.8	46.3	47.3	
		0.03	0.15	49.3	51.6	42.8	44.0	49.3	52.6	42.8	44.0	
			0.25	32.8	33.0	28.0	29.1	32.8	34.9	28.0	29.1	
			0.25 ( $r = 0.06$ )	51.9	55.3	43.6	44.6	51.9	56.9	43.6	44.6	
mandatory periodic	$\infty$	0.01	0.15	67.9	67.8	49.4	48.1	63.3	63.6	49.4	48.1	
			0.25	49.5	49.3	33.3	32.4	45.4	45.6	33.3	32.4	
			0.25 ( $r = 0.06$ )	66.4	66.3	46.2	44.9	62.1	62.3	46.2	44.9	
		0.03	0.15	0.15	47.7	47.3	41.6	40.7	43.7	43.6	41.6	40.7
				0.25	31.9	31.8	27.1	26.6	28.6	28.7	27.1	26.6
				0.25 ( $r = 0.06$ )	55.3	55.1	43.3	42.4	51.0	51.1	43.3	42.4
		0.00	0.01	0.15	43.2	41.8	43.2	44.0	43.2	42.5	43.2	44.0
				0.25	28.9	30.7	28.9	30.2	28.9	33.3	28.9	30.2
				0.25 ( $r = 0.06$ )	42.2	38.0	42.2	42.9	42.2	42.4	42.2	42.9
	0.03	0.15	0.15	28.9	29.2	28.9	29.7	28.9	28.3	28.9	29.7	
			0.25	18.1	17.8	18.1	21.3	18.1	19.8	18.1	21.3	
			0.25 ( $r = 0.06$ )	32.9	34.4	32.9	41.7	32.9	32.0	32.9	41.7	
	0.01	0.01	0.15	63.7	67.8	49.4	52.0	61.4	63.9	49.4	52.0	
			0.25	44.3	49.3	33.3	34.9	43.4	48.0	33.3	34.9	
			0.25 ( $r = 0.06$ )	58.4	62.2	46.2	48.0	57.9	63.8	46.2	48.0	
		0.03	0.15	0.15	46.2	50.0	41.6	43.8	43.5	47.7	41.6	43.8
				0.25	30.5	32.9	26.9	27.2	28.4	30.0	26.9	27.2
				0.25 ( $r = 0.06$ )	49.3	53.5	43.2	46.0	48.5	53.9	43.2	46.0
reinvestment option	$\infty$	0.01	0.15	65.7	67.1	49.0	47.3	62.1	64.4	49.0	47.3	
			0.25	47.5	47.7	32.9	32.2	44.2	44.6	32.9	32.2	
			0.25 ( $r = 0.06$ )	64.3	65.4	45.9	44.9	60.9	62.2	45.9	44.9	
		0.03	0.15	0.15	45.7	45.8	40.7	39.3	42.5	42.7	40.7	39.3
				0.25	30.3	30.6	26.3	25.5	27.9	28.7	26.3	25.5
				0.25 ( $r = 0.06$ )	53.2	54.5	42.7	41.8	49.7	50.8	42.7	41.8
		0.00	0.01	0.15	43.2	44.3	43.2	45.6	43.2	49.3	43.2	45.6
				0.25	28.9	34.3	28.9	33.4	28.9	28.8	28.9	33.4
				0.25 ( $r = 0.06$ )	42.2	47.2	42.2	43.4	42.2	48.7	42.2	43.4
	0.03	0.15	0.15	28.9	27.2	28.9	28.6	28.9	28.2	28.9	28.6	
			0.25	18.1	17.1	18.1	21.6	18.1	20.7	18.1	21.6	
			0.25 ( $r = 0.06$ )	32.9	32.1	32.9	40.4	32.9	31.6	32.9	40.4	
	0.01	0.01	0.15	62.5	65.6	49.0	49.0	60.4	65.8	49.0	49.0	
			0.25	43.7	46.6	32.9	33.5	42.6	47.7	32.9	33.5	
			0.25 ( $r = 0.06$ )	58.1	64.0	45.9	46.3	57.2	62.3	45.9	46.3	
		0.03	0.15	0.15	45.1	48.7	40.7	41.7	42.4	44.7	40.7	41.7
				0.25	29.7	32.9	26.3	26.3	27.6	28.9	26.3	26.3
				0.25 ( $r = 0.06$ )	48.8	51.7	42.6	43.8	48.0	52.0	42.6	43.8

Table 6.1: Comparison of the dynamic programming algorithm and the SPLS algorithm for contract of 5 years.

Premium	$\delta$	$g$	$\sigma$	$n =$	Fund protection						Premium protection					
					Point-to-point			Annual Reset			Point-to-point			Annual Reset		
					10	20	30	10	20	30	10	20	30	10	20	30
single	$\infty$	0.01	0.15	78.7	93.1	83.6	48.4	48.9	48.0	78.7	93.1	83.6	48.4	48.9	48.0	
			0.25	60.8	83.3	62.7	32.7	33.2	32.5	60.8	83.3	62.7	32.7	33.2	32.5	
			0.25 ( $r = 0.06$ )	76.8	97.1	73.3	44.9	45.5	44.5	76.8	97.1	73.3	44.9	45.5	44.5	
		0.03	0.15	57.1	70.9	65.0	45.0	47.5	47.2	57.1	70.9	65.0	45.0	47.5	47.2	
			0.25	38.9	53.9	40.3	29.9	31.9	31.7	38.9	53.9	40.3	29.9	31.9	31.7	
			0.25 ( $r = 0.06$ )	65.2	85.5	65.5	44.1	45.3	44.4	65.2	85.5	65.5	44.1	45.3	44.4	
	0.00	0.01	0.15	41.9	43.1	45.0	43.4	44.0	46.7	42.0	43.7	43.9	43.8	43.1	45.9	
			0.25	27.2	27.6	31.3	31.7	31.6	31.6	28.5	27.4	31.7	31.0	31.8	31.4	
			0.25 ( $r = 0.06$ )	41.8	42.7	48.6	41.8	43.0	44.4	40.1	42.9	48.3	43.8	41.6	43.9	
		0.03	0.15	27.8	27.7	29.8	28.5	27.4	29.4	28.2	27.5	29.1	28.3	28.0	28.6	
			0.25	19.1	17.6	18.9	17.3	17.1	18.0	18.5	17.4	19.2	17.7	17.1	17.9	
			0.25 ( $r = 0.06$ )	33.4	33.3	36.5	32.6	32.4	41.0	34.4	33.5	36.3	33.1	31.7	41.0	
	0.01	0.01	0.15	64.9	67.6	71.7	49.7	49.5	49.9	69.7	68.8	68.9	49.8	49.4	50.4	
			0.25	44.8	48.9	48.9	33.8	33.2	34.0	47.5	48.0	50.3	33.7	33.2	33.8	
			0.25 ( $r = 0.06$ )	60.5	61.6	64.4	45.8	45.8	46.2	58.2	60.8	64.3	45.9	46.0	46.0	
		0.03	0.15	51.3	50.9	52.1	45.1	43.9	45.2	51.9	51.0	52.7	45.1	43.8	45.2	
			0.25	33.1	34.8	34.7	29.2	28.9	29.8	34.1	33.3	36.8	29.8	29.2	29.8	
			0.25 ( $r = 0.06$ )	49.9	53.7	55.7	43.3	43.4	44.9	51.2	55.0	53.5	43.8	43.1	45.1	
mandatory periodic	$\infty$	0.01	0.15	76.8	89.7	83.6	48.8	49.1	48.0	72.5	86.2	80.9	48.4	49.0	48.0	
			0.25	57.8	76.7	63.7	33.0	33.3	32.5	53.8	71.7	60.8	32.7	33.3	32.4	
			0.25 ( $r = 0.06$ )	74.6	92.0	75.5	45.2	45.7	44.4	70.6	88.4	73.5	45.0	45.6	44.4	
		0.03	0.15	55.2	67.9	64.5	44.2	47.3	46.9	51.1	63.5	60.7	43.0	46.5	46.4	
			0.25	37.3	50.8	41.4	29.2	31.7	31.4	33.8	46.1	38.5	28.3	31.1	31.0	
			0.25 ( $r = 0.06$ )	63.0	80.6	66.9	44.1	45.5	44.3	59.0	76.1	64.3	43.4	45.2	44.1	
	0.00	0.01	0.15	44.4	47.3	43.7	44.2	37.7	45.6	48.4	54.3	54.3	43.6	48.8	44.1	
			0.25	29.6	34.0	35.8	32.9	32.6	16.1	27.0	32.9	41.0	31.5	31.6	16.2	
			0.25 ( $r = 0.06$ )	40.5	52.9	55.0	44.8	42.8	43.3	40.4	44.7	48.4	43.2	43.9	44.0	
		0.03	0.15	31.8	30.9	27.7	31.5	31.3	28.9	28.5	29.8	32.2	29.6	31.3	29.1	
			0.25	20.0	17.7	21.9	16.6	14.8	14.9	18.3	20.7	20.3	20.4	19.7	19.7	
			0.25 ( $r = 0.06$ )	38.9	34.2	37.3	29.9	28.2	27.2	33.2	35.5	38.1	40.9	41.6	29.5	
	0.01	0.01	0.15	65.0	64.1	65.6	48.9	50.2	48.5	65.9	64.9	66.2	48.3	49.3	48.8	
			0.25	45.8	45.1	47.7	33.0	33.4	33.5	46.8	46.8	46.3	33.5	33.3	32.3	
			0.25 ( $r = 0.06$ )	65.1	62.3	59.7	46.1	45.7	45.7	59.6	54.4	54.3	44.7	45.9	45.2	
		0.03	0.15	46.1	48.9	47.5	42.7	43.2	42.5	46.4	46.1	47.2	41.7	41.1	42.4	
			0.25	31.9	30.7	33.3	27.2	28.8	29.5	30.8	29.6	29.7	25.9	27.6	28.5	
			0.25 ( $r = 0.06$ )	51.7	53.9	50.3	45.0	43.3	44.1	48.4	51.6	48.6	42.3	42.2	43.0	
reinvestment option	$\infty$	0.01	0.15	75.9	88.4	83.1	47.6	46.6	46.0	71.6	84.4	81.8	47.2	46.6	46.0	
			0.25	57.0	75.2	63.0	33.0	33.5	32.3	51.9	68.4	57.8	32.7	33.2	32.1	
			0.25 ( $r = 0.06$ )	73.4	91.0	79.3	45.1	45.5	44.4	69.0	86.6	73.8	44.7	45.3	44.3	
		0.03	0.15	53.4	66.2	63.4	44.1	45.8	45.8	50.0	60.3	59.3	42.2	45.3	45.5	
			0.25	36.4	48.2	40.0	27.3	30.9	30.4	33.2	43.3	38.9	26.2	29.8	29.6	
			0.25 ( $r = 0.06$ )	61.5	78.5	67.6	44.4	45.1	44.3	58.8	73.5	63.8	43.2	44.9	44.3	
	0.00	0.01	0.15	46.4	40.9	43.7	46.1	41.4	43.7	48.6	49.6	44.1	45.2	45.5	44.0	
			0.25	31.9	27.2	29.5	32.1	32.4	32.3	31.8	28.5	29.7	32.5	16.9	18.2	
			0.25 ( $r = 0.06$ )	45.9	39.7	41.6	45.1	44.2	44.2	47.1	51.0	45.5	42.6	43.8	44.7	
		0.03	0.15	29.7	27.7	30.5	30.7	27.8	30.5	28.3	30.0	31.4	30.2	30.7	30.3	
			0.25	19.0	18.0	20.2	20.6	17.9	19.2	21.6	20.5	19.5	20.0	21.4	20.8	
			0.25 ( $r = 0.06$ )	35.6	31.8	35.8	39.4	31.7	41.5	42.8	36.0	36.3	41.1	41.5	29.8	
	0.01	0.01	0.15	68.3	78.3	82.5	48.5	51.2	53.6	67.0	62.6	67.4	48.1	48.3	51.5	
			0.25	50.6	50.6	55.6	33.4	34.9	36.5	46.2	43.9	47.0	32.6	32.6	33.9	
			0.25 ( $r = 0.06$ )	64.3	66.5	69.9	45.6	46.4	47.5	61.5	60.6	61.7	45.8	45.2	44.9	
		0.03	0.15	54.3	58.3	74.8	45.1	51.1	56.2	44.4	46.2	49.0	39.8	41.2	44.1	
			0.25	32.6	37.4	50.2	29.2	33.1	39.7	29.1	30.4	31.5	25.5	27.3	30.5	
			0.25 ( $r = 0.06$ )	54.4	59.4	62.1	44.1	47.3	50.7	50.8	48.6	53.8	42.7	42.1	44.0	

Table 6.2: SPLS fair participation rate for contract with long maturity  $n$ .

## References

- [1] Andreatta, G. and Corradin S. (2003) Valuing the Surrender Options Embedded in a Portfolio of Italian Life Guaranteed Participating Policies: a Least Squares Monte Carlo Approach. *Real option theory meets practice, 8th Annual International Conference 2004*, Montreal, Canada.
- [2] Bacinello, A.R. (2003a) Fair Valuation of a Guaranteed Life Insurance Participating Contract Embedding a Surrender Option. *Journal of Risk and Insurance* 70(3): 461-487.
- [3] Bacinello, A.R. (2003b) Pricing Guaranteed Life Insurance Participating Policies with Annual Premiums, and Surrenders Option. *North American Actuarial Journal* 7(3): 1-17.
- [4] Bacinello, A.R. (2005) Endogenous Model of Surrender Conditions in Equity-Linked Life Insurance, *Insurance: Mathematics and Economics* 37(2): 270-293.
- [5] Bertsekas, D.P.(2005) *Dynamic programming and optimal control* vol. 1, 3rd ed., Athena Scientific, Belmont, Massachusetts.
- [6] Bowers, N.L. Jr., Gerber, H.U., Hickman, J.C., Jones, D.A. and Nesbitt, C. (1997) *Actuarial Mathematics*, Second Edition. Society of Actuaries, Schaumburg, Illinois.
- [7] Black, F. and Scholes, M.S. (1973) The Pricing of Options and Corporate Liabilities. *Journal of Political Economy* 81(May-June): 637-654.
- [8] Boyle, P. and E. S. Schwartz (1977) Equilibrium prices of guarantees under equity-linked contracts. *Journal of Risk and Insurance* 44(4): 639-660.
- [9] Brennan, M. J. and E. S. Schwartz (1976) The pricing of equity-linked life insurance policies with an asset value guarantee. *Journal of Financial Economics* 3: 195-213.
- [10] Carriere, J.F. (1996) Valuation of the early-exercise price for options using simulations and nonparametric regression. *Insurance: mathematics and Economics* 19(1): 19-30.
- [11] Chi, Y. and Lin, X. S. (2012) Are Flexible Premium Variable Annuities Under-Priced? *Astin Bulletin* 42(02): 559-574.



- [12] Google Cloud (2020) Google Cloud Platform. <http://cloud.google.com>
- [13] Costabile, M., Massabo, I. and Russo, E. (2008) A binomial model for valuing equity-linked policies embedding surrender options. *Insurance: Mathematics and Economics* In Press.
- [14] Cox, J., Ross, S.A. and Rubinstein, M. (1979) Option Pricing: A Simplified Approach. *Journal of Financial Economics* 7(3): 229-263.
- [15] Essis-Breton, N. and Gaillardetz, P. (2020) Fast Lower and Upper Estimates for the Price of Constrained Multiple Exercise American Options by Single Pass Lookahead Search and Nearest-Neighbor Martingale. *Working Paper*.
- [16] Gaillardetz, P. and Lakhmiri, Y.J. (2009) New Premium Principle for Equity-Indexed Annuities. Accepted for publication in the *Journal of Risk and Insurance*.
- [17] Gurobi (2020) Gurobi Optimizer Reference Manual. <http://www.gurobi.com>.
- [18] Hardy, M.R. (2003) *Investment Guarantees: Modeling and Risk Management for Equity-Linked life Insurance*. Wiley & Sons, Inc., New Jersey.
- [19] Hyvönen, V. et al. (2016) Fast nearest neighbor search through sparse random projections and voting. *Big Data (Big Data), 2016 IEEE International Conference on. IEEE*: 881–888.
- [20] Insured Retirement Institute (2019) *IRI Issues 2018 Fourth Quarter Annuity Sales Report*. <http://www.irionline.org/>.
- [21] Lee, H. (2003) Pricing equity-indexed annuities with path-dependent options. *Insurance: Mathematics and Economics* 33(3): 677-690.
- [22] Lin, X.S. and Tan, K.S. (2003) Valuation of Equity-Indexed Annuities under Stochastic Interest Rate. *North American Actuarial Journal* 7(3): 72-91.
- [23] Longstaff, F.A. and Schwartz, E.S. (2001) Valuing American options by simulation: a simple least-squares approach. *The Review of Financial Studies* 14(1): 113-147.

- [24] MGDB (1994) *1994 Variable Annuity Minimum Guarantee Death Benefit (MGDB) - Male. Basis: Age Nearest Birthday*. <http://mort.soa.org/ViewTable.aspx?&TableIdentity=881>.
- [25] Merton, R.C. (1973) The Theory of Rational Option Pricing, *The Bell Journal of Economics and Management Science* 4 (Spring): 141-183
- [26] Møller, T. (1998). Risk-Minimizing Hedging Strategies for Unit-Linked Life Insurance Contracts. *Astin Bulletin* 28(1): 17-47.
- [27] Møller, T. and Steffensen, M. (2007) *Market-Valuation Methods in Life and Pension Insurance*. Cambridge University Press, Cambridge.
- [28] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T (1992) *Numerical recipes in C: the art of scientific computing* 2nd edition, Cambridge University Press.
- [29] Tiong, S. (2000) Valuing Equity-indexed Annuities. *North American Actuarial Journal* 4(4): 149-163.
- [30] Wei X., Gaudenzi M., Zanette A. (2013) Pricing Ratchet equity-indexed annuities with early surrender risk in a CIR++ model. *North American Actuarial Journal* 17(3): 229-252.
- [31] Yuen, F.L. and Yang, H. (2010) Pricing Asian options and equity-indexed annuities with regime switching by the trinomial tree method. *North American Actuarial Journal* 14(2): 256-272.
- [32] Young, V.R. (2006) *Premium Principles* Encyclopedia of Actuarial Science, Wiley.

# Chapter 4

## Conclusion and Future Work

The passage from a stochastic control problem to its computer solution is an important problem that arises in many fields. In finance, in particular in option pricing, an efficient solution to this algorithmic translation problem facilitates the risk-management of any derivative and helps in the negotiation of complex derivative contracts. In this thesis, a new solution to this algorithmic translation problem was presented, and the solution meets the efficiency needed in option pricing practice. The solution consists in highlighting a family of exercise strategy and a family of dual martingale that are tractable and that exhibit good convergence guarantees. The associated algorithms are called single pass lookahead search and nearest-neighbor martingale.

They are 3 broad possible directions for future work. A first direction for future work is the application of the algorithms to other areas in finance. Possible areas are optimal asset allocation, real option pricing, and high-frequency trading. These different areas usually relies on dynamic programming and the proposed algorithms can widen the complexity of problems handled within these areas.

A second direction for future work is to find application of the proposed algorithms in other fields than finance. In particular, the algorithms can be used to find approximate solution to multi-stage stochastic programs. Within this perspective, the algorithms can be embedded directly in existing linear programming solvers, so that the solvers handle multi-stage stochastic program out-of-the-

box. Currently available algorithmic solution to multi-stage stochastic program usually relies on scenarios tree or genetic optimization and can hardly handle complex program or program with long maturity.

A third direction for future work is a study of the computational complexity of the proposed algorithms. In the thesis, probabilistic convergence proofs are provided for the speed of convergence to the best possible solution. However, these convergence rates do not quantify the computational effort behind the algorithms. In the numerical examples, the thesis demonstrates that the algorithms scale to very complex options, that are intractable for dynamic programming. A proof that the computational effort behind the algorithms grows for example polynomially with the complexity of the option will confirm the efficiency observed empirically.

# References

- Black, F. and M. Scholes (1973) The pricing of options and corporate liabilities. *The journal of political economy*, 637–654.
- Merton, R. C. (1973) Theory of rational option pricing. *The Bell Journal of economics and management science*, 141–183.