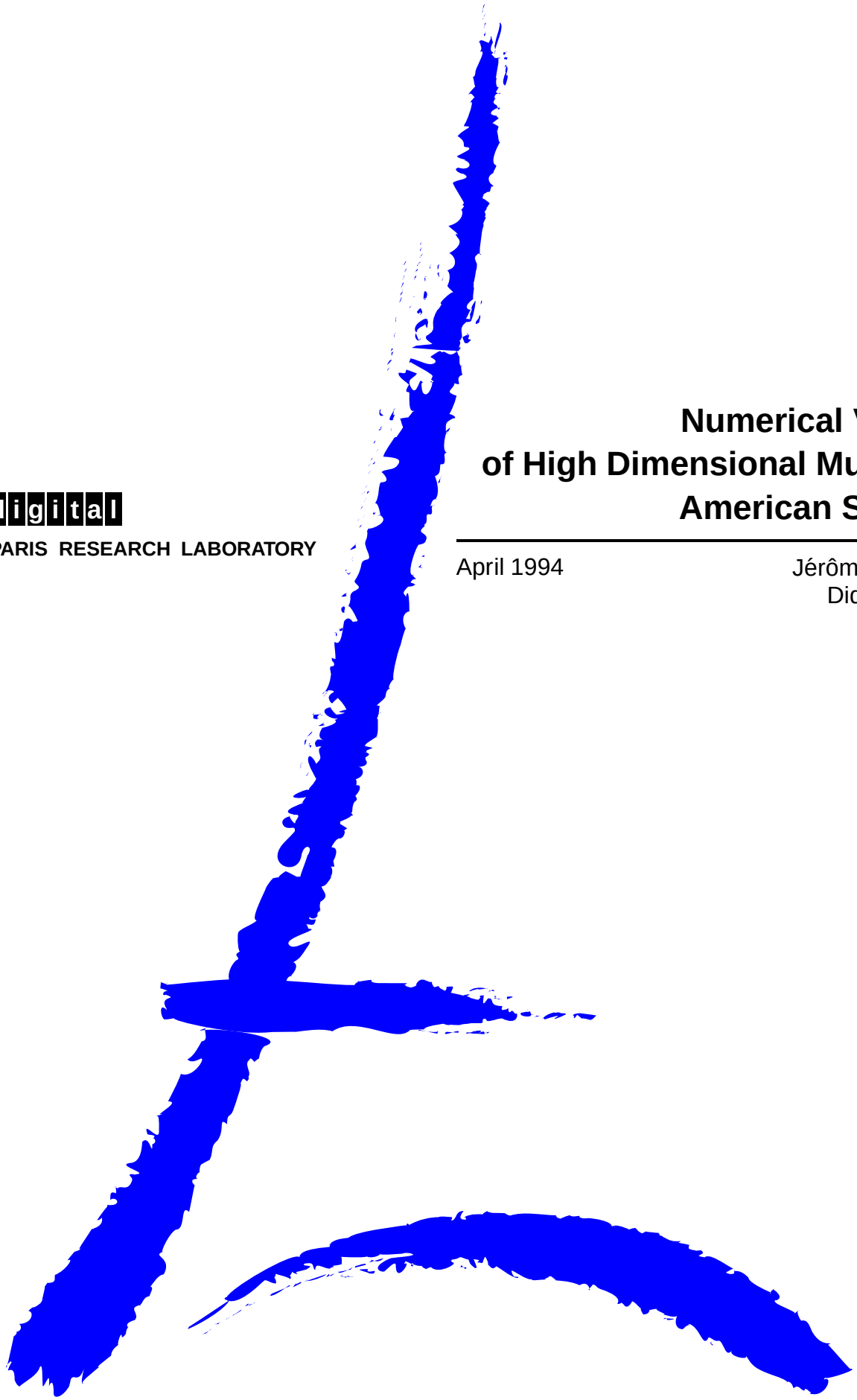

38

Numerical Valuation of High Dimensional Multivariate American Securities

digital
PARIS RESEARCH LABORATORY

April 1994

Jérôme Barraquand
Didier Martineau

38

Numerical Valuation of High Dimensional Multivariate American Securities

Jérôme Barraquand
Didier Martineau

April 1994

Publication Notes

The authors can be contacted at the following addresses:

Jerome Barraquand
Digital Equipment Corporation
Paris Research Laboratory
85 Avenue Victor Hugo
92500 Rueil-Malmaison, France
barraquand@prl.dec.com

Didier Martineau
Digital Equipment Corporation
Paris Research Laboratory
85 Avenue Victor Hugo
92500 Rueil-Malmaison, France
martino@prl.dec.com

Abstract

We consider the problem of pricing an American contingent claim whose payoff depends on several sources of uncertainty. Using classical assumptions from the Arbitrage Pricing Theory, the theoretical price can be computed as the maximum over all possible early exercise strategies of the discounted expected cash flows under the modified risk-neutral information process.

Several efficient numerical techniques exist for pricing American securities depending on one or few (up to 3) risk sources. They are either lattice-based techniques or finite difference approximations of the Black-Scholes diffusion equation. However, these methods cannot be used for high-dimensional problems, since their memory requirement is exponential in the number of risk sources.

In this paper, we present an efficient numerical technique that combines Monte Carlo simulation with a particular partitioning method of the underlying assets space, which we call *Stratified State Aggregation* (SSA). Using this technique we can compute accurate approximations of prices of American securities with an arbitrary number of underlying assets. Our numerical experiments show that the method is practical for pricing American claims depending on up to 400 risk sources. On all problems for which we could compare the method with known optimal solutions, the price computed through stratified state aggregation was indistinguishable from the optimal theoretical price. Several numerical examples are presented and discussed.

Acknowledgements

This research benefited from discussions with Bruno Langlois. We wish to thank Thierry Pudet for reviewing an earlier draft.

Contents

1 Introduction 1

2 Relation to other work 4

3 Arbitrage pricing theory 5
- 3.1 Diffusion model of information process . . . 5
- 3.2 European securities . . . 5
- 3.3 Arbitrage pricing . . . 6
- 3.4 American securities . . . 7
- 3.5 European and American options . . . 8
  - *3.5.1 European options* . . . 8
  - *3.5.2 American options* . . . 10

4 Numerical methods for American security pricing 11
- 4.1 Stochastic dynamic programming . . . 11
- 4.2 Finite differences and the Cox-Ross-Rubinstein approach . . . 12

5 State aggregation 14
- 5.1 State aggregation price . . . 14
- 5.2 Markovian approximation . . . 15
- 5.3 Recursive state aggregation . . . 16
- 5.4 Stratified state aggregation (SSA) . . . 17

6 Monte Carlo estimation of American price 19
- 6.1 Generation of sample paths . . . 19
- 6.2 Conditional probabilities and payoff expectations . . . 19
- 6.3 Backward integration algorithm . . . 20

7 Quadratic resampling 20
- 7.1 Quadratic resampling for multidimensional Monte Carlo integration . . . 20
- 7.2 Quadratic resampling in space$^{\text{time}}$ . . . 22

8 Experimental results 22
- 8.1 A test case . . . 22
- 8.2 One underlying asset . . . 24
- 8.3 Two underlying assets . . . 24
- 8.4 Three underlying assets . . . 29
- 8.5 Ten underlying assets . . . 29
- 8.6 Efficiency of extended quadratic resampling . . . 32
- 8.7 Experimental time complexity . . . 32
- 8.8 Parallel implementation . . . 33

9 Conclusion 36

1 Introduction

Since the seminal work of Black and Scholes (1973) and Merton (1973) in the early 1970s, the arbitrage principle underlying option valuation theory has been extended to a broad range of other financial instruments (see *e.g.* Ross (1976), Cox and Rubinstein (1985)). Indeed, any security whose returns are contractually related to the returns on some other security or group of securities can theoretically be valuated using the same arbitrage principle. In some cases, explicit closed form analytical formulas for the computation of the arbitrage price can be derived from this theory. In particular, the original paper of Black and Scholes (1973) provides a closed form solution for a European option on a single common stock. Unfortunately, few other cases can be solved analytically, and computing the arbitrage price often requires numerical simulations. Following an idea initially presented in an early edition of Sharpe (1985), Cox et al. (1979) developed a discrete model for the valuation of an American option on a single stock that can be easily computed numerically. However, the effective implementation of the arbitrage principle is not always such an easy task, and may sometimes become intractable. Tractable algorithms have been developed recently for pricing European contingent claims with many underlying assets (see e.g. Barraquand (1993)). However, these algorithms cannot be used for pricing American contingent claims.

There are several reasons motivating the development of efficient methods for multidimensional contingent claim pricing. In particular, applications exist in the pricing of Over The Counter (OTC) warrants, path dependent instruments (Barraquand and Pudet (1994)), multidimensional interest rate term structure contingent claims (Heath et al. (1992)) such as mortgage-backed securities, and life insurance policies (Fabozzi and Pollack (1987)), futures contracts with quality delivery options (Cheng (1987); Boyle (1989)). Also, pricing models taking into account the stochastic nature of volatility (Wiggins (1987); Dothan (1987); Hull and White (1988)) require multidimensional modeling. Other applications exist in assets and liabilities management, and in corporate capital budgeting (see e.g. Mason and Merton (1985); Coppeland (1989); Brealey and Myers (1991)). Finally, applications exist in property/liability insurance (Merton (1977); Smith (1979); Kraus and Ross (1982); Doherty and Garven (1986); Cummins (1988); Shimko (1992)). The tremendous development of financial engineering during the past decade can be expected to continue, and new types of securities requiring multidimensional modeling are likely to appear at a sustained pace in the future.

For pricing purposes, financial assets can be divided into two majors classes. The first class is that of assets whose future cash-flows cannot be influenced by decisions from the holder taken after the purchase date. We will call European instruments all financial assets belonging to this first class. In particular, stocks, bonds, futures contracts, European options, swaps, caps, floors, mortgage-backed securities are European instruments. The second class is that of assets whose cash flows can be influenced a posteriori by the holder. American options belong to this class. As another example, the crediting policy of a Life Insurance company selling SPDAs greatly influences the present value of the liabilities of the company. This crediting policy can be adjusted by the company after signature of the contracts with the policyholders. Therefore, the liability associated with the sale of SPDA can be viewed as an American security. We will

call American instruments all financial assets belonging to this second class.

Following the general theory of arbitrage pricing, the theoretical price of a European contingent claim is the discounted expected value of its future cash flows under the so-called "risk-neutral" probability distribution of the underlying economic factors (Harrison and Kreps (1979); Harrison and Pliska (1981); Duffie (1988); Karatzas and Shreve (1988)). Mathematically, computing the arbitrage price reduces to computing an integral (sum) over the space of the underlying economic factors. When the dimension of the space of the underlying economic factors is small, standard techniques for numerical integration can be used. In some cases, the integral can even be computed analytically (*e.g.* Black-Scholes formula). However, the computational complexity of evaluating the integral is clearly exponential in the dimension of the space. Efficient numerical techniques for pricing high-dimensional European claims are presented in Barraquand (1993).

The price of an American claim is the maximum over all possible cash flow monitoring strategies of the associated present values of cash flows. For example, the price of an American option is the maximum over all possible early exercise strategies of the corresponding present values. Since the space of cash flow monitoring strategies is generally huge, direct maximization of the present value is rarely practical (see Bossaerts (1989) for a discussion). However, when the underlying economy is modeled as a Markov process, one can use the Bellman principle of dynamic programming (Bellman (1957)) to compute the optimal monitoring strategy. American options are typically priced using a discrete approximation of the dynamic programming principle. This is the case in particular of the CRR model (Cox et al. (1979)) for American stock option pricing. This approach becomes however impractical when the underlying economic space has many dimensions, since the dynamic programming algorithm requires a memory space exponential in the number of dimensions. This fact is known as the "curse of dimensionality" problem for dynamic programming.

In this paper, we present a particular state space partitioning technique that attempts to circumvent the curse of dimensionality problem for American security pricing. More precisely, we partition the space of underlying assets (the state space) into a tractable number of cells, and we compute an approximate early exercise strategy that is constant over those cells. The hope is that, if the partition is appropriately chosen, the approximate strategy will be close to the actual optimal strategy. Such a partitioning technique is classically called a *state aggregation* technique.

Among the many possible ways of choosing a partition, one solution is to fix a particular real-valued function mapping the state (i.e. the prices of the underlying assets) that particularly influences the optimal strategy in the problem at hand. We call this function stratification map. Then, the partition chosen is a stratification of the state space into thin layers along this map. In other words, we limit our search to strategies that only depend upon the stratification map, and not upon the entire state itself. We call this technique *Stratified State Aggregation*.

In the case of American security pricing, an obvious candidate for the stratification map is the *payoff* of the security, i.e. the function representing the future cash-flows associated with the security. When the stratification map chosen is the payoff of the American security, we call

the technique *Stratified State Aggregation along the Payoff* (SSAP).

After quantization of the payoff, the SSAP method can be combined with Monte Carlo simulation techniques in order to compute the set of conditional probabilities corresponding to changes in the payoff value over time. Using these conditional probabilities, an approximation of the American price can then be computed backwards in time using techniques reminiscent from the classical CRR integration method.

We implemented the SSAP method on American option pricing problems in dimensions ranging from 1 to 400. On all problems for which we could compare the SSAP method with known optimal solutions, the SSAP price was indistinguishable from the optimal theoretical price. In particular, in dimensions 1, 2, and 3, both put and call prices of options on the maximum of the underlying assets were computed accurately by the SSAP method. Also, the SSAP price of an American call on the maximum of n assets paying no dividends was indistinguishable from the European price for n ranging from 1 to 400, in accordance with a well known theoretical result[1]. In other cases, no other method exists to compare to our results. However, the SSAP price seems to constitute an accurate approximation of the American price in arbitrarily high dimensions.

To the best of our knowledge, the SSAP method is the first capable of computing American prices and exercise strategies in high dimensional cases.

In order to speed-up the Monte Carlo simulation of conditional probabilities, we developed an original variance reduction technique called Quadratic Resampling. Quadratic Resampling was originally presented in Barraquand (1993) for European security pricing. In this paper, we present an extension of the original QR method that applies to both European and American asset pricing problems. Quadratic Resampling consists in correcting the samples obtained through classical Monte Carlo simulation in such a way that the expected value of any polynomial of degree two or less in the space variables is computed exactly. Our experiments show that QR is very efficient for American pricing problems in up to 10 dimensions. The average speedup obtained through QR ranges from 5 to 35, with an average of about 10. In higher dimensions (11 and higher), the speedup is only of 2 to 3 in average, and slowly decreases with the dimension.

We implemented a parallel version of the SSAP method on a network of workstations equipped with a high-bandwidth interconnect (called a *workstation farm*). We observed a speedup linear in the number of workstations in the network. Both measured and simulated parallelization experiments are reported in this paper.

This paper is organized as follows. In Section 2, we relate our contribution to previous work in American security pricing, multidimensional asset pricing, and Monte Carlo valuation. In Section 3, we recall the usual assumptions on the stochastic processes governing the evolution of securities prices, and the main results of the Arbitrage Pricing Theory. In Section 4, we

[1] Indeed, since the discounted prices of assets paying no dividends are martingales under the risk neutral measure, the discounted maximum of such n assets prices is a submartingale. Hence, the corresponding European call price is always higher than the immediate payoff.

briefly review the current numerical methods used for American security pricing. In Section 5, we present the method of Stratified State Aggregation. In Section 6, we show how SSA prices can be computed through Monte Carlo simulation. In Section 7, we present the method of Quadratic Resampling. In Section 8, we present numerical experiments illustrating the efficiency of the SSAP method.

2 Relation to other work

The theoretical analysis of optimal stopping times for early exercise of American options dates back to the work of McKean (1965). This theory has then been further developed by several authors (Merton (1973); Harrison and Kreps (1979); Bensoussan (1984); Karatzas (1988); Jaillet et al. (1988)). Myneni (1992) surveys the theory of American option pricing.

The most widely used valuation technique for American options with one underlying asset is the binomial lattice approach of Cox et al. (1979). Cox and Rubinstein (1985) outline the principle of the multidimensional extension. Other numerical valuation techniques are presented in Geske and Johnson (1984), Barone-Adesi and Whaley (1987), Barone-Adesi and Elliott (1991).

The valuation of options depending of several underlying assets has been extensively studied. Brennan and Schwartz (1979) addresses the problem of pricing options under two sources of risk by direct finite-difference approximation of the generalized Black-Scholes equation. In this example the two sources of risk are the short term and the long term interest rates. The approach is clearly limited to a few assets, since the memory space requirements and the computation time are both exponential in the number of underlying assets. Boyle et al. (1989) developed a multinomial lattice method for pricing multidimensional options, in the spirit of the approach outlined in Cox and Rubinstein (1985). According to the authors, the computation becomes very burdensome for more than two assets. In fact, the multinomial lattice approach can be viewed as a finite-difference approximation of the generalized Black-Scholes equation using an explicit Euler scheme and an appropriate change of variables (Brennan and Schwartz (1978)).

Stulz (1982) presents an analytical solution to the problem of pricing a European option on the maximum or minimum of two underlying assets. The analytical solution is generalized in Johnson (1987) to the case of an arbitrary number of assets, taking as given the cumulative multivariate normal distribution function. Boyle (1989) and Boyle and Tse (1990) developed an approximate method for the same problem. Although the problem is solved analytically in Johnson (1987), the approximate method does not require preliminary computation of the cumulative multivariate normal distribution function. To the best of our knowledge, no closed form solutions have ever been obtained for American pricing problems.

Good reviews of the Monte Carlo method and different variance reduction techniques such as antithetic variables, covariates, stratified sampling, importance sampling can be found in many sources such as Hammersley and Hanscomb (1964), Zaremba (1968), Haber (1970), Kalos and Whitlock (1986), and references thereof.

The application of the Monte Carlo method to option pricing was first presented in Boyle (1977), in the context of claims contingent to a single underlying asset. It has then been used by several authors for the valuation of path dependent contingent claims. In particular, the method has been used for pricing mortgage-backed securities (see Schwartz and Torous (1989), Hutchinson and Zenios (1991)). Barraquand (1993) presents the method of Quadratic Resampling for Monte Carlo valuation of European securities with many underlying assets. The Quadratic Resampling method presented in this paper is an extension of this earlier work to the American pricing problem.

3 Arbitrage pricing theory

The arbitrage pricing theory is described in many textbooks. We refer the reader to Duffie (1992) for a comprehensive and rigorous presentation. In the following, we briefly present through intuitive arguments the main results of the theory. These developments do not constitute mathematical proofs, and are only aimed at illustrating the main concepts underlying the computational approaches to arbitrage pricing.

3.1 Diffusion model of information process

We model the economy as a finite-dimensional vector of real-valued state variables $X(t) = (x_1(t), \ldots, x_n(t))$, called *factors*, representing all the information available to investors at time t. Since $X(t)$ represents all information available to agents at time t, in a frictionless market, prices of securities must be deterministic functions of time and $X(t)$. It is said that securities are contingent claims on the state variable $X(t)$.

For the sake of simplicity, we will assume that the information process $X(t)$ is a *diffusion process*. However, our results on security valuation described in the next sections apply to more general types of stochastic processes. If $X(t)$ is a diffusion process, it is a solution of a stochastic differential equation of the type (Itô and McKean (1965)):

$$dX = A(X(t), t)dt + B(X(t), t)dW \tag{3.1}$$

The vector A is called the drift of process X. A is the derivative of the expected value of X. The matrix $\Gamma = BB^T$ is the derivative of the covariance of X. The vector $W = (w_1, \ldots, w_n)$ is a n-dimensional standard Brownian motion.

Often, the variables x_i are prices of securities available on the market, and are therefore strictly positive processes. The expected increments and covariance of increments are then expressed in relative values:

$$\forall i \in [1, n], \qquad \frac{dx_i}{x_i} = \mu_{x_i}(x_1, \ldots, x_n, t)dt + \sum_{j=1}^{n} v_{ij}(x_1, \ldots, x_n, t)dw_j$$

with

$$\mu_{x_i} = a_i/x_i, \qquad v_{ij} = b_{ij}/x_i$$

The matrix $V = (v_{ij})_{(i,j)\in[1,n]^2}$ is called volatility matrix, and the covariance of relative returns is the matrix $\mathcal{K} = (k_{ij})_{(i,j)\in[1,n]^2}$:

$$\mathcal{K} = VV^T$$

3.2 European securities

A security is called *European security* iff future cash flows cannot be influenced by decisions from the holder taken after the purchase date (besides of course selling back the security). Then the cash flows are only functions of time and information.

The cash flow generated by a European security $\mathcal{C}$ during the time interval dt, assuming $\mathcal{C}$ is held indefinitely, will be denoted by $f_{\mathcal{C}}(X(t),t)dt$.

If the price $C(t)$ of a European security $\mathcal{C}$ is positive, we can define the instantaneous relative cash flow rate or *dividend yield* as:

$$d_{\mathcal{C}}(X(t),t) = f_{\mathcal{C}}(X(t),t)/C(t)$$

Let $\mu_{\mathcal{C}}$ denote the expected capital rate of return of $\mathcal{C}$:

$$\mu_{\mathcal{C}}dt = E_t(\frac{dC}{C})$$

In the above formula, E_t denotes the expectation conditional to the information available at time t. The expected total rate of return of $\mathcal{C}$, *i.e.* the expected rate of return of the total gain process is:

$$\mu_{G_{\mathcal{C}}} = \mu_{\mathcal{C}} + d_{\mathcal{C}}$$

The dividend yield of the money market account $\mathcal{L}$, called short term interest rate, is denoted by $r(X)$. If the proceeds of the money market account are continuously reinvested, the total gain process $L(t_0,t)$ follows the equation:

$$L(t_0,t_0) = 1, \quad dL = r(X)L(t_0,t)dt$$

or equivalently:

$$L(t_0,t) = \exp\left(\int_{t_0}^{t} r(X(\tau))d\tau\right)$$

3.3 Arbitrage pricing

For any risk factor x_i, let $e(C/x_i)$ denote the *elasticity* of price C to x_i:

$$e(C/x_i) = \frac{x_i}{C}\frac{\partial C}{\partial x_i}$$

The major result of the Arbitrage Pricing Theory is the following. There exist numbers $\lambda_1,\ldots,\lambda_n$, called market prices for risk, such that for any security $\mathcal{C}$, the following relationship

holds:

$$\mu_{G_C} = r + \sum_{i=1}^{n} e(C/x_i)\lambda_i$$

Furthermore, if factor x_i is traded, its market price for risk is the expected total rate of return on x_i in excess of the riskless interest rate. In particular, if all x_i are traded, we have:

$$\mu_{G_C} - r = \sum_{i=1}^{n} e(C/x_i)(\mu_{G_{x_i}} - r) = \sum_{i=1}^{n} e(C/x_i)(\mu_{x_i} + d_{x_i} - r)$$

Using the properties of diffusion processes, the above results lead to the following partial differential equation, called Black-Scholes equation:

$$-\frac{\partial C}{\partial t} = (d_C - r)C + \sum_{i=1}^{n} \frac{\partial C}{\partial x_i}(\mu_{x_i} - \lambda_i)x_i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 C}{\partial x_i \partial x_j}\gamma_{ij} \tag{3.2}$$

If factor x_i is the price of a traded security, the term $\mu_{x_i} - \lambda_i$ is simply $r - d_{x_i}$. In particular, if all factors are traded, the above equation does not depend on the market prices for risk. Therefore, we can replace the information process X by the so-called *risk-neutral* information process for which all market prices for risk are zero:

$$\forall i \in [1,n], \qquad \frac{dx_i}{x_i} = (r - d_{x_i})dt + \sum_{j=1}^{n} v_{ij} dw_j \tag{3.3}$$

Using a theorem known under the name of Feynman-Kac Formula, we can represent explicitly the solution of the above equation:

$$C(X(t),t) = \tilde{E}_t\left(\int_t^{\infty} \frac{f_C(X(\tau),\tau)}{L(t,\tau)} d\tau\right)$$

where $\tilde{E}_t$ represents the expectation under the fictitious risk-neutral information process X following the equation (3.3).

3.4 American securities

A security is called *American security* iff it is not European, i.e. if future cash flows can be influenced by decisions from the holder taken after the purchase date. Then, the cash flows are not only functions of time and information, but also functions of the decisions taken by the security's holder. A cash flow monitoring strategy u is a stochastic process associating with each time and information state a decision $u(X(t),t) \in U$, U being an appropriate *decision space*. We denote by CMS the space of cash flows monitoring strategies. For the sake of simplicity, we will consider only finite American securities, i.e. those for which CMS is a finite set. The cash-flow generated by an American security $\mathcal{C}$ during the time dt can be written:

$$f_C(u,X,t)dt$$

Let $\mathcal{C}$ be an American security with a cash flow function $f_{\mathcal{C}}(u, X, t)$. To each cash flow monitoring strategy u we can associate the European security $\mathcal{C}^u$ whose cash flow function $f^u_{\mathcal{C}}$ is:

$$f^u_{\mathcal{C}}(X,t) = f_{\mathcal{C}}(u(X,t), X, t)$$

Let us assume that the market is complete, i.e. that any factor x_i can be perfectly hedged by building an appropriate dynamic portfolio of traded assets, called *trading strategy*. Then, for any cash flow monitoring strategy u, the security $\mathcal{C}^u$ can be replicated by a trading strategy. We can therefore consider its arbitrage price C^u. In particular, we can consider the European security $\mathcal{C}^{u^*}$ maximizing the market value:

$$C^{u^*} = \max_{u \in \mathrm{CMS}} C^u$$

Barring arbitrage, we must have:

$$C = C^{u^*} = \max_{u \in \mathrm{CMS}} C^u$$

Indeed, if $C < C^{u^*}$, we can buy $\mathcal{C}$, sell $\mathcal{C}^{u^*}$, and take the proceeds $C^{u^*} - C$ immediately. Then, by selecting the cash flow monitoring strategy u^* for $\mathcal{C}$, all future cash outflows generated by the sale of $\mathcal{C}^{u^*}$ will be exactly compensated by the cash inflows generated by $\mathcal{C}$, so that we will not be obligated to any future payments. In effect, we will have made money without any investment and without risk.

Reciproqually, if $C > C^{u^*}$, let u_0 be the cash flow monitoring strategy chosen by a purchaser of security $\mathcal{C}$. Since by definition of u^* we have $C^{u^*} \geq C^{u_0}$, we must have $C > C^{u_0}$. By selling $\mathcal{C}$ and buying $\mathcal{C}^{u_0}$, the buyer can immediately take the proceeds $C - C^{u_0}$, without any change in the future cash flows. This is again an arbitrage opportunity.

We can state:

> *In a complete market, the price of an American security is the maximum over all possible cash flow monitoring strategies of the corresponding European prices.*

In other words, computing the price of an American security $\mathcal{C}$ reduces to computing that of the equivalent European security $\mathcal{C}^{u^*}$.

$$C(X(t), t) = \max_{u \in \mathrm{CMS}} \tilde{E}_t \left(\int_t^\infty \frac{f_{\mathcal{C}}(u(X(\tau), \tau), X(\tau), \tau)}{L(t, \tau)} d\tau \right)$$

Therefore, the differentiation between European and American securities is irrelevant in financial terms in a complete market free of arbitrage opportunities. On the other hand, if the market is not complete, there is no specific relationship between the American price and the corresponding European prices.

As a computational problem however, American security pricing is much harder that European security pricing. Indeed, the determination of the equivalent European security requires to precompute the optimal cash flow monitoring strategy. In many cases, this precomputation step is practically much more complex than the European price computation itself.

3.5 European and American options

3.5.1 European options

We consider an arbitrary asset $\mathcal{S}$ that can be replicated by an appropriate trading strategy. In a complete market, we can assume that the information vector $X = (x_1, \ldots, x_n)$ represents the prices of n given traded securities.

For example, $\mathcal{S}$ could be the right to purchase any one of the n securities at a given expiration date T. Barring arbitrage, the price $S(X(T), T)$ at expiration date is

$$S(X(T), T) = \max_{i \in [1,n]} x_i(T)$$

More generally, the price S can be any contractual function g of the underlying securities prices $x_1, \ldots, x_n$ at expiration.

$$S(X(T), T) = g(x_1(T), \ldots, x_n(T))$$

The function g, which represents the unique cash-flow associated with the contractual asset $\mathcal{S}$, is called the *payoff* function.

By definition, a European call (resp. put) option on an asset $\mathcal{S}$ with expiration date T and strike price K gives its owner the right to purchase (resp. sell) *at time* T the asset $\mathcal{S}$ for the price K. Since they leave a choice to the owner, European options are theoretically American securities. Indeed, the holder can choose to exercise or not exercise the option at expiration date. The decision space corresponds to this choice: $U = \{\texttt{exercise}, \texttt{no-exercise}\}$.

The cash flow monitoring strategy is any process associating the decision to the time t and the available information at t. The space of admissible cash flow monitoring strategies CMS is composed of all adapted processes u verifying:

$$\forall t < T, \forall X, \quad u(X, t) = \texttt{no-exercise}$$

and taking the two possible values at time T:

$$u(X, T) \in \{\texttt{exercise}, \texttt{no-exercise}\}$$

For the sake of simplicity, we will assume that the payoff of the option at exercise time T is distributed during a small time interval $[T, T + \Delta t]$. We will also assume that exercise decisions $u(X(t), t)$ are piecewise-constant, i.e. change only at the beginning of time intervals of duration Δt. For notational convenience, we will define the "spike" function $\delta(u)$ associating the value $1/\Delta t$ to the decision `exercise` and the value 0 to the decision `no-exercise`.

The cash flow function of a call option is:

$$f(u, X, t) = (S(X, t) - K)\delta(u)$$

The optimal exercise strategy u^* is easily identified:

$$\forall t \neq T, \quad u^*(X, t) = \texttt{no-exercise}$$

and

$$u^*(X,T) = \begin{cases} \texttt{exercise} & \text{if } S(X,T) > K \\ \texttt{no-exercise} & \text{otherwise} \end{cases}$$

Therefore, we identify the call option with its canonically associated optimal European security, whose cash flow function is:

$$f^{u^*}(X,t) = f(u^*(X,t),X,t) = \max(S(X,t) - K, 0)\delta(u^*)$$

Hence, from a computational viewpoint, the European call option can be considered as a European security.

If the payoff at expiration date T is $g(X)$, the price of the European call option can be written:

$$C(X(t),t) = \tilde{E}_t\left(\frac{f(X(T))}{L(t,T)}\right)$$

with

$$f(X) = \max(0, g(X) - K)$$

More generally, any European contingent claim with a single cash flow f at date T can be priced according to the above formula. Efficient numerical techniques based upon Monte Carlo simulation exist for computing numerically the above expectation in arbitrarily many dimensions (see e.g. Barraquand (1993)).

3.5.2 American options

By definition, an American call (resp. put) option on an asset $\mathcal{S}$ with expiration date T and strike price K gives its owner the right to purchase (resp. sell) *on or before time T* the asset $\mathcal{S}$ for the price K. The space of admissible cash flow monitoring strategies CMS is composed of all adapted processes u taking the two possible values:

$$\forall t \in [0,T], \quad u(X,t) \in \{\texttt{exercise}, \texttt{no-exercise}\}$$

and verifying:

$$\forall t \leq T, \quad u(X(t),t) = \texttt{exercise} \Rightarrow \forall \tau > t, u(X(\tau),\tau) = \texttt{no-exercise}$$

That is, exercise cannot occur twice. The cash flow function of an American call option is identical to that of a European call option. Unfortunately, in this case, the computation of the optimal early exercise strategy is not straightforward, since exercise can happen before expiration. We can simplify the above formulation by noticing that if it is optimal to exercise at a given underlying asset price S_0, it is also optimal to exercise at any higher price. Therefore, if we denote $H(X(t),t)$ the smallest possible value of S_0, the optimal early exercise stopping time τ^* is the solution of the following equation:

$$\tau^* = \inf\{t \in R_+, S(X(t),t) = H(X(t),t)\}$$

The following arbitrage argument shows that

$$H = K + C$$

where C is the American call option price. Indeed, whenever $C < S - K$, any investor could buy the option, exercise it immediately, and take the proceeds $(S - K) - C > 0$. Therefore, it is optimal to exercise whenever $S > K + C$, hence $H \leq K + C$.

Reciprocally, if $C > S - K$, no investor holding the option would be willing to exercise it and take $S - K$, since by just selling it he would make a higher immediate profit $C > S - K$. At any time, and for any information state, the optimal stopping time τ^* verifies:

$$C(X(\tau^*), \tau^*) = S(X(\tau^*), \tau^*) - K$$

Hence, the optimal early exercise strategy can be written:

$$u^*(X,t) = \begin{cases} \texttt{exercise} & \text{if} \quad C(X,t) \leq S(X,t) - K \\ \texttt{no-exercise} & \text{otherwise} \end{cases}$$

We see that the computation of the optimal early exercise strategy requires to precompute the pricing relationship between the option and the underlying asset, which is what we were trying to compute in the first place.

If the payoff at the exercise date is $g(X)$, the price of the American call (resp. put) option can be written:

$$C(X(t),t) = \max_{u(X(\tau),\tau),\tau \geq t, u \in \mathrm{CMS}} \tilde{E}_t \left(\sum_{\tau=t}^{T} \frac{f(X(\tau))}{L(t,\tau)} \delta(u(X(\tau),\tau)) \Delta t \right) \tag{3.4}$$

with

$$f(X) = \max(0, g(X) - K) \quad (\textit{resp.} \ \max(0, K - g(X)))$$

More generally, any contingent claim entitling its holder to the single cash flow f on or before an expiration date T can be priced according to the above formula.

4 Numerical methods for American security pricing

4.1 Stochastic dynamic programming

The explicit numerical valuation of an American option using the above formula involves a maximization over the set of of all possible early exercise strategies. The strategy u can be any function associating to each current value of the underlying assets $X = (x_1, \ldots, x_n)$ and each current time t the binary decision to exercise or not exercise. Since the number of such possible strategies is huge, direct maximization is rarely practical (see Bossaerts (1989) for a discussion).

The only practical technique to date consists in using the Bellman principle of Dynamic Programming. This principle can be applied since the information vector X is assumed to be a

Markov process, and therefore the optimal early exercise strategy $u(X(t),t)$ only depends upon time and the current vector $X(t)$.

Assuming that exercise decisions can only be taken at discrete times intervals of constant duration Δt, the maximization problem (3.4) can be rewritten using the law of iterated expectations and the property $L(t,t^*) = L(t,t+\Delta t)L(t+\Delta t,t^*)$:

$$C(X(t),t) = \max_{u(X(t),t),u\in \mathrm{CMS}} \tilde{E}_t \left[\frac{f(X(t))}{L(t,t+\Delta t)} \delta(u(X(t),t))\Delta t + \max_{u(X(\tau),\tau),\tau\geq t+\Delta t,u\in \mathrm{CMS}} \tilde{E}_{t+\Delta t} \left(\sum_{\tau=t+\Delta t}^{T} \frac{f(X(\tau))}{L(t+\Delta t,\tau)} \delta(u(X(\tau),\tau))\Delta t \right) \right]$$

Examining successively the two cases $u(X(t),t) =$ `exercise` and $u(X(t),t) =$ `no-exercise`, and using the expression of $C(X(t+\Delta t),t+\Delta t)$ from equation (3.4), we obtain

$$C(X(t),t) = \exp(-r\Delta t)\max\left(f(X(t)),\tilde{E}_t(C(X(t+\Delta t),t+\Delta t))\right) \tag{4.1}$$

assuming that the interest rate r is piecewise constant on intervals of duration Δt. The above recursive expression, called Bellman equation, allows to compute the price C of an American option by proceedings backwards in time from the expiration date T. Using the properties of diffusion processes (Itô's formula), it can be shown (see e.g. Jaillet et al. (1988)) that the solution of the above equation (4.1) converges towards the solution of the Black-Scholes equation (3.2) when Δt converges towards 0.

$$-\frac{\partial C}{\partial t} = -rC + \sum_{i=1}^{n} \frac{\partial C}{\partial x_i}(r-d_{x_i})x_i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 C}{\partial x_i \partial x_j} k_{ij}x_i x_j$$

with the additional time-dependent boundary condition:

$$C(X,t) \geq f(X)$$

4.2 Finite differences and the Cox-Ross-Rubinstein approach

The general method for solving the above partial differential equation (PDE) is to quantize it using a finite difference method (see e.g. Duffie (1992), chapter 10).

We will illustrate this approach using a number of simplifying assumptions. We will assume that the process of the underlying securities is jointly lognormal, i.e. that the mean and covariance matrix of relative returns are constant. We will also assume that the interest rate r and the dividend yields of the underlying securities are constant. Then, the change of variable below simplifies the finite difference approximation. We consider the vector $Y = (y_1,\ldots,y_n)^T$:

$$\forall i \in [1,n],\ y_i = \log x_i - (r - d_{x_i} - \frac{1}{2}k_{ii})t$$

and we then define the vector $W = (w_1,\ldots,w_n)^T$

$$W = V^{-1}Y$$

By construction, $Y = VW$, and W follows a k-dimensional standard Brownian motion. The Black-Scholes equation in the variable W writes:

$$-e^{rt}\frac{\partial(e^{-rt}C)}{\partial t} = \frac{1}{2}\sum_i \frac{\partial^2 C}{\partial w_i^2}$$

Defining

$$W_i^+ = (w_1, \ldots, w_i + \Delta w_i, \ldots, w_n)^T, \quad W_i^- = (w_1, \ldots, w_i - \Delta w_i, \ldots, w_n)^T$$

then writing

$$\forall i \in [1,n], \quad \frac{\partial^2 C}{\partial w_i^2} = \frac{C(W_i^+, t+\Delta t) - 2C(W, t+\Delta t) + C(W_i^-, t+\Delta t)}{\Delta w_i^2} + O(\Delta w_i^2)$$

and

$$-e^{rt}\frac{\partial(e^{-rt}C)}{\partial t} = \frac{e^{r\Delta t}C(W,t) - C(W,t+\Delta t)}{\Delta t} + O(\Delta t)$$

with

$$\forall i \in [1,n], \quad \Delta w_i = \sqrt{n\Delta t}$$

we get the simple explicit Euler scheme (see Duffie (1992) for a description of more sophisticated schemes):

$$C(W,t) = e^{-r\Delta t}\frac{1}{2n}\sum_{i=1}^{n} C(W_i^+, t+\Delta t) + C(W_i^-, t+\Delta t)$$

with the terminal boundary condition:

$$C(W(T), T) = f(X(T))$$

and the early exercise condition:

$$C(W(t), t) \geq f(X(t))$$

where $X(t) = (x_1(t), \ldots, x_n(t))$ is obtained by the inverse formula:

$$\forall i \in [1,n], \quad x_i(t) = x_i(0)\exp\left((r - d_{x_i} - \frac{1}{2}k_{ii})t + \sum_{j=1}^{n} v_{ij}w_k(t)\right) \tag{4.2}$$

We implemented the above numerical scheme for an arbitrary number n of underlying assets. The method yields accurate results for $n \leq 3$, but its memory requirement is intrinsically exponential in n. Hence, it cannot be used for $n > 3$ with a quantization step Δt small enough to yield accurate results (See section 8). Many variants and generalizations of the above finite-difference method have been studied by a number of authors (see Duffie (1992)).

An alternative technique consists in quantizing directly the Bellman equation (4.1). This is the paradigm underlying the original Cox-Ross-Rubinstein approach (Cox et al. (1979)). The

Brownian motion W is approximated by an n-dimensional binomial process $\tilde{W}^{\Delta t}$ defined as follows:

$$\forall \epsilon = (\epsilon_1, \ldots, \epsilon_n) \in \{-1, 1\}^n, \quad \text{Prob}(\tilde{W}^{\Delta t}(t + \Delta t) = \tilde{W}^{\Delta t}(t) + \epsilon\sqrt{\Delta t}) = \frac{1}{2^n}$$

Then, the quantized Bellman equation writes:

$$C(W, t) = \exp(-r\Delta t) \max \left(f(X(t)), \frac{1}{2^n} \sum_{\epsilon \in \{-1,1\}^n} C(W + \epsilon\sqrt{\Delta t}, t + \Delta t) \right) \tag{4.3}$$

For $n = 1$, the two formulations are exactly equivalent. In higher dimensions, they yield almost identical results for small enough Δt. In summary, the finite-difference and binomial lattice methods are essentially equivalent, and both intractable for $n > 3$ due to their exponential memory requirement.

5 State aggregation

5.1 State aggregation price

Classical numerical methods being unable to deal with high-dimensional American valuation problems, one must resort to alternate approximation schemes. State aggregation is a classical approximation technique for the numerical solution of stochastic optimal control problems (see e.g. Bertsekas (1987), Kushner and Dupuis (1992)).

For the problem of American security pricing, the relevant state space in the Bellman equation (4.1) is the n-dimensional space of the underlying assets values $X = (x_1, \ldots, x_n)$. State aggregation consists in partitioning the state space into a tractable number of cells, and in computing an approximate early exercise strategy $u(X, t)$ that is constant over those cells. The hope is that, if the partition is appropriately chosen, the approximate strategy will be close to the actual optimal strategy.

$\forall t \in \{0, \Delta t, \ldots, T\}$, let us consider a finite partition $P(t) = (P_1(t), \ldots, P_{k(t)}(t))$ of the state space R_+^n, i.e. a set of $k(t)$ subsets of R_+^n verifying:

$$\bigcup_{i \in [1,k(t)]} P_i(t) = R_+^n \text{ and } \forall (i, j) \in [1, k(t)]^2, i \neq j, \;\; P_i(t) \bigcap P_j(t) = \emptyset$$

We assume that the partition $P(0)$ only has two cells:

$$P_1(0) = \{X(0)\}, \quad \text{and} \;\; P_2(0) = R_+^n \backslash \{X(0)\}$$

Among the set CMS of all possible early exercise strategies, we consider the subset $\mathcal{U}(P)$ of piecewise constant strategies, i.e. of strategies $u(X, t)$ that are *constant* over each cell $P_i(t)$ of the partition.

Then, we define the *state aggregation price* $C^*(i,t)$ as the maximum over all possible piecewise constant strategies in $\mathcal{U}(P)$ of the expected risk-neutral discounted future cash-flow conditionally to the event $X(t) \in P_i(t)$:

$$C^*(i,t) = \max_{u(X(\tau),\tau),\tau\geq t,\ u\in\mathcal{U}(P)} \tilde{E}_0\left(\sum_{\tau=t}^{T} \frac{f(X(\tau),\tau)}{L(t,\tau)}\delta(u(X(\tau),\tau))\Delta t \mid X(t)\in P_i(t)\right) \quad (5.1)$$

Since $P_1(0) = \{X(0)\}$, and since $\mathcal{U}(P) \subset$ CMS, the state aggregation price at initial time $C^*(1,0)$ is obviously upper bounded by the true American price $C(X(0),0)$. Furthermore, since the strategy u_{Euro} corresponding to the European price consists in never exercising before the expiration date, it is clearly constant over the cells of any partition before expiration. Hence, by definition of the state aggregation price, the European price is upper bounded by the state aggregation price at the initial time $C^*(1,0)$. We can state

For any family of finite partitions P, the state aggregation price $C^* = C^*(1,0)$ is lower bounded by the European price and upper bounded by the American price.

$$C^{Euro} \leq C^* \leq C^{Amer}$$

We will now derive a recursive backward valuation formula for the state aggregation price, under an additional Markovian assumption.

5.2 Markovian approximation

We will now assume that the partition P is such that the process $I(t)$ defined by $X(t) \in P_{I(t)}(t)$ is approximately Markov under the risk neutral measure:

$$\begin{aligned}\forall i,j,\phi,\quad & \tilde{E}_0\left(\phi(X(t+2\Delta t)) \mid X(t)\in P_i(t),\ X(t+\Delta t)\in P_j(t+\Delta t)\right)\\ & \approx \tilde{E}_0\left(\phi(X(t+2\Delta t)) \mid X(t+\Delta t)\in P_j(t+\Delta t)\right)\end{aligned}$$

Applying again the law of iterated expectations to the definition of the state aggregation price in equation (5.1), we get:

$$\begin{aligned}C^*(I(t),t) = \max_{u(X(t),t),u\in\mathcal{U}(P)} \tilde{E}_0\Bigg[&\frac{f(X(t))}{L(t,t+\Delta t)}\delta(u(X(t),t))\Delta t + \max_{u(X(\tau),\tau),\tau\geq t+\Delta t,u\in\mathcal{U}(P)}\\ &\tilde{E}_0\left(\sum_{\tau=t+\Delta t}^{T}\frac{f(X(\tau))}{L(t+\Delta t,\tau)}\delta(u(X(\tau),\tau))\Delta t \mid X(t)\in P_{I(t)}(t),\ X(t+\Delta t)\in P_{I(t+\Delta t)}(t+\Delta t)\right)\\ &\mid X(t)\in P_{I(t)}(t)\Bigg]\end{aligned}$$

By examining successively the two cases $u(X(t),t) =$ `exercise` and $u(X(t),t) =$ `no-exercise`, and applying formula (5.1) at time $t+\Delta t$:

$$\begin{aligned}&C^*(I(t),t) \approx\\ &\quad e^{-r\Delta t}\max\left(\tilde{E}_0(f(X(t)) \mid X(t)\in P_{I(t)}(t)), \tilde{E}_0\left(C^*(I(t+\Delta t),t+\Delta t) \mid X(t)\in P_{I(t)}(t)\right)\right)\end{aligned}$$

where the substitution of formula (5.1) at time $t + \Delta t$ is justified by the above Markovian approximation.

By definition of a partition, we have:

$$1 = \sum_{j=1}^{k(t+\Delta t)} 1_{X(t+\Delta t) \in P_j(t+\Delta t)}$$

where 1_A is the indicator variable of the event A (i.e. takes the value 1 iff A is true, and 0 otherwise). Combining this with the above recursive equation, we get:

$$\forall i \in [1, k(t)], \quad C^*(i,t) \approx e^{-r\Delta t} \max \left(f_i(t), \sum_{j=1}^{k(t+\Delta t)} p_{ij}(t) C^*(j, t+\Delta t) \right)$$

where we have defined for notational convenience

$$f_i(t) = \tilde{E}_0(f(X(t)) \mid X(t) \in P_i(t))$$

and

$$p_{ij}(t) = \text{Prob}\left(X(t+\Delta t) \in P_j(t+\Delta t) \mid X(t) \in P_i(t)\right)$$

Furthermore, the value at expiration date is determined by the terminal condition:

$$C^*(i,T) = f_i(T)$$

5.3 Recursive state aggregation

We define the recursive state aggregation price C_{SA} as the solution of the following program:

$$C_{SA}(i,T) = f_i(T)$$

and

$$C_{SA}(i,t) = e^{-r\Delta t} \max \left(f_i(t), \sum_{j=1}^{k(t+\Delta t)} p_{ij}(t) C_{SA}(j, t+\Delta t) \right)$$

When the partitions $P(t)$ are chosen in such a way that the process $I(t)$ is actually Markovian, the recursive state aggregation price C_{SA} is exactly the true state aggregation price C^*. The implementation of a recursive state aggregation program proceeds in two steps.

1) Definition of an appropriate family of partitions,

2) Computation of the expected payoffs $f_i(t)$ and the conditional probability matrices $p_{ij}(t)$.

Then, the approximate price of an American contingent claim with terminal payoff f can be computed backwards in time using the above recursive equation.

Finite difference or lattice-based methods can be viewed as particular instances of the recursive state aggregation method. Indeed, each grid point or lattice point W^i can be viewed as the center of the hypercube $\text{Hyper}(W^i)$ defined by:

$$\text{Hyper}(W^i) = \{Z \in R^n, \forall j \in [1,n], \quad w_j^i - \frac{1}{2}\Delta w_j \leq z_j < w_j^i + \frac{1}{2}\Delta w_j\}$$

Although the process $I(t)$ such that $W(t) \in \text{Hyper}(W_{I(t)})$ is not actually Markovian, it can be approximated by a Markov process with reasonable accuracy for a fine enough lattice grid. In the lattice approach, this corresponds to approximating the Brownian motion W by the binomial process $W^{\Delta t}$. Then, the expected value over the hypercube $f_i(t)$ is approximated by the value in the center $f(X(W^i))$. Similarly, the conditional probability is simply taken from that of the binomial approximation, which yields the recursive formula (4.3). As explained in section 4.2, the limitation of this approach is the exponential growth of the number of lattice hypercubes in the number n of underlying assets.

One solution for generating a partition $P(t)$ with a tractable number of cells is to fix a priori a small number k and set for all $t > 0$, $k(t) = k$. Then, one can sample for each time t the state space with k samples following the risk-neutral distribution of $X(t)$. For example, if $X(t)$ follows a jointly lognormal distribution, the samples $X^i(t)$ can be computed from the samples W^i through formula (4.2). In turn, the samples $W^i(t)$ can be generated following a jointly standard normal distribution with variance t along each coordinate. Then, the cell $P_i(t)$ of the partition $P(t)$ can be defined, in the Brownian Motion space (i.e. the space of the variable W), as the set of points W closer to $W^i(t)$ than to any other sample. Such a partition is commonly called the Voronoi partition associated with the samples $W^1(t), \ldots, W^k(t)$. Unfortunately, Voronoi partitions have an undesirable asymptotic property for large n. The cells tend to become so large that the probability $p_{ij}(t)$ of moving into cell j at time $t + \Delta t$ from cell i at time t is almost 0 for all j's but one. Hence, a Voronoi partition of the n-dimensional state space does not accurately reproduce the diffusion effect in the information vector $X(t)$. In order to devise partitions such that several of the conditional probabilities $p_{ij}(t)$ are non-zero, one must build cells having a small *directional diameter* in the Brownian Motion space, i.e. such that the numbers

$$D_i = \sup_{X \in P_i(t)} \inf_{s=(s_1,\ldots,s_n), s_1^2+\ldots+s_n^2=1} \{\lambda_{\max} - \lambda_{\min}, \lambda_{\min} = \inf\{\lambda \in R, X + \lambda s \in P_i(t)\}, \lambda_{\max} = \sup\{\lambda \in R, X + \lambda s \in P_i(t)\}\}$$

are as small as possible. Indeed, the smaller D_i is, the higher is the probability that the Brownian motion process $W(t)$ crosses the boundary of the cell $P_i(t)$.

5.4 Stratified state aggregation (SSA)

One solution for choosing a partition with small cell directional diameters is to fix a real-valued function mapping the state that particularly influences the optimal strategy in the problem at hand. We call such a function a stratification map. Then, the partition chosen is a stratification of the state space into thin layers along this map. If the layers are chosen sufficiently thin, the

diameters of the cells along the direction of the gradient of the map will be small. Hence, the probability of crossing the boundary of a cell during a small time interval will be relatively high, and the drawback of Voronoi-based partitions will be avoided.

In order words, stratification consists in limiting the search to strategies that only depend upon the stratification map, and not upon the entire state itself. We call this technique *Stratified State Aggregation*.

In general, we can consider a vector-valued l-dimensional stratification map ($l < n$):

$$\begin{aligned} h: \quad R^n \times R &\longrightarrow R^l \\ (X,t) &\longrightarrow h(X,t) \end{aligned} \tag{5.2}$$

and a family Q of partitions of R^l. From the family Q and the map h, we can build the reciprocal image partition $P = h^{-1}(Q)$:

$$P_i(t) = \{X \in R^n_+, h(X,t) \in Q_i(t)\}$$

In the case of American security pricing, an obvious candidate for the stratification map is the *payoff* of the security. When the stratification map chosen is the payoff of the American security, we call the technique *Stratified State Aggregation along the Payoff* (SSAP).

Let us consider an American security with a single cash-flow $f(X)$ on or before an expiration date T. In particular, in the case of an American call option, the cash-flow is $f(X) = \max(0, g(X) - K)$, where K is the strike price. In order to illustrate the SSAP method, we set $l = 1$ and choose $h(X,t) = f(X)$ for the stratification map. We also take $k(t) = k$ constant for all times $t \in [0,T]$. In the numerical examples developed in section 8, we assume that $g(X) = \max_{i\in[1,n]} x_i$, and that the process X is jointly lognormal, of the form described in section 4.2. The partitions $Q(t)$ of the image space $R^l = R$ are chosen logarithmic in all these examples, i.e. the interval $Q_i(t), t > 0$ is of the form:

$$\forall i \in [2, k-1], \quad Q_i(t) = \left(A(t)e^{B(t)(i-2)}, A(t)e^{B(t)(i-1)}\right]$$

and

$$Q_1(t) = \left(-\infty, A(t)\right], \quad Q_k(t) = \left(A(t)e^{B(t)(k-2)}, +\infty\right)$$

for adequate parameters $A(t)$ and $B(t)$. The cell $P_i(t)$ is then by definition:

$$\forall i \in [2, k-1], \ P_i(t) = \{X \in R^n_+, \ A(t)e^{B(t)(i-2)} < f(X) \leq A(t)e^{B(t)(i-1)}\}$$

and

$$P_1(t) = \{X \in R^n_+, \ f(X) \leq A(t)\}, \qquad P_k(t) = \{X \in R^n_+, \ f(X) > A(t)e^{B(t)(k-2)}\}$$

In our experiments, the number k of cells is set to 100. The numbers $A(t)$ and $B(t)$ are automatically adjusted so as to ensure:

$$\text{Prob}\left(X(t) \in P_1(t)\right) \approx \text{Prob}\left(X(t) \in P_k(t)\right) \approx 0.1\%$$

The numerical results obtained with the SSAP method, presented in section (8), show that these empirical parameters are adequate for a broad range of American security pricing problems.

6 Monte Carlo estimation of American price

6.1 Generation of sample paths

Once the family of partitions P has been chosen, for example using the SSAP method, it remains to compute numerically the expected payoffs $f_i(t)$ and conditional probabilities $p_{ij}(t)$. These numbers can be expressed as integrals over the state space. In general, they must be computed numerically. The only general tractable method for computing such high-dimensional integrals is the Monte Carlo method.

It consists in generating a given number M of sample paths for the underlying assets price process $X(t)$. In general, this can be done through direct numerical integration of the Itô equation (3.3). A simple explicit Euler scheme is given by:

$$x_i(t+\Delta t) = x_i(t)\exp\left(\left(r - d_{x_i} - \frac{1}{2}k_{ii}\right)(X(t),t)\Delta t + \sum_{j=1}^{n} v_{ij}(X(t),t)\sqrt{\Delta t}\ z_j^t\right)$$

where z_j^t follow independent standard normal distributions for all j and t. $d = T/\Delta t$ being the number of time steps in $[0,T]$, we must draw a total of $M \times d \times n$ standard normal variates in order to generate M n-dimensional sample paths $X^1(t),\ldots,X^M(t)$ for all $t > 0$.

In general, Δt must be chosen small enough so as to reach a reasonable accuracy. In practice, a number of time steps $d = 100$ is sufficient in most asset pricing applications. However, when the joint process $X(t)$ is assumed lognormal as in section (4.2), d can be chosen much smaller. Indeed, the underlying assets price process X can then be obtained by formula (4.2) from a standard Brownian motion W. In our experiments, we found that a number of time steps $d = 10$ is sufficient for American security pricing with lognormal underlying assets price processes.

6.2 Conditional probabilities and payoff expectations

Once the M sample paths $X^1(t),\ldots,X^M(t)$ are computed, the number $a_i(t)$ of samples crossing $P_i(t)$ and the number $b_{ij}(t)$ of samples moving from $P_i(t)$ to $P_j(t+\Delta t)$ are easily computed:

$$\begin{aligned} a_i(t) &= \text{Card}\{k \in [1,M], X^k(t) \in P_i(t)\} \\ b_{ij}(t) &= \text{Card}\{k \in [1,M], X^k(t) \in P_i(t) \quad \text{and} \quad X^k(t+\Delta t) \in P_j(t+\Delta t)\} \end{aligned}$$

Similarly, the sum $c_i(t)$ over of samples X^k of payoff values $f(X^k(t))$ is computed from:

$$c_i(t) = \sum_{\{k\in[1,M], X^k(t)\in P_i(t)\}} f(X^k(t))$$

By the law of large numbers, we have the following identities:

$$p_{ij}(t) = \lim_{M\longrightarrow\infty} \frac{b_{ij}(t)}{a_i(t)} \qquad f_i(t) = \lim_{M\longrightarrow\infty} \frac{c_i(t)}{a_i(t)}$$

6.3 Backward integration algorithm

Using the above Monte-Carlo estimates of the conditional probabilities and payoff expectations, an approximation of the American price can then be computed backwards in time using the simple algorithm described below.

- At time T, the approximate SSAP price is initialized at:

$$C(i,T) = \frac{c_i(T)}{a_i(T)}$$

- At time $T - \Delta t$, we can compute for all $i \in [1,k]$:

$$C(i,T-\Delta t) = e^{-r\Delta t} \max\left(\frac{c_i(T-\Delta t)}{a_i(T-\Delta t)}, \sum_{j=1}^{k} C(j,T)\frac{b_{ij}(T-\Delta t)}{a_i(T-\Delta t)}\right)$$

- The above procedure is then applied recursively, backwards in time, to compute all the prices $C(i,T-2\Delta t), C(i,T-3\Delta t), \ldots, C(1,0) = C_{SSAP}$.

The memory required in the SSAP method is proportional to $k^2 \times d$, corresponding to the storage of the conditional probabilities $p_{ij}(t)$. The computation time is proportional to $M \times n^2 \times d + k^2 \times d$, the first term corresponding to the drawing of the M Monte Carlo sample paths, and the second to the backwards integration. Hence the memory and time complexities of the SSAP method are polynomial in n. This is to be contrasted with classical PDE methods which are exponential in n.

7 Quadratic resampling

7.1 Quadratic resampling for multidimensional Monte Carlo integration

Each standard normal variable z_j^t is simulated by generating M standard normal deviates $z_j^t(1), \ldots, z_j^t(M)$. Many variance reduction techniques exist to improve the computational efficiency of the Monte Carlo method. In particular, we use the well known technique of Antithetic Variables (AV), which consists in generating only $M/2$ standard normal deviates $z_j^t(1), \ldots, z_j^t(M/2)$ and take for the remaining samples $z_j^t(M/2+1), \ldots, z_j^t(M)$ the opposite values $-z_j^t(1), \ldots, -z_j^t(M/2)$ (M is assumed even).

In addition to the technique of Antithetic Variables, we also use an extension of the Quadratic Resampling technique. Quadratic Resampling was first presented in Barraquand (1993) for reducing the variance of multivariate Monte Carlo integration.

We briefly present below the original QR method before describing the extension we developed for the purpose of the SSA method. We consider the problem of computing the integral:

$$E(f(X)) = \int_{R^N} f(X)p(X)dX$$

where p is a given probability density function over R^N, and f any integrable function. We assume that the expected vector $E(X)$ and the covariance matrix K_X have been precomputed. In the application we consider here, $p(X)$ is the density of the N-dimensional standard normal distribution. Hence $E(X) = 0$ and $K_X = I_N$.

Then, we can approximate the integral $E(f(X))$ by:

$$\overline{f(X)} = \frac{1}{M}\sum_{i=1}^{M} f(X^i)$$

In particular, the empirical mean is:

$$\overline{X} = \frac{1}{M}\sum_{i=1}^{M} X^i$$

We can likewise define the empirical covariance:

$$\overline{K_X} = \overline{(X-\overline{X})(X-\overline{X})^T} = \overline{XX^T} - \overline{X}\,\overline{X}^T$$

We define the gain matrix (for M large enough, $\overline{K_X}$ is regular):

$$H = \sqrt{K_X}\sqrt{(\overline{K_X})}^{-1}$$

and the new random variable:

$$Y = H(X-\overline{X}) + E(X)$$

We consider the M samples $Y^i = H(X^i - \overline{X}) + E(X)$. For this particular sampling, we get:

$$\overline{Y} = \frac{1}{M}\sum_{i=1}^{M} Y^i = E(X)$$

Similarly:

$$\overline{K_Y} = \overline{(Y-\overline{Y})(Y-\overline{Y})^T} = \overline{(H(X-\overline{X}))(H(X-\overline{X}))^T} = H\overline{K_X}H^T$$

Using the definition of H we get:

$$\overline{K_Y} = K_X$$

Hence, the empirical first and second order moments using the samples Y^i are exactly equal to the real first and second order moments of X. In particular, the empirical mean of any polynomial f of degree two or less in the variables $x_1, \ldots, x_N$ verifies:

$$\overline{f(Y)} = \frac{1}{M}\sum_{i=1}^{M} f(Y^i) = E(f(X))$$

The method of quadratic resampling consists in using the samples Y^i in place of the samples X^i in the quadrature formula. We can state:

> *Any numerical quadrature formula generated through quadratic resampling is exact for any polynomial of degree two or less*

7.2 Quadratic resampling in space$^{\text{time}}$

In the American pricing problem, we must sample not only the underlying assets prices $X(T)$ at expiration date, but also those prices $X(t)$ for all possible early exercise dates. Hence, quadratic resampling must be applied to the underlying assets space domain elevated to the power of the time domain. In other words, we consider the variable $\mathcal{Z} = (z_j^t)_{j\in[1,n],t\in[\Delta t,T]}$ in the $n\times d$-dimensional space$^{\text{time}}$ domain $R^{n\times d}$. $\mathcal{Z}$ is a $n\times d$-dimensional standard normal variable. Given M standard normal deviates $z_j^t(1),\ldots,z_j^t(M)$ (with antithetic variables) for each of the $n\times d$ variables z_j^t, we can consider them as M vector samples $\mathcal{Z}^1,\ldots,\mathcal{Z}^M$ in $R^{n\times d}$. Then, we can apply Quadratic Resampling to the $n\times d$-vector variable $\mathcal{Z}$.

We have $E(\mathcal{Z}) = 0$ and $K_{\mathcal{Z}} = I_{n\times d}$. Since the samples are generated using antithetic variables, the empirical mean $\overline{\mathcal{Z}}$ is 0. We can compute the empirical $(n\times d)\times(n\times d)$ covariance matrix $\overline{K_{\mathcal{Z}}}$. Then, we can apply to each sample $\mathcal{Z}^i$ the transform:

$$\mathcal{Y}^i = \sqrt{(\overline{K_{\mathcal{Z}}})}^{-1}\,\mathcal{Z}^i$$

Finally, we can replace in the Monte Carlo simulations presented in section (6) the samples $z_j^t(i)$ by the samples $y_j^t(i)$, components of the $n\times d$-vector $\mathcal{Y}^i$. Experimental results reported in section (8) demonstrate the efficiency of the extended quadratic resampling method for American option pricing.

8 Experimental results

8.1 A test case

We implemented the method of Stratified State Aggregation along the Payoff function (SSAP). We also used the quadratic resampling technique for drawing the Monte Carlo sample paths. We present below some numerical results for several European and American option pricing problems ranging from 1 to 400 underlying assets. In all the experiments, we assumed that the American options can only be exercised at $d = 10$ different dates during the life of the option. This corresponds to choosing a time step $\Delta t = T/d = T/10$. We experimented with several different payoff functions, in particular with payoffs corresponding to the maximum, the minimum, or the average of the n underlying assets. We obtained similar results for all these different payoff functions. We only present below the case of an option on the maximum of the underlying assets. Its payoff function is defined by:

$$g(x_1,\ldots,x_n) = \max(0,\max(x_1,\ldots,x_n) - K)$$

where K is the strike price of the option.

In all the experiments, we assumed that the underlying assets price process $X(t)$ is lognormal, with a covariance matrix of relative returns $\mathcal{K}$ of the form:

$$\forall i \in [1,n],\ \ k_{ii} = \sigma_i^2$$

			Call option prices: $x_1(0)$ = \$40, r = 5%					
Parameters			European			American		
σ_1	T	K	C_{BS}	C_{SSAP}	4*stdev*	C_{BS}	C_{SSAP}	4*stdev*
20 %	1	35	5.15	5.15	0.001	5.15	5.15	0.002
		40	1.00	1.00	0.006	1.00	1.00	0.008
		45	0.02	0.02	0.001	0.02	0.02	0.001
	4	35	5.76	5.76	0.003	5.76	5.76	0.005
		40	2.16	2.16	0.006	2.16	2.16	0.008
		45	0.50	0.50	0.004	0.50	0.50	0.004
	7	35	6.40	6.40	0.004	6.40	6.40	0.006
		40	3.00	2.99	0.010	3.00	3.00	0.010
		45	1.09	1.09	0.006	1.09	1.09	0.006
40 %	1	35	5.38	5.38	0.003	5.38	5.40	0.007
		40	1.91	1.91	0.010	1.91	1.92	0.010
		45	0.41	0.41	0.003	0.41	0.41	0.003
	4	35	6.88	6.88	0.010	6.88	6.90	0.020
		40	3.96	3.96	0.020	3.96	3.97	0.020
		45	2.08	2.08	0.006	2.08	2.09	0.010
	7	35	8.07	8.08	0.010	8.07	8.10	0.020
		40	5.35	5.34	0.024	5.35	5.36	0.040
		45	3.40	3.40	0.012	3.40	3.42	0.020

			Put option prices: $x_1(0)$ = \$40, r = 5%					
Parameters			European			American		
σ_1	T	K	P_{BS}	P_{SSAP}	4*stdev*	P_{PDE}	P_{SSAP}	4*stdev*
20 %	1	35	0.00	0.00	0.001	0.00	0.00	0.001
		40	0.83	0.83	0.008	0.84	0.84	0.008
		45	4.84	4.84	0.001	5.00	5.00	0.000
	4	35	0.19	0.19	0.003	0.19	0.19	0.004
		40	1.50	1.50	0.006	1.56	1.56	0.010
		45	4.77	4.77	0.004	5.06	5.07	0.010
	7	35	0.41	0.41	0.003	0.42	0.42	0.006
		40	1.86	1.86	0.010	1.96	1.96	0.012
		45	4.82	4.82	0.006	5.24	5.23	0.016
40 %	1	35	0.24	0.24	0.003	0.24	0.24	0.003
		40	1.74	1.74	0.010	1.75	1.76	0.010
		45	5.23	5.23	0.003	5.27	5.29	0.015
	4	35	1.31	1.31	0.010	1.32	1.33	0.020
		40	3.31	3.31	0.020	3.36	3.37	0.020
		45	6.35	6.35	0.006	6.47	6.49	0.015
	7	35	2.08	2.09	0.012	2.12	2.13	0.015
		40	4.21	4.21	0.020	4.31	4.31	0.016
		45	7.12	7.13	0.012	7.36	7.34	0.030

Table 1: Results of the SSAP method with 1 underlying asset

and

$$\forall (i,j) \in [1,n]^2, i \neq j, \;\; k_{ij} = \rho \sigma_i \sigma_j$$

for $n+1$ numbers $\sigma_1 > 0, \ldots, \sigma_n > 0$ and $-1/(n-1) \leq \rho \leq 1$.

Volatilities (σ_i), correlations (ρ), and interest rate (r) are counted in percent per year. The time to expiration T is counted in months, with the convention 1 month = 30 days. All asset and strike prices are counted in dollars.

Since SSAP uses Monte Carlo simulation, we report confidence intervals together with all results. These confidence intervals are computed from the central limit theorem, i.e. we assume that at a confidence level of 99.95%, the error must be less than 4 times the observed standard deviation of the result. The confidence interval reported is $4 \times$ stdev.

All the simulations were run on a DEC 3000 model 500X workstation, with an ALPHA AXP processor running at a clock rate of 200 Mhz, and 1 Gigabyte of main memory.

8.2 One underlying asset

We first study the one-dimensional case. In this case, the SSAP price should converge toward the theoretical arbitrage price when both the number of time steps d and the number of cells k converge towards infinity. Both European calls and European puts can be priced according to the original Black-Scholes formula. These prices are reported in the columns European C_{BS} and European P_{BS} of table (1). The American call can also be priced according to the same formula, since we assume the underlying asset pays no dividends. The price is reported in column American C_{BS}. For the American put, we computed the price using the finite-difference method presented in section (4.2). We call this method PDE, since it consists in solving a Partial Differential Equation. In dimension 1, it is essentially equivalent to the Cox-Ross-Rubinstein binomial lattice method. We used 120 time steps for T (time to expiration) ranging from 1 to 4 months, and 210 time steps for T = 7 months. The corresponding price is reported in column American C_{PDE}. The SSAP prices where computed using $M = 100,000$ samples, and $k = 100$ buckets. The number of time steps was set to $d = 10$ in all the experiments.

The observed differences between the SSAP prices and the reference prices are below 0.7%. The confidence interval values are below 1% of the reference prices. American put prices given by the SSAP method are very accurate even when the difference with the European put prices are important (up to 30 cents). The computation time of a price using the SSAP method is about 21 seconds, compared with less than one second with a classical integration method (PDE). In dimension 1, classical finite difference of binomial lattice methods should be preferred to the SSAP method.

8.3 Two underlying assets

In this case the SSAP method only finds an approximation of the optimal price. However, numerical experiments show that the SSAP price always remains within a few cents of the optimal theoretical price.

			Call option prices: $x_1(0) = x_2(0)$ = \$40					
			σ_1 = 20%, σ_2 = 30%, r = 5%					
Parameters			European			American		
ρ	T	K	C_{PDE}	C_{SSAP}	4*stdev*	C_{PDE}	C_{SSAP}	4*stdev*
0 %	1	35	6.80	6.79	0.010	6.80	6.80	0.012
		40	2.10	2.11	0.002	2.10	2.11	0.002
		45	0.18	0.18	0.003	0.18	0.18	0.003
	4	35	8.90	8.89	0.010	8.90	8.90	0.010
		40	4.48	4.48	0.015	4.48	4.49	0.015
		45	1.66	1.66	0.006	1.66	1.66	0.008
	7	35	10.43	10.42	0.020	10.43	10.43	0.020
		40	6.15	6.15	0.016	6.15	6.16	0.020
		45	3.10	3.08	0.004	3.10	3.09	0.010
50 %	1	35	6.36	6.35	0.004	6.36	6.36	0.005
		40	1.87	1.87	0.006	1.87	1.87	0.008
		45	0.17	0.17	0.004	0.17	0.17	0.004
	4	35	8.09	8.08	0.015	8.09	8.09	0.016
		40	3.99	3.98	0.010	3.99	3.99	0.010
		45	1.51	1.50	0.006	1.51	1.51	0.006
	7	35	9.41	9.40	0.015	9.41	9.41	0.015
		40	5.48	5.47	0.010	5.48	5.47	0.016
		45	2.77	2.77	0.010	2.77	2.78	0.012
100 %	1	35	5.61	5.61	0.002	5.61	5.61	0.006
		40	1.45	1.45	0.008	1.45	1.46	0.008
		45	1.16	1.16	0.002	1.16	1.16	0.003
	4	35	6.69	6.68	0.003	6.69	6.69	0.008
		40	3.09	3.07	0.010	3.09	3.08	0.012
		45	1.24	1.24	0.004	1.24	1.25	0.010
	7	35	7.62	7.61	0.006	7.62	7.62	0.012
		40	4.23	4.21	0.020	4.23	4.22	0.020
		45	2.24	2.22	0.012	2.24	2.23	0.020

Table 2: Prices for a call option with 2 underlying assets

			Put option prices: $x_1(0) = x_2(0)$ = \$40					
			σ_1 = 20%, σ_2 = 30%, r = 5%					
Parameters			European			American		
ρ	T	K	P_{PDE}	P_{SSAP}	$4stdev$	P_{PDE}	P_{SSAP}	$4stdev$
0 %	1	35	0.00	0.00	0.0000	0.00	0.00	0.0000
		40	0.29	0.29	0.0030	0.40	0.40	0.0040
		45	3.34	3.35	0.0060	5.00	5.00	0.0000
	4	35	0.03	0.03	0.0020	0.04	0.04	0.0020
		40	0.53	0.54	0.0080	0.75	0.75	0.0080
		45	2.62	2.63	0.0200	5.00	5.00	0.0000
	7	35	0.08	0.08	0.0030	0.10	0.10	0.0040
		40	0.66	0.67	0.0120	0.95	0.95	0.0100
		45	2.47	2.46	0.0200	5.00	5.00	0.0000
50 %	1	35	0.00	0.00	0.0006	0.00	0.00	0.0006
		40	0.50	0.50	0.0030	0.56	0.57	0.0050
		45	3.78	3.78	0.0080	5.00	5.00	0.0000
	4	35	0.09	0.09	0.0030	0.10	0.10	0.0030
		40	0.92	0.91	0.0100	1.08	1.08	0.0100
		45	3.35	3.35	0.0100	5.00	5.00	0.0000
	7	35	0.21	0.21	0.0040	0.24	0.24	0.0040
		40	1.14	1.13	0.0080	1.38	1.38	0.0120
		45	3.29	3.30	0.0120	5.00	5.00	0.0000
100 %	1	35	0.01	0.01	0.0008	0.01	0.01	0.0006
		40	0.83	0.83	0.0040	0.84	0.85	0.0060
		45	4.52	4.52	0.0040	5.00	5.00	0.0000
	4	35	0.19	0.19	0.0040	0.19	0.20	0.0040
		40	1.51	1.51	0.0060	1.56	1.56	0.0120
		45	4.58	4.59	0.0040	5.02	5.02	0.0120
	7	35	0.41	0.41	0.0020	0.42	0.42	0.0060
		40	1.87	1.86	0.0120	1.96	1.97	0.0100
		45	4.74	4.73	0.0080	5.20	5.21	0.0120

Table 3: Prices for a put option with 2 underlying assets

			Call option prices: $x_1(0) = x_2(0) = x_3(0)$ = \$40					
			σ_1 = 20%, σ_2 = 30%, σ_3 = 50%, r = 5%					
Parameters			European			American		
ρ	T	K	C_{PDE}	C_{SSAP}	4*stdev*	C_{PDE}	C_{SSAP}	4*stdev*
0 %	1	35	8.59	8.58	0.008	8.59	8.59	0.010
		40	3.84	3.83	0.010	3.84	3.84	0.012
		45	0.89	0.89	0.006	0.89	0.90	0.007
	4	35	12.55	12.53	0.020	12.55	12.55	0.016
		40	7.87	7.85	0.014	7.87	7.87	0.020
		45	4.26	4.25	0.014	4.26	4.27	0.014
	7	35	15.29	15.27	0.020	15.29	15.30	0.030
		40	10.72	10.70	0.016	10.72	10.73	0.035
		45	6.96	6.95	0.020	6.96	6.98	0.020
50 %	1	35	7.78	7.77	0.010	7.78	7.78	0.012
		40	3.18	3.17	0.010	3.18	3.18	0.012
		45	0.82	0.82	0.004	0.82	0.83	0.004
	4	35	10.97	10.95	0.010	10.97	10.96	0.015
		40	6.69	6.67	0.016	6.69	6.69	0.020
		45	3.70	3.69	0.012	3.70	3.71	0.025
	7	35	13.23	13.21	0.016	13.23	13.24	0.030
		40	9.11	9.09	0.020	9.11	9.12	0.040
		45	5.98	5.98	0.016	5.98	5.99	0.020
100 %	1	35	6.53	6.52	0.006	6.53	6.54	0.010
		40	2.38	2.37	0.010	2.38	2.38	0.012
		45	0.74	0.74	0.002	0.74	0.74	0.003
	4	35	8.51	8.50	0.008	8.51	8.53	0.016
		40	4.92	4.90	0.012	4.92	4.93	0.020
		45	2.97	2.96	0.008	2.97	2.99	0.020
	7	35	10.04	10.03	0.010	10.04	10.07	0.016
		40	6.64	6.63	0.016	6.64	6.67	0.030
		45	4.61	4.60	0.016	4.61	4.64	0.040

Table 4: Prices for a call option with 3 underlying assets

			Put option prices: $x_1(0) = x_2(0) = x_3(0)$ = \$40					
			σ_1 = 20%, σ_2 = 30%, σ_3 = 50%, r = 5%					
Parameters			European			American		
ρ	T	K	P_{PDE}	P_{SSAP}	4*stdev*	P_{PDE}	P_{SSAP}	4*stdev*
0 %	1	35	0.00	0.00	0.000	0.00	0.00	0.000
		40	0.13	0.13	0.003	0.23	0.23	0.003
		45	2.26	2.27	0.012	5.00	5.00	0.000
	4	35	0.01	0.01	0.002	0.01	0.01	0.001
		40	0.25	0.25	0.006	0.44	0.45	0.006
		45	1.55	1.56	0.010	5.00	5.00	0.000
	7	35	0.03	0.03	0.004	0.04	0.04	0.002
		40	0.31	0.32	0.010	0.57	0.58	0.015
		45	1.41	1.42	0.020	5.00	5.00	0.000
50 %	1	35	0.00	0.00	0.000	0.00	0.00	0.000
		40	0.38	0.39	0.003	0.48	0.49	0.006
		45	3.00	3.01	0.010	5.00	5.00	0.000
	4	35	0.07	0.08	0.006	0.09	0.09	0.004
		40	0.72	0.72	0.006	0.93	0.94	0.010
		45	2.65	2.66	0.010	5.00	5.00	0.000
	7	35	0.17	0.17	0.006	0.20	0.20	0.005
		40	0.91	0.91	0.012	1.19	1.21	0.010
		45	2.63	2.65	0.012	5.00	5.00	0.000
100%	1	35	0.01	0.01	0.001	0.01	0.01	0.001
		40	0.84	0.84	0.004	0.84	0.85	0.006
		45	4.18	4.18	0.003	5.00	5.00	0.000
	4	35	0.19	0.19	0.003	0.19	0.19	0.004
		40	1.51	1.51	0.006	1.56	1.57	0.008
		45	4.49	4.49	0.005	5.00	5.00	0.008
	7	35	0.41	0.41	0.004	0.42	0.42	0.004
		40	1.87	1.86	0.010	1.96	1.97	0.012
		45	4.70	4.70	0.008	5.20	5.20	0.012

Table 5: Prices for a put option with 3 underlying assets

The European and American call and put option prices can be computed by the PDE method. This integration requires 120 time steps for $T = 1$ and $T = 4$ months and 210 time steps for $T = 7$ months. These results are reported in the columns European C_{PDE}, P_{PDE} and American C_{PDE}, P_{PDE} in tables (2) and (3). The SSAP method was run using $M = 100,000$ samples, and $k = 100$ buckets. The number of time steps was set to $d = 10$ in all the experiments.

The observed differences between the SSAP prices and the reference prices are below 1%. The confidence interval value is below 1% of the reference prices, except for very low prices where it remains under 1 cent. American put prices given by the SSAP method are very accurate. The computation time of a price using the SSAP method is about 25 seconds, compared with 23 seconds for the classical integration method (PDE). In dimension 2, classical finite difference or binomial lattice methods are essentially equivalent to the SSAP method.

8.4 Three underlying assets

In this case again, the SSAP method only finds an approximation of the optimal price. However, numerical experiments show that the SSAP price always remains within a few cents of the optimal theoretical price. All parameters are identical to those of the 2D case, both for the PDE and the SSAP method. Results are presented in tables (4) and (5).

The observed differences between the SSAP prices and the reference prices are below 1%. The confidence interval value is below 1% of the reference prices, except for very low prices where it remains under 1 cent. American put prices given by the SSAP method are very accurate. The computation time of a price using the SSAP method is about 32 seconds, compared with 202 seconds for the classical integration method (PDE). In dimension 3, the SSAP method is as accurate and about 6 times faster than the classical integration method PDE.

8.5 Ten underlying assets

In the previous subsections 8.2, 8.3 and 8.4, we compared the efficiency and accuracy of the SSAP method with that of the classical integration method (PDE). In this subsection, we report results obtained with the SSAP method on American option pricing problems with 10 underlying assets (tables (6) and (7)). Since no other method exists to compare to our results, and since the SSAP method only provides an approximation of the optimal price, we cannot guarantee the accuracy of the American premiums reported below. However, both the observed confidence intervals and the fact that the SSAP prices for the American calls without dividends equal the European prices lead us to believe that the SSAP method is reliable in general on 10-dimensional American pricing problems. The parameters of the SSAP method are again $M = 100,000$ and $k = 100$.

The differences between European and American call prices are below 0.5 %. Since the payoff is the maximum of n underlying assets prices without dividends, these two prices should indeed be identical. Confidence intervals are around 1% in the worst case. The computation time using the SSAP method is about 82 seconds.

			Call option prices: $x_1(0) = \ldots = x_{10}(0)$ = \$40			
			$\sigma_1 = \ldots = \sigma_{10}$ = 40%, r = 5%			
Parameters			European		American	
ρ	T	K	C_{SSAP}	4*stdev*	C_{SSAP}	4*stdev*
0 %	1	35	12.66	0.010	12.66	0.010
		40	7.68	0.020	7.68	0.020
		45	2.98	0.020	2.98	0.020
	4	35	21.53	0.060	21.54	0.050
		40	16.62	0.030	16.62	0.030
		45	11.76	0.040	11.76	0.040
	7	35	27.91	0.060	27.92	0.060
		40	23.06	0.060	23.08	0.060
		45	18.24	0.040	18.25	0.040
50 %	1	35	10.36	0.008	10.36	0.008
		40	5.54	0.010	5.54	0.010
		45	1.90	0.010	1.90	0.010
	4	35	16.52	0.020	16.53	0.020
		40	11.87	0.020	11.87	0.020
		45	7.81	0.040	7.81	0.040
	7	35	20.91	0.060	20.92	0.050
		40	16.38	0.040	16.38	0.040
		45	12.28	0.020	12.28	0.020
100 %	1	35	5.41	0.003	5.42	0.010
		40	1.93	0.006	1.93	0.010
		45	0.42	0.004	0.42	0.004
	4	35	6.93	0.006	6.95	0.010
		40	4.00	0.012	4.02	0.020
		45	2.11	0.006	2.11	0.010
	7	35	8.14	0.010	8.16	0.020
		40	5.40	0.016	5.42	0.020
		45	3.44	0.012	3.45	0.020

Table 6: Prices of a call option with 10 underlying assets

			Put option prices:$x_1(0) = \ldots = x_{10}(0)$ = \$40			
			$\sigma_1 = \ldots = \sigma_{10}$ = 40%, r = 5%			
Parameters			European		American	
ρ	T	K	P_{SSAP}	4*stdev*	P_{SSAP}	4*stdev*
0 %	1	35	0.00	0.0000	0.00	0.0000
		40	0.00	0.0000	0.00	0.0000
		45	0.28	0.0060	5.00	0.0000
	4	35	0.00	0.0000	0.00	0.0000
		40	0.00	0.0008	0.01	0.0008
		45	0.06	0.0060	5.00	0.0000
	7	35	0.00	0.0000	0.00	0.0000
		40	0.00	0.0008	0.01	0.0010
		45	0.04	0.0030	5.00	0.0000
50 %	1	35	0.00	0.0008	0.00	0.0006
		40	0.16	0.0030	0.26	0.0040
		45	1.49	0.0120	5.00	0.0000
	4	35	0.05	0.0040	0.07	0.0030
		40	0.32	0.0060	0.52	0.0060
		45	1.17	0.0200	5.00	0.0000
	7	35	0.10	0.0060	0.15	0.0060
		40	0.42	0.0120	0.69	0.0080
		45	1.18	0.0200	5.00	0.0000
100 %	1	35	0.24	0.0020	0.24	0.0020
		40	1.73	0.0060	1.75	0.0080
		45	5.20	0.0040	5.27	0.0100
	4	35	1.30	0.0060	1.32	0.0100
		40	3.28	0.0160	3.34	0.0120
		45	6.31	0.0060	6.45	0.0100
	7	35	2.06	0.0100	2.11	0.0160
		40	4.18	0.0160	4.29	0.0160
		45	7.08	0.0150	7.30	0.0240

Table 7: Prices of a put option with 10 underlying assets

	Call confidence interval: 4*stdev*					
	$x_i(0) = 40\$, \sigma_i = 40\%, \rho = 1/2, T = 4, r = 5\%$					
	European			American		
Dimension (N)	no QR	QR	gain	no QR	QR	gain
1	0.138842	0.0455633	3.04	0.116406	0.0522995	2.23
2	0.178123	0.0355280	5.01	0.150975	0.0372755	4.05
3	0.120440	0.0359650	3.35	0.122352	0.0416212	2.94
5	0.127231	0.0641982	1.98	0.117853	0.0523125	2.25
10	0.254137	0.0813637	3.12	0.253496	0.0858543	2.95
20	0.163366	0.1256530	1.30	0.159423	0.1439880	1.11
40	0.188222	0.1185550	1.59	0.188743	0.1212830	1.56

	Put confidence interval: 4*stdev*					
	$x_i(0) = 40\$, \sigma_i = 40\%, \rho = 1/2, T = 4, r = 5\%$					
	European			American		
Dimension (N)	no QR	QR	gain	no QR	QR	gain
1	0.0837939	0.0454326	1.84	0.0717408	0.0364325	1.97
2	0.1112430	0.0202023	5.51	0.0712059	0.0270917	2.63
3	0.0703473	0.0362193	1.94	0.0650541	0.0231927	2.80
5	0.0649703	0.0236217	2.75	0.0762900	0.0177446	4.30
10	0.0504558	0.0215351	2.34	0.0412889	0.0240727	1.71
20	0.0351752	0.0216874	1.62	0.0338044	0.0144007	2.35
40	0.0192215	0.0115251	1.67	0.0199901	0.0104772	1.91

Table 8: Efficiency of quadratic resampling (QR)

8.6 Efficiency of extended quadratic resampling

In this subsection we analyse the efficiency of the Quadratic Resampling method presented in section 7. The results shown in table (8) are the confidence intervals with (*QR*) and without (*noQR*) quadratic resampling. The *gain* value is the ratio of these two figures ($noQR/QR$). We present in figure (1) the speed-up obtained through quadratic resampling. Since the accuracy of the Monte Carlo method is proportional to the square root of the number of samples (central limit theorem), the speedup obtained through Quadratic Resampling is the square of the gain ($speed-up = gain^2$). In these experiments, we used $M = 10,000$ sample paths and $k = 100$ buckets.

Quadratic resampling is very efficient for a small number of underlying assets (up to 10). The speed-up ranges from 5 to 30, with an average of about 10. The speed-up decreases in higher dimensions. A speed-up of 2.5 is obtained with 40 underlying assets.

8.7 Experimental time complexity

We analyze in this subsection the computation time required by the SSAP method as a function of the number n of underlying assets. We compare these results when possible to those of

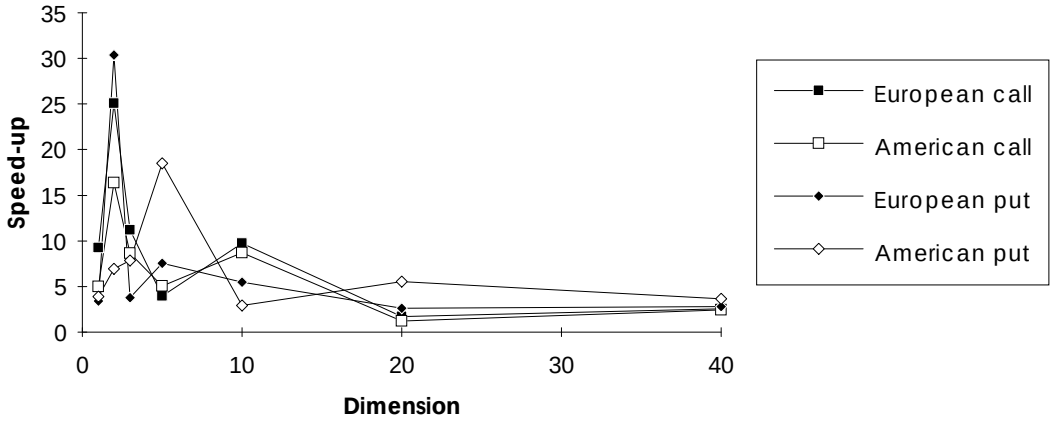

Figure 1: Speed-Up factor obtained through quadratic resampling

classical integration methods (PDE). Observed computation times are presented in table (9). The number of Monte Carlo samples is $M = 100,000$. The number of cells is $k = 100$ as before.

Figure (2) shows that for up to 40 underlying assets, the computation time is linear in n. The dominating term is the computation of the payoff function which is linear in n ($M \times n \times d$). Figure (3) shows that for larger n the complexity is quadratic in n ($M \times n^2 \times d$), as expected.

For $n = 2$ the SSAP is comparable to the classical integral method. For $n = 3$ the SSAP method is faster by a factor of 6. For $n > 3$ the SSAP method is the only one which can compute the price accurately. The integral (PDE) method is implemented using 60 time steps in the Cox-Rubinstein tree to obtain comparable precision. The exercise condition is also applied only 10 times during the life period of the option.

8.8 Parallel implementation

We implemented a parallel version of the SSAP method on a network of 4 DEC 3000 model 500 ALPHA AXP workstations equipped with a high-bandwidth Gigaswitch fiber optic interconnect (called a *workstation farm*). We observed a speedup linear in the number of workstations in the network: the parallel version is 4 times faster than the sequential implementation (table (10)). We anticipate that these figures would scale up with the number of workstations in the network. Indeed, the parallelization paradigm for Monte Carlo simulation is particularly simple. It consists in distributing on different processors the computation of different sample paths, and finally adding the results obtained by all processors.

Let us assume that we have λ independent computational units. The preliminary computations of the values $a_i(t), b_{ij}(t), c_i(t)$ (see section 6) by Monte Carlo simulation can be done separately on each unit for M/λ Monte Carlo samples. Then they must be consolidated on one unit, called the master unit. The backward integration in time is then performed on the master unit using the

Dimension (N)	Time SSAP method (second)	Time PDE method (second)
1	21	0.12
2	25	23
3	32	202
5	46	Out of memory
10	81	Out of memory
20	151	Out of memory
40	309	Out of memory
60	567	Out of memory
80	894	Out of memory
100	1018	Out of memory
200	2438	Out of memory
400	7069	Out of memory

Table 9: Computation times as functions of the dimension

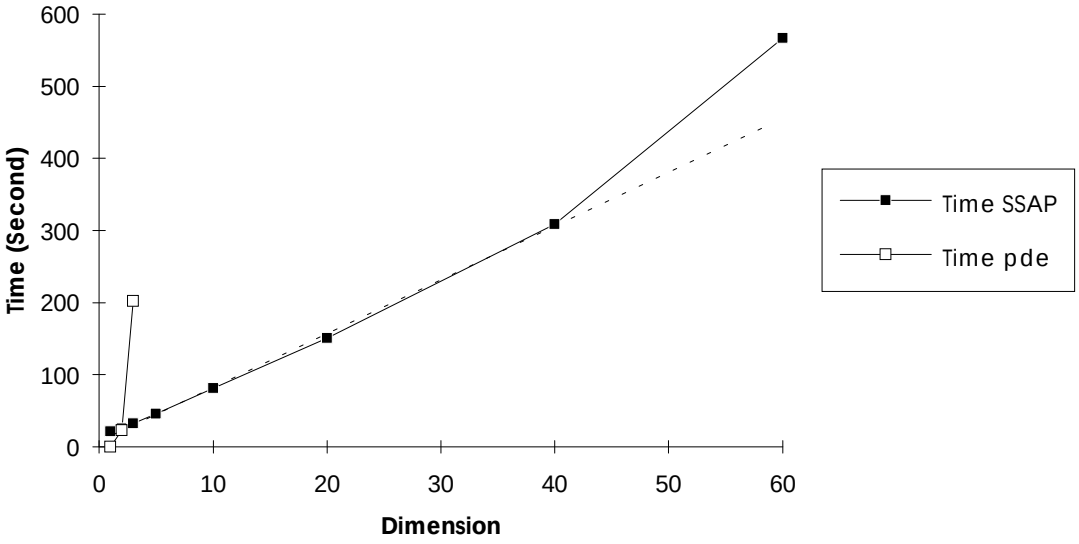

Figure 2: Linear behavior of computation time for $0 < n \leq 60$

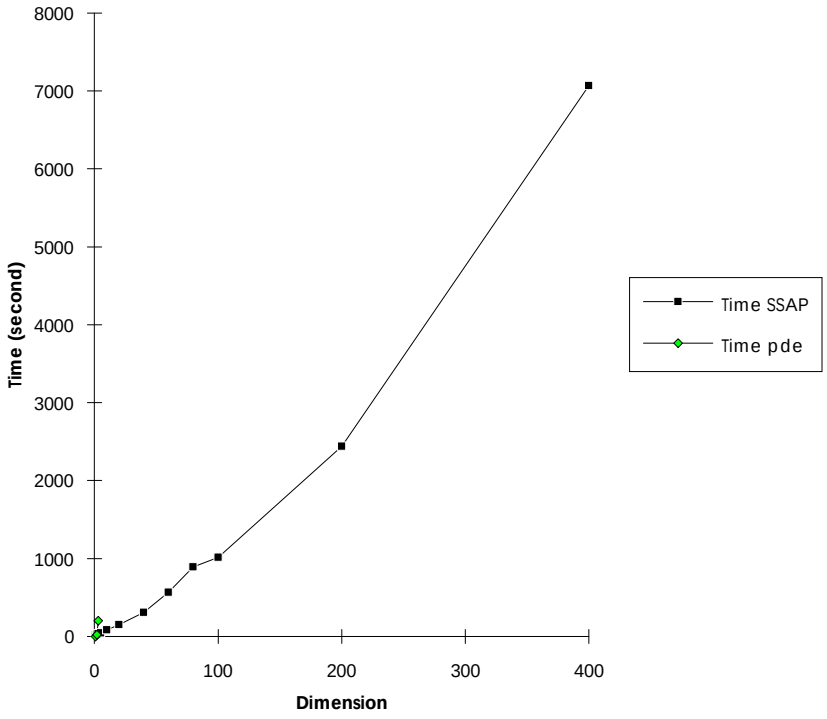

Figure 3: Quadratic behavior of computation time for $0 < n \leq 400$

Dimension (n)	$\lambda = 4$ processors (measured)	$\lambda = 32$ processors (estimated)
1	??	1.65
2	??	1.78
3	??	2.00
5		2.44
10		3.53
20		5.72
40		10.65
60		18.72
80		28.94
100		32.82
200		77.19
400		220.91

Table 10: Linear speedup of the parallel implementation

consolidated values $a_i(t), b_{ij}(t), c_i(t)$. The memory requirement is multiplied by λ as compared to a sequential implementation of the SSAP method. The gain in computation time is almost linear, i.e. the time complexity can be divided by λ. Indeed, the results of the previous subsection have shown that the dominating terms are: $(M \times n \times d)$ for the computation of the payoffs and $(M \times n^2 \times d)$ for drawing the Monte Carlo sample paths. But these two operations are done on each computational unit separately. The computation time required for the backward integration can be neglected. ($\simeq$ 1 second with the parameters of the previous subsection). The only overheads added by the parallelization are:

- λ communications of an amount of memory proportional to $n^2 \times d$ (typically 2 Mbytes).
- Consolidation of the λ sets of values.

For n and d fixed, this overhead is constant and can be neglected in practice ($\simeq$ 1second for the parameters of the previous subsection). We present in table (10) observed figures for $\lambda = 4$, and estimated figures for $\lambda = 32$.

9 Conclusion

In this article, we described a systematic numerical technique for pricing arbitrarily complex American contingent claims, i.e. generalized option contracts with possibilities of early exercise. Besides its obvious applications to trading and hedging in organized and Over The Counter (OTC) capital markets, American security pricing has many important applications in various areas of risk management such as assets and liabilities management and corporate investment decision making. Using this technique, we were able to compute the prices of complex American instruments in a few tens of seconds on a workstation, and within a few

seconds on a network of workstations.

Our approach essentially relies on appropriate state aggregation techniques that circumvent the intractability of the computation of the early exercise boundary, combined with a classical Monte Carlo simulation for the computation of the conditional probabilities in the backwards pricing formula. We call this method Stratified State Aggregation along the Payoff function (SSAP). We have successfully implemented the SSAP method for problems with up to 400 dimensions. To the best of our knowledge, no other method has ever been developed to date for pricing American contingent claims with many (more than 3 or 4) underlying assets.

We feel that the method presented in this paper and the experimental results thus obtained make it possible to realistically envision the use of multidimensional stochastic models for practical real-world quantitative risk management problems. This capability of computing the joint influences of several tens of risk factors such as interest rates of various terms in different currencies, equity and commodities of various kinds, and any other relevant economic variables, may dramatically increase the competitive advantage of quantitative methods over more traditional analysis techniques. An application of particular interest is the pricing and hedging of complex long-dated commodity and index warrants offered on international OTC markets. We plan to backtest on actual market data the performance of the SSAP method as compared to more classical delta-hedging techniques currently used on capital markets.

References

Barone-Adesi, G. and R. Elliott (1991). Approximations for the Values of American Options. *Stochastic Analysis and Applications*, 9:115–131.

Barone-Adesi, G. and R. E. Whaley (1987). Efficient Analytic Approximation of American Option Values. *Journal of Finance*, 42:301–320.

Barraquand, J. (1993). Numerical Valuation of High Dimensional Multivariate European Securities. Research Report 26, Digital Equipment Corporation, Paris Research Laboratory.

Barraquand, J. and T. Pudet (1994). Pricing of American Path-Dependent Contingent Claims. Research Report 37, Digital Equipment Corporation, Paris Research Laboratory.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Bensoussan, A. (1984). On the Theory of Option Pricing. *Acta Applicandae Mathematicae*, 2:139–158.

Bertsekas, D. P. (1987). *Dynamic Programming, Deterministic and Stochastic Models*. Prentice Hall, Inc., Englewood Cliffs, N. J.

Black, F. and M. Scholes (1973). The Pricing of Options and Corporate Liabilities. *Journal of Political Economics*, vol 81:637–659.

Bossaerts, P. (1989). Simulation Estimators of Optimal Early Exercise. Unpublished manuscript, Graduate School of Industrial Administration, Carnegie Mellon University.

Boyle, P. P. (1977). Options: A Monte Carlo Approach. *Journal of Financial Economics*, 4:323–338.

Boyle, P. P. (1989). The Quality Option and the timing Option in Futures Contracts. *Journal of Finance*, 44:101–113.

Boyle, P. P., J. Evnine, and S. Gibbs (1989). Numerical Evaluation of Multivariate Contingent Claims. *The Review of Financial Studies*, 2:241–250.

Boyle, P. P. and Y. Tse (1990). An Algorithm for Computing Values of Options on the Maximum or Minimum of Several Assets. *Journal of Financial and Quantitative Analysis*, 25-2:215–227.

Brealey, R. A. and S. C. Myers (1991). *Principles of Corporate Finance*. McGraw-Hill. Fourth Edition.

Brennan, M. and E. Schwartz (1978). Finite-Difference Methods and Jump Processes Arising in the Pricing of Contingent Claims: A Synthesis. *Journal of Financial and Quantitative Analysis*, 13:461–474.

Brennan, M. and E. Schwartz (1979). A Continuous Time Approach to the Pricing of Bonds. *Journal of Banking and Finance*, 3:133–155.

Cheng, S. (1987). Pricing Models for Multiple Currency Option Bonds. Unpublished manuscript, Graduate School of Business, Stanford University.

Coppeland (1989). *Valuation*.

Cox, J. and M. Rubinstein (1985). *Options Markets*. Prentice Hall, Englewood Cliffs, NJ.

Cox, J. C., S. A. Ross, and M. Rubinstein (1979). Option Pricing: A Simplified Approach. *Journal of Financial Economics*, vol 7:229–263.

Cummins, J. D. (1988). Risk-Based Premiums for Insurance Guaranty Funds. *Journal of Finance*, 43:823–840.

Doherty, N. A. and J. R. Garven (1986). Price Regulation in Property-Liability Insurance: A Contingent Claim Approach. *Journal of Finance*, 41:1031–1050.

Dothan, M. U. (1987). A Random Volatility Correction for the Black-Scholes Option-Pricing Formula. *Advances in Futures and Options Research*, 2:97–115.

Duffie, D. (1988). *Security Markets: Stochastic Models*. Academic Press, Boston.

Duffie, D. (1992). *Dynamic Asset Pricing Theory*. Princeton University Press.

Fabozzi, F. J. and I. M. Pollack (1987). *The Handbook of Fixed Income Securities*. Dow Jones-Irwin, Homewood, Illinois. Second Edition.

Geske, R. and H. Johnson (1984). The American Put Option Valued Analytically. *Journal of Finance*, 39:1511–1524.

Haber, S. (1970). Numerical Evaluation of Multiple Integrals. *SIAM review*, 12:481–526.

Hammersley, J. M. and D. C. Hanscomb (1964). *Monte Carlo Methods*. Methuen, London.

Harrison, J. M. and D. M. Kreps (1979). Martingales and Arbitrage in Multiperiod Securities Markets. *Journal of Economic Theory*, 20:381–408.

Harrison, J. M. and S. R. Pliska (1981). Martingales and Stochastic Integrals in the Theory of Continuous Trading. *Stochastic Processes and their Applications*, 11:261–271.

Heath, D., R. Jarrow, and A. Morton (1992). Bond Pricing and the Term Structure of Interest Rates: A New Methodology for Contingent Claims Valuation. *Econometrica*.

Hull, J. and A. White (1988). An Analysis of the Bias in Option Pricing Caused by a Stochastic Volatility. *Advances in Futures and Options Research*, 3:29–61.

Hutchinson, J. M. and S. A. Zenios (1991). Financial Simulations on a Massively Parallel Connection Machine. *The International Journal of Supercomputer Applications*, 5-2:27–45.

Itô, K. and H. McKean (1965). *Diffusion Processes and Their Sample Paths*. Springer Verlag, Berlin.

Jaillet, P., D. Lamberton, and B. Lapeyre (1988). Inéquations variationelles et théorie des options. *Comptes rendus de l'Académie des Sciences de Paris*, 307:961–965.

Johnson, H. (1987). Options on the Maximum or the Minimum of Several Assets. *Journal of Financial and Quantitative Analysis*, 22-3:277–282.

Kalos, M. H. and P. A. Whitlock (1986). *Monte Carlo Methods*. John Wiley and Sons. Volume I: Basics.

Karatzas, I. and S. E. Shreve (1988). *Brownian Motion and Stochastic Calculus*. Springer-Verlag.

Karatzas, M. (1988). On the Pricing of American Options. *Applied Mathematics and Optimization*, 17:37–60.

Kraus, A. and S. Ross (1982). The Determination of Fair Profits for the Property-Liability Insurance Firm. *Journal of Finance*, 37:1015–1028.

Kushner, H. J. and P. G. Dupuis (1992). *Numerical Methods for Stochastic Control Problems in Continuous Time*. Springer Verlag.

Mason, S. P. and R. C. Merton (1985). The Role of Contingent Claims Analysis in Corporate Finance. In Altman, E. and M. Subrahmanyan, editors, *Recent Advances in Corporate Finance*. Richard D. Irwin, Inc., Homewood, Ill.

McKean, H. (1965). Appendix: Free Boundary Problem for the Heat Equation Arising from a Problem in Mathematical Economics. *Industrial Management Review*, 6:32–39.

Merton, R. (1973). The Theory of Rational Option Pricing. *Bell Journal of Economics and Management Science*, 4:141–183.

Merton, R. C. (1977). An Analytic Derivation of the Cost of Deposit Insurance and Loan Guarantees: An Application of Modern Option Pricing Theory. *Journal of Banking and Finance*, 1:3–11.

Myneni, R. (1992). The Pricing of American Options. *Annals of Applied Probability*, 2:1–23.

Ross, S. A. (1976). The Arbitrage Theory of Capital Asset Pricing. *Journal of Economic Theory*, 13:341–360.

Schwartz, E. S. and W. N. Torous (1989). Prepayment and the Valuation of Mortgage-Backed Securities. *Journal of Finance*, 44.

Sharpe, W. F. (1985). *Investments*. Prentice-Hall, Englewood Cliffs, New Jersey. Third Edition.

Shimko, D. (1992). The Valuation of Multiple Claim Insurance Contracts. *Journal of Financial and Quantitative Analysis*, 27-2.

Smith, C. (1979). Applications of Option Pricing Analysis. In Bicksler, J., editor, *Handbook of Financial Economics*. North Holland Publ. Co.

Stulz, R. M. (1982). Options on the Minimum or the Maximum of Two Risky Assets: Analysis and Applications. *Journal of Financial Economics*, 10:161–185.

Wiggins, J. B. (1987). Option Values under Stochastic Volatility: Theory and Empirical Estimates. *Journal of Financial Economics*, 19:351–372.

Zaremba, S. K. (1968). The Mathematical Basis of Monte Carlo and Quasi-Monte Carlo Methods. *SIAM Review*, 10:303–314.

PRL Research Reports

The following documents may be ordered by regular mail from:

Librarian – Research Reports
Digital Equipment Corporation
Paris Research Laboratory
85, avenue Victor Hugo
92563 Rueil-Malmaison Cedex
France.

It is also possible to obtain them by electronic mail. For more information, send a message whose subject line is `help` to `doc-server@prl.dec.com` or, from within Digital, to `decprl::doc-server`.

Research Report 1: *Incremental Computation of Planar Maps*. Michel Gangnet, Jean-Claude Hervé, Thierry Pudet, and Jean-Manuel Van Thong. May 1989.

Research Report 2: *BigNum: A Portable and Efficient Package for Arbitrary-Precision Arithmetic*. Bernard Serpette, Jean Vuillemin, and Jean-Claude Hervé. May 1989.

Research Report 3: *Introduction to Programmable Active Memories*. Patrice Bertin, Didier Roncin, and Jean Vuillemin. June 1989.

Research Report 4: *Compiling Pattern Matching by Term Decomposition*. Laurence Puel and Ascánder Suárez. January 1990.

Research Report 5: *The WAM: A (Real) Tutorial*. Hassan Aït-Kaci. January 1990.†

Research Report 6: *Binary Periodic Synchronizing Sequences*. Marcin Skubiszewski. May 1991.

Research Report 7: *The Siphon: Managing Distant Replicated Repositories*. Francis J. Prusker and Edward P. Wobber. May 1991.

Research Report 8: *Constructive Logics. Part I: A Tutorial on Proof Systems and Typed λ-Calculi*. Jean Gallier. May 1991.

Research Report 9: *Constructive Logics. Part II: Linear Logic and Proof Nets*. Jean Gallier. May 1991.

Research Report 10: *Pattern Matching in Order-Sorted Languages*. Delia Kesner. May 1991.

† This report is no longer available from PRL. A revised version has now appeared as a book: "Hassan Aït-Kaci, *Warren's Abstract Machine: A Tutorial Reconstruction. MIT Press, Cambridge, MA (1991).*"

Research Report 11: *Towards a Meaning of LIFE*. Hassan Aït-Kaci and Andreas Podelski. June 1991 (Revised, October 1992).

Research Report 12: *Residuation and Guarded Rules for Constraint Logic Programming*. Gert Smolka. June 1991.

Research Report 13: *Functions as Passive Constraints in LIFE*. Hassan Aït-Kaci and Andreas Podelski. June 1991 (Revised, November 1992).

Research Report 14: *Automatic Motion Planning for Complex Articulated Bodies*. Jérôme Barraquand. June 1991.

Research Report 15: *A Hardware Implementation of Pure Esterel*. Gérard Berry. July 1991.

Research Report 16: *Contribution à la Résolution Numérique des Équations de Laplace et de la Chaleur*. Jean Vuillemin. February 1992.

Research Report 17: *Inferring Graphical Constraints with Rockit*. Solange Karsenty, James A. Landay, and Chris Weikart. March 1992.

Research Report 18: *Abstract Interpretation by Dynamic Partitioning*. François Bourdoncle. March 1992.

Research Report 19: *Measuring System Performance with Reprogrammable Hardware*. Mark Shand. August 1992.

Research Report 20: *A Feature Constraint System for Logic Programming with Entailment*. Hassan Aït-Kaci, Andreas Podelski, and Gert Smolka. November 1992.

Research Report 21: *The Genericity Theorem and the Notion of Parametricity in the Polymorphic λ-calculus*. Giuseppe Longo, Kathleen Milsted, and Sergei Soloviev. December 1992.

Research Report 22: *Sémantiques des langages impératifs d'ordre supérieur et interprétation abstraite*. François Bourdoncle. January 1993.

Research Report 23: *Dessin à main levée et courbes de Bézier : comparaison des algorithmes de subdivision, modélisation des épaisseurs variables*. Thierry Pudet. January 1993.

Research Report 24: *Programmable Active Memories: a Performance Assessment*. Patrice Bertin, Didier Roncin, and Jean Vuillemin. March 1993.

Research Report 25: *On Circuits and Numbers*. Jean Vuillemin. November 1993.

Research Report 26: *Numerical Valuation of High Dimensional Multivariate European Securities*. Jérôme Barraquand. March 1993.

Research Report 27: *A Database Interface for Complex Objects*. Marcel Holsheimer, Rolf A. de By, and Hassan Aït-Kaci. March 1993.

Research Report 28: *Feature Automata and Sets of Feature Trees*. Joachim Niehren and

Andreas Podelski. March 1993.

Research Report 29: *Real Time Fitting of Pressure Brushstrokes*. Thierry Pudet. March 1993.

Research Report 30: *Rollit: An Application Builder*. Solange Karsenty and Chris Weikart. April 1993.

Research Report 31: *Label-Selective λ-Calculus*. Hassan Aït-Kaci and Jacques Garrigue. May 1993.

Research Report 32: *Order-Sorted Feature Theory Unification*. Hassan Aït-Kaci, Andreas Podelski, and Seth Copen Goldstein. May 1993.

Research Report 33: *Path Planning through Variational Dynamic Programming*. Jérôme Barraquand and Pierre Ferbach. September 1993.

Research Report 34: *A Penalty Function Method for Constrained Motion Planning*. Pierre Ferbach and Jérôme Barraquand. September 1993.

Research Report 35: *The Typed Polymorphic Label-Selective λ-Calculus*. Jacques Garrigue and Hassan Aït-Kaci. October 1993.

Research Report 36: *1983–1993: The Wonder Years of Sequential Prolog Implementation*. Peter Van Roy. December 1993.

Research Report 37: *Pricing of American Path-Dependent Contingent Claims*. Jérôme Barraquand and Thierry Pudet. January 1994.

Research Report 38: *Numerical Valuation of High Dimensional Multivariate American Securities*. Jérôme Barraquand and Didier Martineau. April 1994.

38

Numerical Valuation of High Dimensional Multivariate American Securities

Jérôme Barraquand and Didier Martineau

digital

PARIS RESEARCH LABORATORY

85, Avenue Victor Hugo
92563 RUEIL MALMAISON CEDEX
FRANCE