

POLITECNICO DI MILANO
SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE
Corso di Laurea Magistrale in Ingegneria Matematica



Option Pricing in Regime-Switching

Relatore: Prof. Daniele Marazzina

Tesi di laurea di:
Matteo Gardini
Matr. 770105

Anno Accademico 2012 - 2013

*A papà,
ora più che mai.*

Sommario

In questa tesi si considera un mercato, descritto da un modello regime-switching, in cui i parametri che rappresentano l'economia sottostante possano variare casualmente tra un numero finito di stati. Il cambiamento è modellizzato tramite una hidden Markov chain. Saranno studiate equazioni alle derivate parziali per il pricing di opzioni Europee ed Americane. Verranno presentati metodi numerici per il pricing di opzioni Europee quali *FFT*, metodo Monte Carlo e metodo delle differenze finite, e Americane: algoritmo di Longstaff-Schwartz, *SOR* proiettato e metodo delle linee. Si mostrerà come tali metodi, utilizzati generalmente per modelli senza switching sui parametri, possano essere facilmente estesi ai modelli qui presentati.

Abstract

This thesis considers a market, described by a regime-switching model, in which the underlying economy parameters casually switch between a finite number of states. The switching is modeled by a hidden Markov chain. Partial differential equations will be derived and European and American contracts priced. Numerical methods will be developed for European options, *FFT*, Monte Carlo method and finite difference scheme, and for American options: Longstaff-Schwartz algorithm, projected *SOR* and the method of lines. It will be shown that this methods, generally used for models with no switching parameters, can be easily-adapted to the discussed models.

Indice

Introduzione	11
1 Opzioni Europee e opzioni Americane	15
1.1 Teoria dell'arbitraggio	15
1.2 Modello di Black-Scholes e opzioni europee	16
1.3 Opzioni Americane	19
1.4 Processi di Lévy	23
2 Regime-Switching	31
2.1 Catene di Markov	31
2.2 Derivazione del modello	34
2.3 Pricing di opzioni Americane	39
2.4 Regime switching con processi di Lévy	40
2.5 Fatti stilizzati	42
3 Fast Fourier Transform	51
3.1 Approccio neutrale al rischio e option pricing	51
3.2 Algoritmo FFT per l'option pricing	56
4 Metodi Montecarlo	61
4.1 Generalità sul metodo	61
4.2 Metodo Montecarlo per opzioni europee	63
4.3 Algoritmo di Longstaff-Schwartz	69
4.4 Un'ulteriore estensione	73
5 Metodo delle differenze finite	77
5.1 Generalità del metodo ed opzioni Europee	78
5.2 Metodi numerici per le opzioni Americane	89

6	Metodo delle linee	95
6.1	Il metodo	96
6.2	Soluzione numerica delle equazioni	101
6.3	L'algoritmo	105
7	Risultati numerici	111
7.1	Opzioni europee	111
7.2	Opzioni Americane	118
7.3	Un caso più generale	126
7.4	L'influenza del regime-switching	129
	Conclusioni	137
A	Funzione Caratteristica	139
B	Codici	143
B.1	Opzioni Europee	143
B.2	Opzioni Americane	154
B.3	Funzioni Ausiliarie	165

Introduzione

Qualunque realtà, sia essa fisica o emotiva, è determinata dall'interazione di vari fattori o parametri, siano essi osservabili o meno, che insieme concorrono alla determinazione della stessa. Ci si può accorgere nella vita di tutti i giorni come questi fattori siano spesso soggetti a cambiamenti imprevedibili i quali determinano il crearsi di nuove situazioni: basti pensare a come una piccola variazione dei parametri biologici di un individuo possa portarlo da uno stato di salute ad uno di malattia. Un passo fondamentale nella modellistica matematica è non solo cercare di capire quali siano i principi che regolano determinare realtà, ma anche capire quali leggi essi seguano e come possano cambiare dando vita a contesti sempre nuovi oppure ripresentando situazioni già verificatesi in passato. Un primo tentativo di formalizzare matematicamente la possibilità di passare casualmente da uno stato del mondo ad un altro fu intrapreso nel 1906 dal matematico russo Andrei Markov il quale propose un concetto matematico che, pochi decenni più tardi, il mondo conobbe come catene di Markov. Tale strumento si rivelò talmente potente e versatile che trovò applicazione in svariati campi delle scienze applicate: dalla fisica, in cui le catene di Markov sono fondamentali per descrivere il comportamento di parametri tempo-varianti, all'informatica, in cui sono elemento irrinunciabile per molti algoritmi tra cui il PageRank di Google. La genetica stessa ne fa uso per la modellizzazione della deriva genetica. Un ambito in cui l'uso delle catene di Markov si è rilevato particolarmente appropriato è quello della finanza. Nel corso degli ultimi decenni si è cercato di trovare dei modelli che dessero spiegazioni e fondamenti matematici a tutti quei fenomeni propri dell'ambito finanziario, come l'andamento dei mercati o il presentarsi di bolle economiche che, ad occhi inesperti, potrebbero sembrare del tutto casuali. Un primo sforzo per modellizzare l'andamento dei prezzi fu compiuto già nel 1900 da Louis Bachelier che propose un modello che potesse replicare l'andamento della Borsa di Parigi, ma il vero capostipite di tali modelli risale al 1973 quando Fisher Black e Myron Scholes proposero un approccio per il pricing di derivati completamente nuovo che li portò al Nobel nel 1994. Con il passare degli anni però, man mano che l'analisi

delle serie storiche si faceva più accurata, ci si accorse di come il modello di Black e Scholes fosse poco aderente alla realtà. Si iniziò ad osservare come la volatilità implicita non fosse costante, così come proposto dal modello, ma presentasse degli smile e come ci fosse una maggiore probabilità di eventi estremi (fenomeno detto delle code grasse). Inoltre si osservò come eventi estremi negativi fossero più frequenti di quelli positivi e come la dinamica dei prezzi non fosse continua ma spesso presentasse dei veri e propri salti: tutti questi sono elementi che il modello di Black e Scholes, per la sua semplicità, non considerava. Fu così che, a partire dagli anni '80, vennero proposti una serie di modelli più complessi che potessero tenere in considerazione queste evidenze empiriche. Nacquero così i modelli di Lévy come il modello di Merton jump-diffusion (1976), i modelli a volatilità stocastica, tra i quali il modello di Heston (1994), ed i modelli di regime-switching proposti per la prima volta da James Hamilton nel 1989 di cui si occupa questa tesi. I vantaggi derivanti all'utilizzo di questi modelli sono legati alla semplicità della loro interpretazione economica e alla facilità della loro trattazione numerica e matematica. Economicamente, i modelli di regime-switching assumono che siano presenti vari stadi dell'economia (eventualmente solo due, uno di *boom* economico ed uno di *recessione*) ognuno descritto da parametri specifici, che caratterizzano un particolare stato dell'economia o un altro. Da un punto di vista matematico e numerico, essi sono più facilmente trattabili di altri, quali i modelli a volatilità stocastica, perchè non comportano la trattazione di un'equazione differenziale stocastica che descrive la dinamica della volatilità, fatto, tra l'altro, economicamente oscuro come osservato da David e Veronesi. Nella formalizzazione dei regime-switching trovano ampio spazio le citate catene di Markov perchè sono queste ultime che determinano il passaggio tra i vari stati dell'economia. L'alternarsi di periodi di boom economico e di recessione è stato a lungo indagato da grandi economisti, tra cui Karl Marx, Joseph Schumpeter e John Keynes, ed è, alla luce di ciò che sta accadendo negli ultimi anni, una realtà evidente. I modelli di regime-switching, inoltre, prevedono che possano esistere periodi di bassa ed alta volatilità: economicamente si osserva che la volatilità cresce quando si è in una situazione di incertezza di mercato: ciò può verificarsi quando alcuni individui sono in possesso di informazioni non disponibili ad altri, potendo quindi creare speculazione che può portare alla formazione delle così dette bolle finanziarie: in sostanza, quindi, i modelli di regime-switching tengono anche in considerazione la possibilità che possano crearsi bolle speculative. Questo lavoro è strutturato come segue: nel Capitolo 1 spiegheremo cosa si intende per prezzo di un'opzione scritta su un sottostante, analizzando contratti di tipo Europeo e Americano, fornendo gli strumenti teorici necessari per la comprensione della trattazione, e presenteremo il modello di Black e Scholes e i modelli di Lévy. Nel Capitolo 2 presenteremo il modello di regime-switching e deriveremo un sistema di equazioni alle derivate parziali che, una volta risolto, ci permetterà di determinare il prezzo di contratti

Europei ed Americani in ipotesi di regime-switching. Nei capitoli successivi forniremo gli algoritmi per il pricing di contratti plain-vanilla: il metodo della trasformata veloce di Fourier (*FFT*), il metodo Monte Carlo e lo schema alle differenze finite per la soluzione della *PDE*. Deriveremo anche una formula esplicita per il pricing di opzioni Europee in ipotesi di regime-switching che ci permetterà di verificare numericamente la correttezza degli algoritmi presentati. Ci concentreremo anche sugli algoritmi necessari per il pricing di contratti Americani presentando l'algoritmo di Longstaff e Schwartz, il *SOR* proiettato ed il metodo delle linee (*MOL*). Nell'ultimo capitolo confronteremo i vari metodi ed analizzeremo i loro punti forza in termini di accuratezza e costo computazionale, delineando quali algoritmi sia meglio utilizzare per il pricing in ambito di regime-switching. Infine, nell'ultima parte del Capitolo 7, analizzeremo come la possibilità di passare da uno stato all'altro dell'economia influisca sul prezzo dell'opzione e come i modelli di regime-switching possano essere utilizzati in situazioni di mercato reali.

Capitolo 1

Opzioni Europee e opzioni Americane

1.1 Teoria dell'arbitraggio

Si consideri un modello di mercato con due asset: uno privo di rischio B ed uno rischioso S . Supponiamo che due parti debbano stipulare un contratto nel quale una parte si riservi il diritto (ma non l'obbligo) di acquistare il sottostante ad una certa data T , ad un prezzo prefissato all'istante attuale, dietro il pagamento di un premio. Il problema che si pone è : qual è il prezzo “equo” che la controparte deve pagare all'istante attuale per riservarsi questo diritto?

Prima di tutto si deve stabilire cosa si intenda per prezzo “equo”. L'assunzione fondamentale che va fatta è che nel mercato non siano presenti opportunità di arbitraggio: la presenza di possibilità di arbitraggio può essere interpretata come l'esistenza di una strategia di investimento che garantisca, dietro un esborso nullo di denaro, un guadagno certo. La presenza di arbitraggio in un mercato è segno di un errore nei prezzi delle opzioni. Il prezzo equo di un contratto è quel prezzo che esclude qualunque possibilità di arbitraggio.

L'obiettivo è capire sotto quali condizioni un certo modello di mercato sia privo di opportunità di arbitraggio. La letteratura su questo argomento è sterminata, in virtù del fatto che la risposta a questa domanda è notevolmente complessa. Una trattazione più approfondita può essere trovata in [5]. Per dare una risposta a questa domanda occorre preliminarmente dare la seguente

Definizione 1. *In un' economia in cui sono presenti N titoli rischiosi, i cui prezzi sono denotati da S_1, \dots, S_N , un titolo privo di rischio B , e in cui l'evoluzione futura dei titoli rischiosi segue una misura di probabilità “reale”*

\mathbb{P} la misura di martingala equivalente (o neutrale rispetto al rischio) \mathbb{Q} è una misura tale che

- \mathbb{P} è equivalente a \mathbb{Q} .
- I processi dei prezzi scontati, cioè $S_i(t)/B(t)$ con $i = 1, \dots, N$ sono delle martingale rispetto a \mathbb{Q} ovvero $S_i(t) = \mathbb{E}^{\mathbb{Q}}[S_i(T) | \mathcal{F}_t]$ con \mathcal{F}_t σ -algebra.

Allora la condizione necessaria e sufficiente perchè in un mercato non ci siano opportunità di arbitraggio è data dal seguente

Teorema 1 (Primo Teorema Fondamentale dell'Asset Pricing [5, p. 141 Teorema 10.5]). *Il modello è privo di arbitraggio se e solo se esiste una misura di martingala equivalente \mathbb{Q} .*

Quindi se esiste una misura di martingala equivalente, allora il mercato è privo di opportunità di arbitraggio. Il fatto che tale misura si chiami anche misura neutrale al rischio è dovuto al fatto che tutte le attività rischiose abbiano, rispetto a tale misura hanno lo stesso rendimento atteso. Questo è in contrasto con quanto accade usando la misura “reale” rispetto alla quale titoli più rischiosi hanno rendimenti attesi più elevati.

Nella seguente sezione verrà presentato il modello di Black-Scholes proposto per la prima volta nel 1973 da Black, Scholes e Merton [6], diventato il capostipite dei modelli di pricing.

1.2 Modello di Black-Scholes e opzioni europee

Consideriamo un modello di mercato con due asset: uno privo di rischio B ed uno rischioso S le cui dinamiche sono le seguenti:

$$\begin{aligned} dB_t &= rB_t dt, \\ dS_t &= S_t \mu(t, S(t)) dt + S_t \sigma(t, S(t)) d\bar{W}_t, \end{aligned}$$

dove $\mu(t, S(t))$ e $\sigma(t, S(t))$ sono due termini osservabili al tempo t mentre $\bar{W}(t)$ è un moto browniano rispetto alla misura \mathbb{P} . Per semplicità assumiamo che il tasso di interesse privo di rischio r , il drift μ e la volatilità σ siano costanti nel tempo. Il tutto rimane valido qualora r , μ e σ siano funzioni deterministiche del tempo. Il modello diventa allora

$$\begin{aligned} dB_t &= rB_t dt, \\ dS_t &= S_t \mu dt + S_t \sigma d\bar{W}_t. \end{aligned}$$

Consideriamo un'opzione di tipo Europeo, il cui payoff è della forma $\chi = \Phi(S(T))$. Si ricordi che un'opzione di tipo europeo è un contratto che

dà al possessore il diritto ma non l'obbligo di acquistare ad una data pattuita un titolo a un dato prezzo d'esercizio (*strike price*). Si definisca ora un'altra misura di probabilità \mathbb{Q} equivalente a \mathbb{P} tale che sotto di essa S abbia la seguente dinamica

$$dS_t = rS_t dt + S_t \sigma dW_t, \quad (1.1)$$

dove W è un moto browniano rispetto alla misura \mathbb{Q} . E' possibile allora calcolare il prezzo $V(t, S(t))$ di non arbitraggio di un contratto $\Phi(S(T))$ tramite la Formula [5, p. 103 Teorema 7.8]

$$V(t, S(t)) = e^{-r(T-t)} \mathbb{E}^{\mathbb{Q}} [\Phi(S(T)) | \mathcal{F}_t] \quad (1.2)$$

L'interpretazione che può essere data alla Formula (1.2) è la seguente: il prezzo dell'opzione alla data odierna t noto il valore attuale del sottostante $S(t)$ è dato dal valore atteso del payoff finale dell'opzione scontato per il fattore $e^{-r(T-t)}$. E' fondamentale notare come il valore atteso non sia calcolato rispetto alla misura "reale" \mathbb{P} , ma rispetto alla misura di martingala equivalente \mathbb{Q} . E' possibile anche dimostrare che il processo di prezzo $\Pi(t)$ di qualunque titolo è una martingala rispetto a \mathbb{Q} se viene normalizzato con $B(t)$ [5, p. 104 Proposizione 7.9]. Grazie all'esistenza di una misura di martingala equivalente \mathbb{Q} è possibile affermare che il modello di Black-Scholes è privo di arbitraggio.

Nel caso di opzioni europee può essere ricavata una formula chiusa per il prezzo delle opzioni. Vale allora la seguente

Proposizione 1.2.1 ([5, p. 105 Proposizione 7.10]). *Il prezzo di un'opzione Call europea con strike price K e maturity T è dato da*

$$C(t, S_t) = S_t \mathcal{N}(d_1) - e^{-r(T-t)} \mathcal{N}(d_2) \quad (1.3)$$

dove

$$\begin{aligned} d_1 &= \frac{\ln\left(\frac{S_t}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}} \\ d_2 &= d_1 - \sigma\sqrt{T-t} \\ \mathcal{N}(x) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{x^2}{2}} dx \end{aligned}$$

Una formula analoga è valida per la Put il cui valore può essere ricavato dalla seguente relazione nota come *Put-Call parity* ([5, p. 126 Proposizione 9.2])

$$P(t, S(t)) = Ke^{-r(T-t)} + C(t, S(t)) - S(t)$$

In alternativa all'approccio utilizzato per ricavare la Formula (1.3), che è puramente probabilistico si può utilizzare un approccio alle *PDE*. L'idea

generale è presentata di seguito: per una trattazione approfondita si rimanda a [5, p. 115-124] o a [6].

Un contratto, il cui valore si indicherà con χ , si dice replicabile se esiste un portafoglio autofinanziante h tale che

$$V^h(T) = \chi, \quad P - a.s. \quad (1.4)$$

Se ogni contratto è replicabile il mercato si dirà completo.

Si consideri ora il modello di Black-Scholes

$$\begin{aligned} dB_t &= rB_t dt, \\ dS_t &= \mu S_t dt + S_t \sigma d\bar{W}_t, \end{aligned}$$

dove il tasso di interesse r è una costante deterministica. Si può dimostrare [6, p. 116 Teorema 8.3] che il modello di Black-Scholes è completo.

Esistono vari modi per derivare l'equazione di Black-Scholes che permette di calcolare il prezzo di un contratto europeo: l'appoggio seguente ricalca quello dell'articolo originale di Black-Scholes [6]. Si consideri un contratto con payoff dato da $\Phi(S(T))$ il cui processo di prezzo è dato da $f(t, S(t)) = \Phi(S(t))$ e si consideri il portafoglio replicante fittizio così costruito

$$\pi = f - \frac{\partial f}{\partial S} S.$$

Applicando il lemma di Itô alla funzione π si ottiene

$$d\pi = \left(\mu S \frac{\partial f}{\partial S} + \frac{\partial f}{\partial t} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} \right) dt + \frac{\partial f}{\partial S} \sigma S d\bar{W}_t - \frac{\partial f}{\partial S} \mu S dt - \frac{\partial f}{\partial S} \sigma S d\bar{W}.$$

Imponiamo allora che questo portafoglio sia istantaneamente privo di rischio il che equivale a porre, sotto l'ipotesi di assenza di arbitraggio, la condizione

$$d\pi = r d\pi dt$$

che conduce all'equazione seguente

$$\frac{\partial f}{\partial t} + rS \frac{\partial f}{\partial S} + \frac{1}{2} S^2 \sigma^2 \frac{\partial^2 f}{\partial S^2} - rf = 0 \quad (1.5)$$

L'Equazione (1.5) è un'equazione alle derivate parziali (PDE) parabolica all'indietro che unita alla condizione finale $f(T, S(T)) = \Phi(S(T))$ permette di trovare il prezzo del contratto in assenza di arbitraggio.

Si dimostra che $C(t, S_t)$, calcolata come in (1.3), è soluzione dell'equazione (1.5).

1.3 Opzioni Americane

Una opzione Americana fornisce al suo compratore il diritto (ma non l'obbligo) di comprare (opzione call) o vendere (opzione put) ad un prezzo prefissato (detto prezzo d'esercizio) una particolare attività finanziaria, detta sottostante, in cambio di un premio prefissato. Quando l'opzione è di tipo americano, la facoltà di vendere o comprare è esercitabile in un qualunque momento fino alla data di scadenza T (maturity). Di seguito si presentano, per ragioni di completezza, in maniera piuttosto informale le questioni legate al pricing di opzioni Americane. Una trattazione più approfondita può essere trovata in [5, p. 339-348].

Per essere aderenti alla realtà, il possessore dell'opzione americana deve far sì che, per ogni istante temporale t , la decisione di esercitare o meno l'opzione sia basata unicamente sull'informazione disponibile fino a quel momento. Ricordiamo, per chiarezza, che la filtrazione \mathcal{F}_t è interpretabile nel modello come informazione disponibile al tempo t . Questa situazione suggerisce di definire la nozione di *tempo di arresto* come segue

Definizione 2 (Tempo di arresto). *Un tempo di arresto τ rispetto alla filtrazione \mathcal{F}_t è una variabile aleatoria \mathcal{F}_t -misurabile $\forall t \geq 0$.*

Sia il payoff dato da $\Phi(t, x)$, il sottostante S_t e si consideri il seguente problema di *tempo di arresto ottimale*

$$\max_{t \leq \tau \leq T} \mathbb{E} [\Phi(\tau, S_\tau)].$$

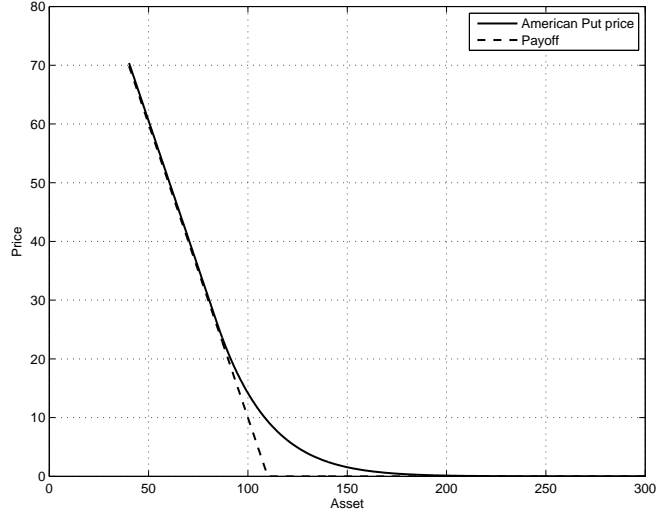
Le opzioni Americane, come detto, differiscono dalle europee in quanto, fissata una data T , possono essere esercitate $\forall \tau \in [t, T]$ dove t è l'istante attuale. Si consideri un'opzione di tipo Americano V^{Am} e si ricordi che il valore del contratto Americano non potrà mai essere minore del corrispettivo Europeo, cioè

$$V^{Am} \geq V^{Eur}.$$

Questo fatto è abbastanza intuitivo se si considera che l'opzione Americana dà al suo possessore maggiori diritti rispetto alla corrispettiva opzione Europea.

Inoltre un'opzione americana ha come valore almeno il suo payoff, condizione necessaria affinché non si verifichino opportunità di arbitraggio. Consideriamo, ad esempio, una Put: se valesse $V^{Am} < (K - S)^+$ si potrebbe acquistare simultaneamente il sottostante e la Put, esercitare la Put immediatamente e vendere il sottostante al prezzo di K così da ottenere $-V + K - S > 0$, cioè un arbitraggio. Si ha quindi, una limitazione inferiore per il prezzo di un'opzione Americana e questo fatto comporta addizionali problemi numerici.

Si consideri ora una Put Americana come illustrata nella Figura 1.1. Si osservi prima la parte a sinistra della figura, cioè per piccoli valori di

Figura 1.1: $V(S, t)$ per una put con $t < T$

$S > 0$ e per qualche $t < T$. La put Americana deve soddisfare la condizione esposta poco sopra cioè $V^{Am} \geq (K - S)^+ \forall (S, t)$. In vista di questo la put deve verificare $V^{Am} = (K - S)^+$ almeno per S piccoli. Per capire cosa succede per valori “medi” di S immaginiamo di approcciarsi da destra cioè dove $V^{Am} > (K - S)^+$. La continuità e la monotonia di V suggeriscono che la curva V^{Am} incontri la linea del payoff per qualche valore di S_f con $0 < S_f < K$. Questo punto di contatto S_f è definito come

$$\begin{aligned} V^{Am} &> (K - S)^+, & \text{se } S > S_f(t), \\ V^{Am} &= (K - S)^+, & \text{se } S \leq S_f(t). \end{aligned} \quad (1.6)$$

Per ogni t , la curva $V^{Am}(S, t)$ si incontra con il suo bordo sinistro nel punto $S_f(t)$. La situazione appena descritta vale per ogni $t < T$ ed il punto di contatto S_f varia con t cioè $S_f = S_f(t)$. La curva $S_f(t)$ separa l'area dove $V > \text{payoff}$ dall'area dove $V = \text{payoff}$. La curva $S_f(t)$ è libera, la sua posizione non è quindi nota a priori. Ecco perchè il problema legato al calcolo del valore di $V^{Am}(S, t)$ per $S > S_f(t)$ si chiama problema di frontiera libera (*free boundary problem*). Per una Call americana il problema è analogo fatta eccezione per il fatto che il contatto con la curva $S_f(t)$ avviene solo se il sottostante distribuisce dividendi.

La notazione $S_f(t)$ è motivata dal procedimento adottato per risolvere la PDE. Tuttavia il significato di $S_f(t)$ è prima di tutto economico. $S_f(t)$ è la **curva di esercizio anticipato**. Si può dimostrare che, tra tutti i possibili tempi di arresto, il tempo di arresto alla curva di esercizio anticipato è ottimale. Questo tempo di arresto ottimale determina il valore dell'opzione

$$V(S, t) = \sup_{t \leq \tau \leq T} \mathbb{E}^{\mathbb{Q}} [e^{-r\tau} \Phi(S_\tau) | S_t = S],$$

dove Φ è il payoff dell'opzione considerata. Chiaramente, il caso delle opzioni Europee è qui incluso, avendo $\tau = T$.

Si osservi in Figura 1.2 che la curva di esercizio anticipato divide il piano in due zone. Se $S > S_f$ l'esercizio anticipato causa una perdita immediata perchè la Condizione (1.6) implica che $-V + K - S < 0$. Ricevere lo stike K non compensa la perdita di S e di V . Per questa ragione il possessore dell'opzione non la eserciterà. L'area in cui $S > S_f$ prende il nome di **regione di continuazione**. Al contrario, dalla parte opposta della frontiera libera S_f , caratterizzata da $V = K - S$ ogni variazione di S è compensata da una variazione corrispondente di V . Qui l'unico modo di creare profitto è esercitare l'opzione ed investire K nel titolo non rischioso per il rimanente periodo $T - t$. Il profitto risultante sarà:

$$Ke^{r(T-t)} - K.$$

Per massimizzare il profitto il possessore dell'opzione dovrà massimizzare $T - t$ e quindi esercitare non appena $V \equiv K - S$ è raggiunta. La curva S_t è la curva di esercizio anticipato e l'area $S \leq S_f$ è detta **regione di arresto**.

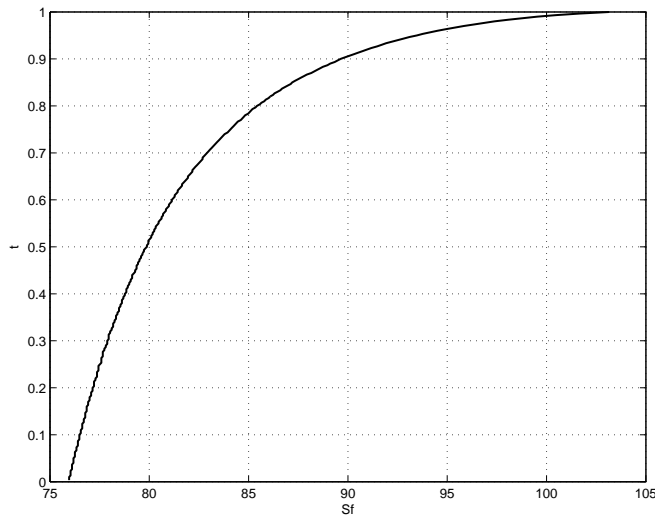


Figura 1.2: Curva di esercizio anticipato per una Put Americana.

Analizziamo ora, in modo più approfondito, il problema della frontiera libera, considerando la solita Put Americana. Per le opzioni Europee la condizione al limite sinistro del dominio è data da $S = 0$. Per le opzioni Americane invece la condizione di sinistra è data dalla curva S_f e quindi, per poter calcolare $S_f(t)$, è necessaria un'ulteriore condizione. Si può dimostrare che, per evitare opportunità di arbitraggio, $V(t, S)$ tocca la funzione payoff tangenzialmente. Inoltre il valore della put $V(t, S)$ è continuo. Si hanno,

dunque, due condizioni

$$V^{Am}(S_t(t), t) = S_f(t) - K, \quad (1.7)$$

$$\frac{\partial V^{Am}(S_f(t), t)}{\partial S} = -1. \quad (1.8)$$

La Condizione (1.7) è detta di *value matching* mentre la Condizione (1.8) è detta di *smooth pasting*.

Nel caso in cui il valore del sottostante evolva seguendo un modello lognormale, per le opzioni di tipo Americano vale la

Proposizione 1.3.1 ([5, p. 344 Proposizione 21.25]). *Se la funzione V è sufficientemente regolare allora valgono le seguenti*

$$\begin{aligned} V(T, S) &= \Phi(T, S), \\ V(t, S) &\geq \Phi(t, S), \quad \forall (t, S), \\ V(t, S) &= \Phi(t, S), \quad (t, S) \in C^c, \\ \left(\frac{\partial}{\partial t} + \mathcal{L}_{Bs}\right) V(t, S) &\leq 0, \quad \forall (t, S), \\ \left(\frac{\partial}{\partial t} + \mathcal{L}_{Bs}\right) V(t, S) &= 0, \quad (t, S) \in C, \end{aligned} \quad (1.9)$$

dove C è la regione di continuazione, C^c è il suo complementare e \mathcal{L}_{Bs} è dato da

$$\mathcal{L}_{Bs}V \doteq \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - \delta) S \frac{\partial V}{\partial S} - rV.$$

Inoltre dato (t, S) il tempo di arresto ottimo τ^* è dato da

$$\tau^* = \inf_{u \geq t} V(u, S_u) = \Phi(u, S_u).$$

Il problema di questo approccio dipende dal fatto che non è nota la frontiera libera. Quindi, non è nemmeno noto in che zona valga l'equazione di Black-Scholes e in che zona valga la disequazione. Un'idea risolutiva è cercare di riscrivere tale problema in maniera analoga senza però far comparire esplicitamente il vincolo che, appunto, non è noto. Vale allora la seguente

Proposizione 1.3.2 (Linear complementary problem) [5, p. 344 Proposizione 21.26]). *In presenza di sufficiente regolarità di V valgono le seguenti*

$$\begin{aligned} V(T, S) &= \Phi(T, S), \\ V(t, S) &\geq \Phi(t, S), \quad \forall (t, S), \\ \left(\frac{\partial}{\partial t} + \mathcal{L}_{Bs}\right) V(t, S) &\leq 0, \quad \forall (t, S), \\ \max \left\{ V(t, S) - \Phi(t, S), \left(\frac{\partial}{\partial t} + \mathcal{L}_{Bs}\right) V(t, S) \right\} &= 0, \quad \forall (t, S), \end{aligned}$$

dove \mathcal{L}_{BS} è dato da

$$\mathcal{L}_{BS}V \stackrel{\text{def}}{=} \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - \delta) S \frac{\partial V}{\partial S} - rV.$$

Il vantaggio di questa riformulazione come insieme di equazioni variazionali è che il vincolo non appare esplicitamente. Tale riformulazione è detta **Problema complementare lineare** (*linear complementary problem*). Per una discussione completa e per le dimostrazioni delle proposizioni precedenti si rimanda a [5, Capitolo 21].

1.4 Processi di Lévy

Osservando i mercati ci si accorge ben presto di quanto il modello di Black-Scholes non riesca a riprodurre efficacemente alcuni fatti empirici osservabili sul mercato reale. Da un'analisi delle serie storiche si nota, ad esempio, come nei prezzi di mercato siano presenti, non di rado, dei salti (Figura 1.3): questo fatto, insieme ad altri che saranno oggetto di discussione, non è preso in considerazione dal modello di Black-Scholes nel quale i prezzi seguono un andamento di tipo lognormale

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t.$$

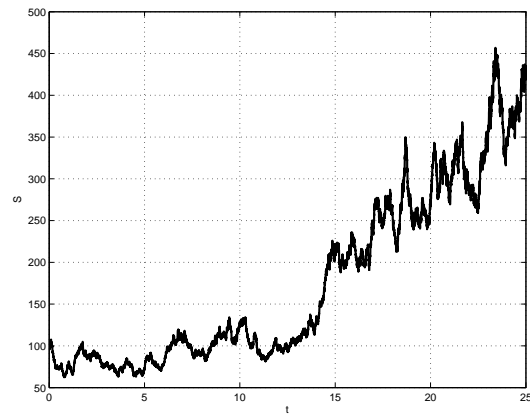
Il processo S viene detto moto browniano geometrico ed ha la proprietà di avere traiettorie continue. I prezzi nel modello di mercato di Black-Scholes condividono con il moto browniano la proprietà di continuità delle traiettorie. Per ovviare a questo fatto si sono introdotti i modelli di Lévy. In questa sezione introdurremo i concetti principali che stanno alla base dei processi di Lévy di cui discuteremo ampiamente in seguito. La teoria è molto complessa e voluminosa. Ci limitiamo unicamente ad un'introduzione allo scopo di fornire una panoramica che possa permettere al lettore di comprendere le linee guida dell'argomento. Per dettagli tecnici e dimostrazioni si rimanda a [12].

Definizione 3 (Processo di Lévy). *Un processo stocastico $(X_t)_{t \geq 0}$ su $(\Omega, \mathcal{F}, \mathcal{P})$ a valori in \mathbb{R}^N tale che $X_0 = 0$ è detto processo di Lévy se:*

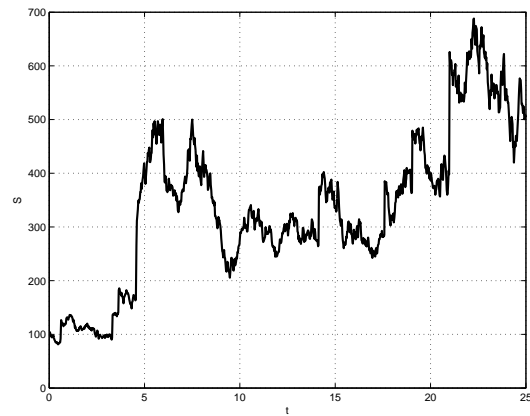
- *Gli incrementi sono indipendenti: per ogni sequenza crescente t_0, \dots, t_n le variabili aleatorie $X_{t_0}, \dots, X_{t_n} - X_{t_{n-1}}$ sono indipendenti.*
- *Gli incrementi sono stazionari cioè la legge di $X_{t+h} - X_t$ non dipende da t .*
- *Vale la continuità stocastica: $\forall \epsilon > 0 \lim_{h \rightarrow 0} \mathbb{P}(|X_{t+h} - X_t| \geq \epsilon) = 0$*



(a) Andamento dei Titolo BorsaItalia



(b) Andamento del Titolo nel modello di Black-Scholes



(c) Andamento del Titolo nel modello di Lévy jump-diffusion

Figura 1.3: Modellizzazione dell'andamento del sottostante.

La terza condizione non implica in alcun modo che le traiettorie siano continue, ma serve ad escludere che i salti possano avvenire in istanti di tempo non casuali.

Si supponga ora che il sottostante, sotto la misura *risk-free*, sia della forma

$$S_t = S_0 \exp \{X_t\},$$

dove X_t è un processo di Lévy che evolve seguendo la dinamica

$$dX_t = \mu^{\mathbb{Q}} dt + \sigma dW_t + dJ_t,$$

dove $\mu^{\mathbb{Q}}$ è il drift rispetto alla misura neutrale al rischio, mentre

$$J_S(B) = \# \{(t, X_t - X_{t-}) \in B\},$$

è una misura aleatoria su $\mathbb{R}^N \times [0, \infty[$ che descrive i salti di S , per ogni insieme misurabile $B \subset \mathbb{R}^N \times [0, \infty[$. Per ogni sottoinsieme misurabile $A \subset \mathbb{R}^N$, $J_S([t_1, t_2] \times A)$ conta il numero di salti di S tra t_1 e t_2 la cui ampiezza sta in A .

Definizione 4. Sia $(X_t)_{t \geq 0}$ un processo di Lévy su \mathbb{R}^N . La misura ν su \mathbb{R}^N è definita da

$$\nu(A) = \mathbb{E} [\# \{t \in [0, 1] : \Delta X_t \neq 0, \Delta X_t \in A\}] \text{ con } A \in \mathfrak{B}(\mathbb{R}^N)$$

è detta misura di Lévy di S : $\nu(A)$ è il valore atteso, per unità di tempo, di salti la cui ampiezza appartiene ad A .

La misura di Lévy per alcuni processi è nota in forma chiusa. Ad esempio nel modello di Merton, dove sono presenti salti distribuiti secondo la legge normale $N \sim (\mu, \delta^2)$, la misura di Lévy assume la forma:

$$\nu(x) = \frac{\lambda}{\delta \sqrt{2\pi}} \exp \left\{ -\frac{(x - \mu)^2}{2\delta^2} \right\}.$$

Definizione 5. Il processo di Lévy $(X_t)_{t \geq 0}$ è detto ad attività finita se

$$\int_{\mathbb{R}^N} \nu(dx) < \infty.$$

Altrimenti si dirà ad attività infinita.

Intuitivamente un processo di Lévy ad attività infinita presenta un “infinito” numero di piccoli salti: la misura di Lévy presenta in tal caso una singolarità nell’origine.

Si può dimostrare che qualsiasi processo di Lévy $(X_t)_{t \geq 0}$ si può decomporre in tre parti: una parte di drift, una parte di moto browniano ed una parte

di salto. Più formalmente si può affermare che esiste un vettore γ , un moto browniano $(B_t)_{t \geq 0}$ con matrice di covarianza A e un processo di puro salto \tilde{X}_t^J tali che:

$$X_t = \gamma t + B_t + \tilde{X}_t^J. \quad (1.10)$$

Ricorda che i termini in (1.10) sono indipendenti. Possiamo inoltre dire che ogni processo di Lévy è completamente determinato dati γ , A e ν , per questo la tripletta (A, ν, γ) si chiama *tripletta caratteristica* del processo X_t . In caso unidimensionale tale tripletta assume la forma:

$$(\sigma^2, \nu, \gamma).$$

Siccome l'andamento del sottostante è modellato come $S_t = \exp\{X_t\}$, dove X_t è un processo di Lévy, il processo S_t si chiama *exponential Lévy*. Di seguito forniremo gli strumenti fondamentali del calcolo stocastico per trattare i processi di Lévy.

1.4.1 Elementi di calcolo stocastico

Nel caso dei processi di Lévy la formula di Itô può essere generalizzata in modo tale che si tenga conto anche della presenza del termine di salto.

Proposizione 1.4.1 (Formula di Itô per processi di Lévy scalari [12, Proposizione 8.15]). Sia $(X_t)_{t \geq 0}$ un processo di Lévy con tripletta caratteristica (σ^2, ν, γ) e sia $f : \mathbb{R} \rightarrow \mathbb{R}$ una funzione C^2 . Allora:

$$f(X_t) = f(0) + \int_0^t \frac{\sigma^2}{2} f''(X_s) ds + \int_0^t f'(X_s) dX_s + \sum_{0 \leq s \leq t, \Delta X_s \neq 0} [f(X_s + \Delta X_s) - f(X_s) - \Delta X_s f'(X_s)] \quad (1.11)$$

Quando si calcola il prezzo di un'opzione generalmente si ha a che fare con valori attesi. Si rivela utile allora decomporre la semimartingala $Y_t = f(X_t)$ in parte di martingala e parte di drift.

Proposizione 1.4.2 (Decomposizione martingala-drift per un processo di Lévy [12, Proposizione 8.16]). Sia $(X_t)_{t \geq 0}$ un processo di Lévy con tripletta caratteristica (σ^2, ν, γ) e sia $f : \mathbb{R} \rightarrow \mathbb{R}$ una funzione C^2 tale che f e le sue derivate siano limitate da una costante C . Allora $Y_t = f(X_t) = M_t + V_t$ dove M è la parte di martingala data da

$$M_t = f(X_0) + \int_0^t f'(X_s) \sigma dW_s + \int_{[0, t] \times \mathbb{R}} \tilde{J}_N(ds, dy) [f(X_{s-} + y) - f(X_{s-})]$$

mentre V_t è dato da

$$V_t = \int_0^t \frac{\sigma^2}{2} f''(X_s) ds + \int_0^t \gamma f'(X_s) dS \\ + \int_{[0,t] \times \mathbb{R}^N} ds \nu(dy) [f(X_{s-} + y) - f(X_{s-}) - y f'(X_{s-}) \mathbb{I}_{|y| \leq 1}]$$

Applicando ora la Formula di Itô (1.11) a $S_t = \exp\{X_t\}$, richiedendo che $\mathbb{E}[|Y_t|] < \infty$, che si può mostrare essere equivalente a richiedere che $\int_{|y| \geq 1} e^y \nu(dy) < \infty$, e riarrangiando i termini di modo da separare la parte di drift e di martingala si ottiene la seguente

Proposizione 1.4.3 ([12, Proposizione 8.20]). *Sia $(X_t)_{t \geq 0}$ un processo di Lévy con tripletta caratteristica (σ^2, ν, γ) che verifichi*

$$\int_{|y| \geq 1} e^y \nu(dy) < \infty.$$

Allora $Y_t = \exp X_t$ è una semimartingala con decomposizione $Y_t = M_t + A_t$ dove la parte di martingala è data da

$$M_t = 1 + \int_0^t Y_{s-} \sigma dW_s + \int_{[0,t] \times \mathbb{R}} Y_{s-} (e^z - 1) \tilde{J}_X(ds, dz),$$

mentre la parte di drift continua è data da

$$A_t = \int_0^t Y_{s-} \left[\gamma + \frac{\sigma^2}{2} + \int_{-\infty}^{+\infty} (e^z - 1 - z - \mathbb{I}_{|z| \leq 1}) \nu(dz) \right] ds.$$

Inoltre Y_t è una martingala se e solo se

$$\gamma + \frac{\sigma^2}{2} + \int_{-\infty}^{+\infty} (e^z - 1 - z - \mathbb{I}_{|z| \leq 1}) \nu(dz) = 0.$$

1.4.2 Pricing di opzioni Europee con processi di Lévy

Si consideri un modello di mercato in cui la dinamica risk-neutral di un sottostante è data dal modello di Lévy esponenziale seguente

$$S_t = S_0 \exp\{X_t\},$$

dove X_t è un processo di Lévy con tripletta caratteristica data da (σ^2, ν, γ) di modo che, sotto una certa misura di martingala equivalente \mathbb{Q} , $e^{-rt}S_t$ sia una martingala. Assumiamo inoltre che

$$\int_{|y| \geq 1} e^{2y} \nu(dy) < \infty \quad (1.12)$$

Questa condizione equivale a richiedere che esistano i momenti di secondo ordine del processo di prezzo. Sotto la dinamica risk-neutral si ha che la dinamica di S_t è data da

$$S_t = S_0 + \int_0^t r S_{u-} du + \int_0^t \sigma S_{u-} dW_u + \int_0^t \int_{-\infty}^{\infty} (e^x - 1) S_{u-} \tilde{J}_X(du, dx)$$

dove \tilde{J}_X denota la misura compensata della misura J_X ([12, Sez. 2.6]).

Il valore di un'opzione Europea con payoff $H(S_T)$ è definito come il valore atteso condizionato scontato del suo payoff sotto la misura di probabilità risk-neutral ovvero

$$C(t, S) = \mathbb{E} \left[e^{-r(T-t)} H(S_T) | S_t = S \right].$$

Differenziando $C(t, S)$ utilizzando le regole di calcolo stocastiche presentate nella precedente sezione e imponendo che il processo sia una martingala otteniamo la seguente equazione *integro-differenziale del secondo ordine alle derivate parziali (PIDE)*

$$\begin{aligned} \frac{\partial C}{\partial t} + rS \frac{\partial C}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 C}{\partial S^2} - rC \\ + \int_{-\infty}^{\infty} \nu(dy) \left[C(t, Se^y) - C - S(e^y - 1) \frac{\partial C}{\partial S} \right] = 0. \end{aligned}$$

Questa equazione è simile a quella derivata in precedenza nel modello standard di Black-Scholes, fatta eccezione per il termine integrale che nella PDE di Black-Scholes non era presente. Notiamo infatti che se il termine di salto non è presente, cioè se $J(x, t) = 0$, si ottiene di nuovo la ben nota equazione di Black-Scholes alle derivate parziali.

La trattazione fin qui svolta ha tralasciato particolari rilevanti che ora andremo a chiarire. Generalmente il payoff non è nel dominio del generatore infinitesimale, dove per generatore infinitesimale si intende L^X definito come

$$L^X f(x) = \gamma \frac{\partial f}{\partial x} + \frac{\sigma^2}{2} \frac{\partial^2 f}{\partial x^2} + \int \nu(dy) \left[f(x+y) - f(x) - y \mathbb{I}_{|y| < 1} \frac{\partial f}{\partial x}(x) \right],$$

e perciò la differenziazione non è possibile. Inoltre il payoff generalmente non è differenziabile. E' possibile mostrare come il prezzo di un'opzione sia

soluzione di viscosità di appropriate *PIDE* con appropriate condizioni al bordo ma questo esula dagli scopi di questa tesi. Per approfondimenti si rimanda a [12, Sezione 12.2]

Assumendo che la funzione payoff $H(S_T)$ sia Lipshitz vale, quindi, la seguente

Proposizione 1.4.4 ([12, Proposizione 12.1]). *Si consideri un modello di Lévy esponenziale $S_t = S_0 \exp\{X_t\}$ dove X_t è un processo di Lévy che verifica la Condizione (1.12). Se $\sigma > 0$ o*

$$\exists \beta \in]0, 2[, \quad \liminf_{\epsilon \rightarrow 0} \epsilon^{-\beta} \int_{-\epsilon}^{\epsilon} |x|^2 \nu(x) > 0$$

allora il valore di un'opzione Europea con scadenza T e payoff $H(S_T)$ è data da $C(t, S)$ dove

$$C : [0, T] \times]0, \infty[\rightarrow \mathbb{R} \\ (t, S) \mapsto C(t, S) = \mathbb{E}[H(S_T) | S_t = S]$$

è continua su $[0, T] \times [0, \infty[$, $C^{1,2}$ su $]0, T[\times]0, \infty[$ e verifica la seguente equazione integro-differenziale alle derivate parziali

$$\begin{aligned} \frac{\partial C}{\partial t} + rS \frac{\partial C}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 C}{\partial S^2} - rC \\ + \int_{-\infty}^{\infty} \nu(dy) \left[C(t, Se^y) - C - S(e^y - 1) \frac{\partial C}{\partial S} \right] = 0. \end{aligned}$$

Risolvendo tale equazione è possibile quindi ottenere il prezzo di un'opzione Europea nelle ipotesi in cui il sottostante evolva secondo un modello exponential Lévy con tripletta caratteristica (σ^2, ν, γ) .

1.4.3 Pricing di opzioni americane con processi di Lévy

Consideriamo ora il problema di prezzare un'opzione Americana in presenza di un sottostante che segue una dinamica dettata da un processo di Lévy. Considerando i risultati delle precedenti sezioni e quanto discusso in [4], si può provare la seguente

Proposizione 1.4.5. *Sia $V : [0, T[\times]0, \infty[$ l'unica soluzione continua e limitata di*

$$\begin{aligned}
V(T, S) &= \Phi(T, S), \\
V(t, S) &> \Phi(t, S) \quad \forall (t, S), \\
V(t, S) &= \Phi(t, S) \quad (t, S) \in C^c, \\
\left(\frac{\partial}{\partial t} + L^S\right) V(t, S) &\leq 0 \quad \forall (t, S), \\
\left(\frac{\partial}{\partial t} + L^S\right) V(t, S) &= 0 \quad (t, S) \in C,
\end{aligned}$$

dove $\sigma > 0$ e

$$\begin{aligned}
L^S V(t, S) &= rS \frac{\partial V}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 V}{\partial S^2} \\
&\quad + \int_{-\infty}^{+\infty} \nu(dy) \left[V(t, Se^y) - V - S(e^y - 1) \frac{\partial V}{\partial S} \right] - rV,
\end{aligned}$$

allora il valore dell'opzione Americana è dato da

$$V_t = V(t, S_t).$$

Tale problema, analogamente a quanto mostrato nel caso di Black-Scholes, può essere espresso nella forma di linear complementary problem come mostrato nella seguente proposizione

Proposizione 1.4.6 (Linear complementary problem [12, Proposizione 12.3]). *Sia $V : [0, T[\times]0, \infty[$ l'unica soluzione continua e limitata di*

$$\begin{aligned}
V(T, S) &= \Phi(T, S), \\
V(t, S) &> \Phi(t, S) \quad \forall (t, S), \\
\left(\frac{\partial}{\partial t} + L^S\right) V(t, S) &\leq 0 \quad \forall (t, S), \\
\left\{ \frac{\partial V}{\partial t} + L^S V \right\} \{V - \Phi(T, S)\} &= 0 \quad \text{su } [0, T] \times]0, \infty[\text{ e } \forall S \in]0, \infty[,
\end{aligned}$$

con $\sigma > 0$ e L^S definito come nella Proposizione 1.13.

Allora il valore dell'opzione Americana è dato da

$$V_t = V(t, S_t).$$

Se V fosse una funzione $C^{1,2}$ la proposizione precedente potrebbe essere dimostrata con una semplice applicazione della formula di Itô [25]. Una prova rigorosa richiede invece il principio di programmazione dinamica e può essere trovata in [34, 35]. Ancora una volta si può notare che, qualora $J_S(y, t) = 0$, si ricadrebbe al caso Black-Scholes.

Capitolo 2

Regime-Switching

In questo capitolo svilupperemo il modello di regime-switching per il pricing di opzioni. Il modello standard di Black-Scholes non risulta adeguato ad essere utilizzato nell'economia reale, le assunzioni lo portano, infatti, ad essere un'eccessiva semplificazione della realtà. Basti pensare al fatto che esso prevede la continuità delle traiettorie dei prezzi mentre, nella realtà, spesso si osservano dei veri e propri salti nella dinamica dei prezzi: abbiamo dunque bisogno di un nuovo modello che replichi questa ed altre proprietà tipiche delle serie storiche. Nella prima parte del capitolo introdurremo brevemente il concetto di catena di Markov a tempo continuo, necessario per regolare il passaggio previsto dal regime-switching da uno stato dell'economia all'altro, ed i principali risultati ad esso collegati. Presenteremo poi il modello di regime-switching sia nel caso standard di Black-Scholes sia nel caso dei più generali modelli di Lévy. Da ultimo osserveremo come le varie regolarità presenti nelle serie storiche siano o meno tenute in considerazione dai vari modelli studiati.

2.1 Catene di Markov

Un processo stocastico markoviano o processo di Markov è un processo stocastico nel quale la probabilità di transizione che determina il passaggio ad uno stato di sistema dipende unicamente dallo stato di sistema immediatamente precedente (proprietà di Markov) e non dal come si è giunti a tale stato (in quest'ultima ipotesi si parla di processo non markoviano). Tale processo fu introdotto per la prima volta dal matematico russo Andrei Andreevic Markov (1856-1922) che per primo ne sviluppò la teoria. Nei nostri modelli assumeremo che i parametri cambino seguendo un processo di tipo markoviano. Di seguito introdurremo solo gli aspetti necessari alla comprensione dell'argomento: una trattazione completa è fornita da [22].

2.1.1 Introduzione

Consideriamo uno spazio di probabilità $(\Omega, \mathcal{F}, \mathbb{P})$ ed un insieme di stati M finito o numerabile che sarà considerato come spazio misurabile munito della σ -algebra $\mathcal{P}(M)$ di tutti i sottoinsiemi di M .

Definizione 6. Una famiglia di variabili aleatorie $(X_t)_{t \geq 0}$ a valori in M è detta catena di Markov se

$$\mathbb{P} = (X_{t_{n+1}} = j | X_{t_n} = i_n, \dots, X_{t_1} = i_1) = \mathbb{P} (X_{t_{n+1}} = j | X_{t_n} = i_n)$$

per ogni n , $i_1, i_2, \dots, i_n \in M$, $t_1 < \dots < t_n$.

Tale proprietà è detta proprietà di Markov: inoltre considereremo catene di Markov continue da destra. La probabilità di transizione nell'intervallo $[s, t]$ da uno stato i ad uno stato j con $i, j \in M$ è definita da

$$p_{ij}(s, s+t) = \mathbb{P}(X_{t+s} = j | X_s = i)$$

Se poi, per ogni $i, j \in M, t, s \geq 0$, vale che $p_{ij}(s, t+s) = p_{ij}(0, t)$ allora la catena si dirà omogenea. Prenderemo in considerazione solo catene omogenee. Ricordiamo, inoltre, per completezza che una matrice stocastica è una matrice $A \in \mathbb{R}^{n \times n}$ tale che

$$\begin{aligned} a_{ij} &\geq 0, \\ \sum_{j=1}^n a_{ij} &= 1, \forall i = 1, \dots, n. \end{aligned}$$

dove l'elemento A_{ij} rappresenta proprio la probabilità di transizione dallo stato i allora stato j .

2.1.2 Tassi di transizione

Fondamentale nello studio delle catene di Markov a tempo continuo è l'introduzione dei così detti tassi di transizione. Indichiamo con $p_{ij}(t)$ la probabilità di transizione $p_{ij}(0, t)$. La famiglia di matrici $(P_t)_{t \geq 0}$, eventualmente infinite, tali che

$$(P_t)_{ij} = p_{ij}(t)$$

è detto semigruppato Markoviano. Indichiamo con \mathbb{I} la matrice $(\delta_{ij})_{i,j \in M}$ con $\delta_{ij} = 1$ se $i = j$ e $\delta_{ij} = 0$ se $i \neq j$.

Si può dimostrare [17, Teorema 1.1] che il semigruppato $(P_t)_{t \geq 0}$ è tale che $P_0 = \mathbb{I}$, che la matrice $(p_{ij}(t))_{t \geq 0}$ è una matrice stocastica per ogni $t \geq 0$ e che vale l'equazione detta di Chapman-Kolmogorov $P_{t+s} = P_t P_s$ per ogni $t, s \geq 0$. La maggior parte delle informazioni su una catena di Markov a tempo continuo si possono estrarre dal semigruppato associato, il quale determina

le leggi finito dimensionali. Infatti per ogni $n \geq 1$, per ogni $t_1 < t_2 < \dots < t_n$ e per $i_0, i_1, \dots, i_n \in M$ si ha che

$$\mathbb{P}(X_{t_n} = i_n, \dots, X_{t_1} = i_1, X_0 = i) = p_{i_0 i_1}(t_1) p_{i_1 i_2}(t_2 - t_1), \dots, \\ p_{i_{n-1} i_n}(t_n - t_{n-1}).$$

D'ora in avanti considereremo solo catene di Markov continue in probabilità. Ricordiamo che dato un tempo t , il processo stocastico $(X_t)_{t \geq 0}$ è detto continua in probabilità in t se

$$\mathbb{P}\left(\omega \in \Omega \left| \lim_{s \rightarrow t} |X_s(\omega) - X_t(\omega)| = 0 \right. \right) = 1.$$

Se la catena di Markov $(X_t)_{t \geq 0}$ è continua in probabilità allora si ha come conseguenza [17, Proposizione 1.2] che le funzioni $t \mapsto p_{ij}(t)$ sono continue. Possiamo ora enunciare il seguente teorema

Teorema 2. [17, Teorema 1.3] *Se le probabilità di transizione sono continue allora esistono i limiti*

$$q_{ij} = \lim_{t \rightarrow 0^+} \frac{p_{ij}(t)}{t}, \text{ se } i \neq j, \quad q_{ii} = \lim_{t \rightarrow 0^+} \frac{p_{ii}(t) - 1}{t},$$

e risulta $0 \leq q_{ij} < \infty$ per $i \neq j$ e $q_{ii} \leq 0$ (eventualmente $q_{ii} = -\infty$). Se l'insieme degli stati M è finito allora risulta che $q_{ii} > -\infty$ e valgono le equazioni

$$\frac{d}{dt} p_{ij}(t) = \sum_{k \in S} p_{ik}(t) q_{kj}, \quad p_{ij}(0) = \delta_{ij}, \\ \frac{d}{dt} p_{ij}(t) = \sum_{k \in S} q_{ik} p_{kj}(t), \quad p_{ij}(0) = \delta_{ij}.$$

I numeri q_{ij} si chiamano tassi di transizione.

Integrando una delle due equazioni differenziali, note rispettivamente come equazioni di Kolmogorov in avanti e all'indietro, è possibile determinare le probabilità di transizione, almeno nel caso in cui l'insieme degli stati M sia finito.

E' facile verificare che la condizione $\sum_{j \in S} p_{ij}(t) = 1$ equivale a $\sum_{j \in S} q_{ij} = 0$. Nel caso in cui M sia infinito occorre essere più cauti perchè, anche se dovesse essere $q_{ii} > -\infty$, non è garantito che la soluzione delle equazioni di Komogorov sia unica. La matrice $Q = (q_{ij})_{i,j \in M}$ si chiama generatore del semigruppato markoviano $(P(t))_{t \geq 0}$. Possiamo notare che, grazie alla differenziabilità di $p_{ij}(t)$ e all'omogeneità della catena di Markov, la probabilità di passare da uno stato i ad uno stato j , in un intervallo $[t, t+h]$, è data da

$$\mathbb{P}(X_{t+h} = j | X_t = i) = \begin{cases} p_{ij}(h) = q_{ij}h + o(h), & \text{se } i \neq j \\ p_{ii}(h) = 1 + q_{ii}h + o(h), & \text{se } i = j \end{cases},$$

con $o(h)$ infinitesimo di ordine superiore ad h .

La nozione di tassi di transizione risulta più chiara una volta introdotto il tempo di prima uscita da uno stato $i \in M$, definito come

$$T_i(\omega) = \inf_{t \geq 0} X_t(\omega) \neq i,$$

con la convenzione tecnica che $\inf_{\emptyset} = +\infty$. Intuitivamente il tempo di prima uscita dallo stato $i \in M$ è il primo istante in cui il processo X_t abbandona lo stato i . Le T_i sono variabili aleatorie su $[0, +\infty]$, cioè misurabili rispetto opportune σ -algebre. Si può dimostrare [17, Teorema 2.1] che se $-\infty < q_{ii} < 0$ allora la variabile aleatoria T_i ha legge esponenziale di parametro $-q_{ii}$. E' possibile verificare inoltre che [17, Teorema 2.2], uscendo da uno stato i , la catena di Markov passa nello stato j con probabilità $q_{ij}/(-q_{ii})$. Notiamo che maggiore sarà q_{ii} in valore assoluto e minore sarà il tempo di soggiorno del processo in quello stato.

Possiamo ora utilizzare le catene di Markov a tempo continuo per regolare il passaggio da uno stato di mercato all'altro del nostro modello di regime-switching. Nella seguente sezione prenderemo in considerazione il modello di Black-Scholes e i modelli di Lévy in presenza di regime-switching: deriveremo per entrambe i modelli metodi di pricing sia per le opzioni Europee che per le opzioni Americane.

2.2 Derivazione del modello

Seguendo l'idea presentata in [7], consideriamo un modello di mercato con un bond ed un titolo rischioso e ci si ponga nel modello di Black-Scholes. Ipotizziamo, inoltre, che la volatilità, il drift ed il tasso risk-free dipendano dallo stato dell'economia. Ciò può essere modellizzato come una catena di Markov continua con un numero finito di stati. Si indichi questa catena con $X = \{X_t, t \geq 0\}$. Chiaramente potrebbero esserci anche solo due stati: uno di certezza ed uno di incertezza oppure uno di mercato positivo, caratterizzato da alto rendimento e bassa volatilità, ed uno di mercato negativo in cui si ha basso rendimento ed alta volatilità. In particolare lo spazio degli stati X può essere preso come l'insieme dei vettori unitari $\{e_1, e_2, \dots, e_N\}$, $e_i = (0, \dots, 0, 1, 0, \dots, 0)' \in \mathbb{R}^N$.

Si supponga che il processo X sia omogeneo nel tempo ed abbia come matrice dei tassi di transizione la matrice Q . Allora se $P_t = \mathbb{E}[X_t] \in \mathbb{R}^N$

$$\frac{dp_t}{dt} = Qp_t.$$

Si può mostrare inoltre che

$$X_t = X_0 + \int_0^t QX_u du + M_t,$$

dove $M = \{M_t, t \geq 0\}$ è una martingala rispetto alla filtrazione generata da X .

Il modello considerato ha due titoli scambiabili: un titolo privo di rischio ed un titolo rischioso. Si considerino i vettori $\mu = (\mu_1, \dots, \mu_N)'$, $r = (r_1, \dots, r_N)'$, e $\sigma = (\sigma_1, \dots, \sigma_N) \in \mathbb{R}^N$ tali che

$$\begin{aligned}\mu_t &= \langle \mu, X_t \rangle, \\ \sigma_t &= \langle \sigma, X_t \rangle, \\ r_t &= \langle r, X_t \rangle,\end{aligned}$$

dove $\langle \cdot, \cdot \rangle$ denota l'usuale prodotto scalare in \mathbb{R}^N . Il bond privo di rischio e il titolo rischioso seguono le dinamiche

$$\begin{aligned}dB_t &= r_t B_t, \\ dS_t &= \mu_t S_t dt + \sigma_t S_t dW_t,\end{aligned}$$

dove $W = \{B_t, T \geq 0\}$ è un moto browniano su $(\Omega, \mathcal{F}, \mathcal{P})$. Sia $\{\mathcal{F}_t\}$ la filtrazione generata da S .

La catena di Markov a tempo continuo, passando da uno stato all'altro, fa sì che σ, r, μ non siano più deterministici ma dipendano dallo stato in cui si trova la catena di Markov in un determinato istante. Seguendo questo procedimento si introduce un'ulteriore fonte di casualità che rende il mercato incompleto. Tale problema può essere risolto introducendo delle Arrow-Debreu security legate al costo dello switch [18, p. 6]. Assumiamo ora di lavorare sotto la misura neutrale al rischio \mathbb{Q} così che, ogni contratto Europeo di valore Π , con data di esercizio T , possa essere prezzato come:

$$\Pi = \mathbb{E}^{\mathbb{Q}} \left[e^{-\int_t^T r_u du} g(S_T) \middle| \mathcal{F}_t \right],$$

dove $g(S_T)$ è il payoff. D'ora in avanti il valore atteso è da intendersi sempre rispetto alla misura risk-free \mathbb{Q} . Per un call Europea al tempo $t \in [0, T]$ vale

$$C(t, T, S_t, X_t) = \mathbb{E} \left[e^{-\int_t^T r_u du} (S_T - K)^+ \middle| \mathcal{F}_t \right].$$

Supponiamo che al tempo $t \in [0, T]$, $S_t = S$ e che $X_t = X$ allora il prezzo di un'opzione call Europea con data di esercizio T e strike K è data da

$$C(t, T, S_t, X_t) = \mathbb{E} \left[e^{-\int_t^T r_u du} (S_T - K)^+ \middle| S_t = S, X_t = X \right].$$

Poniamo

$$V(t, S, X) = e^{-\int_0^t r_u du} C(t, T, S, X) = \mathbb{E} \left[e^{-\int_0^T r_u du} (S_T - K)^+ \middle| \mathcal{G}_t \right],$$

dove $G_t = \sigma \{S_u, X_u : u \leq t\}$. Di conseguenza V è una \mathcal{G}_t -martingala. Poniamo

$$\mathbf{V}(t, S) = (V(t, S, e_1), \dots, V(t, S, e_N))$$

così che $V(t, S_t, X_t) = \langle \mathbf{V}(t, S_t), X_t \rangle$. Applicando la formula di Itô a V si ottiene

$$\begin{aligned} dV &= \frac{\partial V}{\partial t} dt + \frac{\partial V}{\partial S} (r_t S_t dt + \sigma_t S_t dW_t) \\ &\quad + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma_t^2 S_t^2 dt + \langle \mathbf{V}, X_t \rangle, \end{aligned} \quad (2.1)$$

e $dX_t = QX_t dt + dM_t$. Per definizione V è una martingala (si sta operando in ipotesi di misura neutrale al rischio) e allora il drift nell'Equazione (2.1) deve essere nullo. Ne segue che:

$$\frac{\partial V}{\partial t} + rS \frac{\partial V}{\partial S} + \frac{1}{2} \sigma_t^2 S^2 \frac{\partial^2 V}{\partial S^2} + \langle \mathbf{V}, QX \rangle = 0$$

Ora, ricordando che $V = \exp \left(-\int_0^t r_u du \right) C$, si ha che

$$\exp \left(-\int_0^t r_u du \right) \left(-r_t C + \frac{\partial C}{\partial t} + r_t S \frac{\partial C}{\partial S} + \frac{1}{2} \sigma_t^2 S^2 \frac{\partial^2 C}{\partial S^2} + \langle \mathbf{C}, QX \rangle \right) = 0, \quad (2.2)$$

dove si è posto $\mathbf{C}(t, S) = (C(t, S, e_1), \dots, C(t, S, e_N))$. Come condizione finale si ha invece che

$$C(T, T, S, X) = (S_T - K)^+$$

L'Equazione (2.2) si riduce allora ad N equazioni con $X = e_1, e_2, \dots, e_N$. Posto allora

$$\begin{aligned} r_t &= \langle r, X_t \rangle = r_i, \\ \sigma_t &= \langle \sigma, X_t \rangle = \sigma_i, \\ \mu_t &= \langle \mu, X_t \rangle = \mu_i, \end{aligned}$$

e scrivendo

$$C_i = C(t, T, S, e_i), \quad \mathbf{C} = (C_1, C_2, \dots, C_N),$$

si ha che \mathbf{C} soddisfa il seguente sistema di equazioni di Black-Scholes accoppiate

$$-r_i C_i + \frac{\partial C_i}{\partial t} + r_i S \frac{\partial C_i}{\partial S} + \frac{1}{2} \sigma_i^2 S^2 \frac{\partial^2 C_i}{\partial S^2} + \langle \mathbf{C}, Q e_i \rangle = 0, \quad i = 1, \dots, N,$$

con condizione finale

$$C(T, T, S, e_i) = (S_T - K)^+.$$

Risolvendo questo sistema di equazioni alle derivate parziali è possibile determinare il prezzo di N call Europee ciascuna con stato di partenza $i \in N$, in presenza di regime-switching tra N stati.

Semplificando le ipotesi del modello è possibile inoltre trovare una formula chiusa di pricing seguendo quanto proposto in [10]. Supponiamo che il tasso di interesse privo di rischio sia costante: questa assunzione non solo semplifica di molto la trattazione, ma è anche accettabile da un punto di vista finanziario, difatti la volatilità del tasso risk-free è molto inferiore a quella dei rendimenti. Supponiamo, per semplicità, che ci siano solo due stati e seguiamo l'approccio di Guo [18] e di Chang-Der [10]. Supponiamo che, sotto la misura "reale" \mathbb{P} , il sottostante evolva seguendo la dinamica

$$dS_t = S_t \mu_t dt + S_t \sigma_t dW_t. \quad (2.3)$$

Denotiamo con λ_i il tasso di transizione dallo stato i al tempo τ_i . Abbiamo allora che

$$P(\tau_i > t) = e^{-\lambda_i t} \quad i = 0, 1.$$

Supponiamo inoltre che il generatore della catena di Markov sia dato da

$$Q = \begin{bmatrix} -\lambda_1 & \lambda_1 \\ \lambda_2 & -\lambda_2 \end{bmatrix} \quad \lambda_1, \lambda_2 \geq 0$$

Come osservato da Harrison e Pliska in [21], tale modello di mercato non è completo. Un modo per completarlo è supporre l'esistenza di una security sul mercato che paga un'unità di denaro al prossimo tempo τ in cui la catena di Markov cambierà stato. Sotto questa ipotesi il mercato risulta completo ed esiste un unico prezzo per le opzioni scritte sul sottostante S_t . Sotto la misura di martingala neutrale al rischio abbiamo che la dinamica del sottostante è data da:

$$dS_t = S_t r dt + S_t \sigma_t dW_t.$$

Sia $\epsilon(t) = 0, 1$ lo stato della catena di Markov. Indichiamo con T_i il tempo totale tra 0 e T nel quale $\epsilon(0) = 0$ partendo da $i = 0, 1$. Sia $\sigma(i)$ la volatilità associata allo stato i , $\mu(i)$ il rendimento del titolo rischioso nello stato del mondo i e, infine, sia $f_i(t, T)$ la distribuzione di probabilità di T_i .

Proposizione 2.2.1 ([10, Teorema 1]). *Il prezzo privo di arbitraggio di una Call Europea nel modello (2.3) con scadenza T e strike price K è data da*

$$C_i(S_t, t) = \mathbb{E} \left[e^{-rT} (S_T - K)^+ \mid \epsilon(0) = i \right] = e^{-rT} \int_0^\infty \int_0^T \frac{y}{y+K} \rho(\log(y+K), m(t), v(t)) f_i(t, T) dt dy$$

dove $\rho(x, m(t), v(t))$ è la distribuzione normale con media $m(t)$ e varianza $v(t)$ date da

$$m(t) = \log(S_0) + \left(rT - \frac{1}{2}v(t) \right) \\ v(t) = \left(\sigma(0)^2 - \sigma(1)^2 \right) t + \sigma(1)^2 T$$

e dove

$$\rho(x, m(t), v(t)) = \frac{1}{\sqrt{2\pi v(t)}} \exp \left\{ -\frac{(x - m(t))^2}{2v(t)} \right\}$$

$$f_0(t, T) = e^{-\lambda_1 T} \delta_0(T-t) + e^{-\lambda_2(T-t) - q_1 t} \left[\lambda_1 I_0 \left(2(\lambda_1 \lambda_2 t (T-t))^{1/2} \right) + \left(\frac{\lambda_1 \lambda_2 t}{T-t} \right)^{1/2} I_1 \left(2(\lambda_1 \lambda_2 t (T-t))^{1/2} \right) \right] \\ f_1(t, T) = e^{-\lambda_2 T} \delta_0(t) + e^{-\lambda_2(T-t) - \lambda_1 t} \left[\lambda_1 I_0 \left(2(\lambda_1 \lambda_2 t (T-t))^{1/2} \right) + \left(\frac{\lambda_1 \lambda_2 t}{T-t} \right)^{1/2} I_1 \left(2(\lambda_1 \lambda_2 t (T-t))^{1/2} \right) \right]$$

dove I_0 e I_1 sono le funzioni modificare di Bessel tali che

$$I_a(z) = \left(\frac{z}{2} \right)^a \sum_{k=0}^{\infty} \frac{(z/2)^{2k}}{k! \Gamma(k+a+1)!}.$$

Osserviamo che quando $\mu(0) = \mu(1)$ e $\sigma(0) = \sigma(1)$ allora i termini $m(t)$ e $v(t)$ sono indipendenti dal tempo e ci si riduce al caso di Black-Scholes. In Tabella 2.1 riportiamo i prezzi ottenuti dall'integrazione diretta confrontati con quelli ottenuti tramite metodo Monte Carlo, *FFT* e differenze finite, presentati più avanti, per un'opzione Call Europea con regimi di partenza $i = 1, 2$. Si è scelto $S_0 = 100, K = 100, r = 0.1, \sigma_0 = 0.3, \sigma_1 = 0.2, T = 1, \lambda_1 = 1, \lambda_2 = 1$.

Nel caso in cui il numero di stati sia $N > 2$ la trattazione diventa molto complessa. Un approccio alternativo che permette di ottenere una formula approssimata per $M = \{0, 1, \dots, N\}$ viene sviluppato in [10].

<i>Stato</i>	<i>DI</i>	<i>FFT</i>	<i>PDE</i>	<i>MC</i>	<i>IC_{α=0.5}</i>
<i>i</i> = 1	15.8138	15.8138	15.8132	15.8122	[15.6764 15.9479]
<i>i</i> = 2	14.3172	14.3172	14.3164	14.2793	[14.1641 14.3945]

Tabella 2.1: Metodi di pricing per le Europee a confronto

2.3 Pricing di opzioni Americane

Quando detto fin qui è valido per le opzioni Europee. Seguendo le linee guida presentate nella sezione precedente per le opzioni Europee ed i risultati della Sezione 1.3 è possibile estendere la trattazione per le opzioni Americane nel modello di Black-Scholes in ipotesi di regime-switching, per poi essere esteso ai modelli di Lévy. Qui di seguito presentiamo i principali teoremi mentre per una trattazione completa si rimanda a [24].

Proposizione 2.3.1 (Opzione americana nel modello di Black-Scholes con regime-switching [23, p. 2-3]). *Sia $M = \{1, \dots, N\}$ il numero di stati della catena di Markov e sia $P : [0, T[\times]0, \infty[$ l'unica soluzione continua e limitata di*

$$\begin{aligned} \frac{\partial P_i}{\partial t} + L^S P_i - r_i P_i + \langle \mathbf{P}, Qe_i \rangle &\leq 0, \\ P_i - (K - S)^+ &\geq 0, \\ \left\{ \frac{\partial P_i}{\partial t} + L^S P_i - r_i P_i + \langle \mathbf{P}, Qe_i \rangle \right\} \{P_i - (K - S)^+\} &= 0, \\ \text{su } [0, T] \times]0, \infty[\text{ e } \forall S \in]0, \infty[, \\ P(T, S) &= (K - S)^+, \end{aligned}$$

dove $\sigma > 0$ e

$$L^S P(t, S) = r_i S \frac{\partial P}{\partial S} + \frac{\sigma_i^2 S^2}{2} \frac{\partial^2 P}{\partial S^2},$$

allora il valore della Put Americana, con stato di partenza i , è dato da

$$P_i(t) = P_i(t, S_t).$$

Grazie a questo teorema possiamo osservare che, nel caso in cui il sottostante evolva secondo il modello di Black-Scholes, in ipotesi di regime switching tra N stati, il prezzo dell'opzione Put Americana con regime di partenza i si ottiene risolvendo un sistema di N equazioni alle derivate parziali accoppiate. Il vettore soluzione di questo sistema, indicato con \mathbf{P} , conterrà i valori delle Put P_i , dove P_i è il valore della put con volatilità σ_i e tasso risk-free r_i di partenza dal regime i -esimo. Un'analogia formulazione vale per il pricing di una Call Americana che distribuisca dividendi continui δ . Sarà sufficiente porre $C(T, S) = (S - K)^+$ e sostituire r_i con $r_i - \delta_i$. Nel caso in

cui la Call Americana non distribuisca dividendi, il suo prezzo sarà pari a quello della rispettiva Call Europea ([5, p. 345-346]). Quest'ultimo risultato ci permetterà di testare molti metodi numerici proposti, come vedremo nei capitoli successivi.

2.4 Regime switching con processi di Lévy

Il modello classico di Black-Scholes con regime switching, come visto, può essere arricchito assumendo che il sottostante possa presentare dei salti nella sua dinamica. Per chiarezza espositiva considereremo il caso in cui i parametri interessati dal passaggio di stato siano solo quelli relativi parte di diffusione e trasporto mentre quelli propri della parte di salto siano costanti. L'estensione al caso in cui quest'ultimi cambino con il cambiare dello stato dell'economia può essere considerata facilmente basandosi sull'indipendenza tra il processo di salto e il processo di Markov che regola il passaggio di stato. All'atto pratico, tutte le proposizioni enunciate di seguito restano valide a patto di sostituire $\nu(x)$ con $\nu_i(x)$.

2.4.1 Opzioni Europee con processi di Lévy in ipotesi di regime switching

Supponiamo ora che il sottostante evolva come un processo di Lévy esponenziale e che sia presente regime-switching nelle modalità descritte nel Paragrafo 2.2. In questo caso il sottostante è della forma

$$S_t = S_0 \exp \{X_t\},$$

dove X_t segue la dinamica

$$dX_t = \mu_t dt + \sigma_t dW_t + dJ_t.$$

Allora, seguendo quanto svolto nel Paragrafo 2.2 e nel Paragrafo 1.4.2, il prezzo di un'opzione Call Europea si può trovare grazie alla seguente Proposizione.

Proposizione 2.4.1 (PIDE all'indietro per opzioni Europee regime-switching [24]). *Consideriamo un modello di Lévy esponenziale $S_t = S_0 \exp(rt + X_t)$ dove X è un processo di Lévy che verifica la 1.12. Se $\sigma_i > 0$ o*

$$\exists \beta \in]0, 2[, \quad \liminf_{\epsilon \rightarrow 0} \epsilon^{-\beta} \int_{-\epsilon}^{\epsilon} |x|^2 \nu(x) > 0.$$

Si consideri inoltre la catena di Markov

$$R = \{R_t, t \geq 0\} \rightarrow e_i = (0, \dots, 0, 1, 0, \dots, 0)'$$

con matrice dei tassi di transizione Q e con spazio degli stati di dimensione finita N dato da $\{e_1, e_2, \dots, e_N\}$. Sia inoltre $\sigma_t = \langle \sigma, R_t \rangle$.

Allora il valore di un'opzione Europea con scadenza T , regime iniziale pari a i e payoff $H(S_T)$ è dato da $C_i(t, S)$ dove

$$\begin{aligned} C_i &: [0, T] \times]0, \infty[\rightarrow \mathbb{R}, \\ (t, S) &\mapsto C_i(t, S) = \mathbb{E}[H(S_T) | S_t = S], \end{aligned}$$

è continua su $[0, T] \times [0, \infty[$, $C_i^{1,2}$ su $]0, T[\times]0, \infty[$ e posto $\mathbf{C}(t, S) = (C(t, S, e_1), \dots, C(t, S, e_N))$ si ha che \mathbf{C} verifica il seguente sistema di equazioni integro-differenziali alle derivate parziali accoppiate

$$\begin{aligned} &\frac{\partial C_i}{\partial t} + r_i S \frac{\partial C_i}{\partial S} + \frac{\sigma_i^2 S^2}{2} \frac{\partial^2 C_i}{\partial S^2} - r_i C_i \\ &+ \int_{-\infty}^{\infty} \nu(dy) \left[C_i(t, S e^y) - C_i - S(e^y - 1) \frac{\partial C_i}{\partial S} \right] + \langle \mathbf{C}, Q e_i \rangle = 0 \end{aligned}$$

Euristicamente è quello che ci si aspetta poichè, come abbiamo visto nel modello di Black-Scholes, la dinamica del sottostante non influisce in nessun modo sul modo in cui vengono accoppiate tra loro le equazioni. La dimostrazione rigorosa può essere trovata in [24]. Risolvendo questo sistema di equazioni integro-differenziali alle derivate parziali accoppiate si possono calcolare i prezzi delle opzioni in ipotesi di regime switching ciascuna delle quali ha come regime iniziale i . I metodi numerici per risolvere tali problemi saranno presentati nel Capitolo 5.

2.4.2 Opzioni Americane con processi di Lévy in ipotesi di regime switching

Da ultimo consideriamo il caso di option pricing di Americane nell'ipotesi in cui il sottostante evolva secondo un processo di Lévy esponenziale e ci sia uno switch nella volatilità come descritto nel Paragrafo 2.2. Procedendo analogamente si dimostra che è possibile ottenere il prezzo di un'opzione Americana in ipotesi di regime-switching risolvendo un problema alle derivate parziali all'indietro simile a quello nel caso di regime-switching nel modello di Black-Scholes nel quale si ha in aggiunta un termine integrale dovuto alla presenza dei salti nella dinamica del sottostante. Il tutto è sintetizzato nella seguente proposizione.

Proposizione 2.4.2 (Linear Complementary problem regime-switching [23]). Sia $P : [0, T] \times]0, \infty[$ l'unica soluzione continua e limitata

di

$$\begin{aligned} \frac{\partial P_i}{\partial t} + L^S P_i - r P_i + \langle \mathbf{P}, A e_i \rangle &\leq 0, \\ P_i - (K - S)^+ &\geq 0, \\ \left\{ \frac{\partial P_i}{\partial t} + L^S P_i - r P_i + \langle \mathbf{P}, A e_i \rangle \right\} \{P_i - (K - S)^+\} &= 0, \\ su [0, T] \times]0, \infty[\text{ e } \forall S \in]0, \infty[, \\ P(T, S) &= (K - S)^+ , , \end{aligned}$$

dove $\sigma > 0$ e

$$\begin{aligned} L^S P(t, S) &= r_i S \frac{\partial P}{\partial S} + \frac{\sigma_i^2 S^2}{2} \frac{\partial^2 P}{\partial S^2} \\ &\quad + \int_{-\infty}^{+\infty} \nu(dy) \left[P(t, S e^y) - P - S(e^y - 1) \frac{\partial P}{\partial S} \right]. \end{aligned}$$

Allora il valore dell'opzione Americana è dato da

$$P_i(t) = P_i(t, S_t).$$

Quindi in caso di regime-switching tra N stati dovremo risolvere N *linear complementary problem* accoppiati. Il vettore soluzione, indicato con \mathbf{P} , conterrà i valori delle Put P_i , dove P_i è il valore della Put con partenza dal regime i -esimo. I metodi numerici per trattare questo problema saranno presentati nei Capitoli 5 e 6.

2.5 Fatti stilizzati

Nel corso degli ultimi decenni, la letteratura accademica ha dedicato molta attenzione allo studio delle proprietà delle serie storiche finanziarie. Analizzando un'ampia serie di titoli finanziari ci si accorge come siano presenti alcune regolarità empiriche dette **fatti stilizzati**. Le principali sono le seguenti:

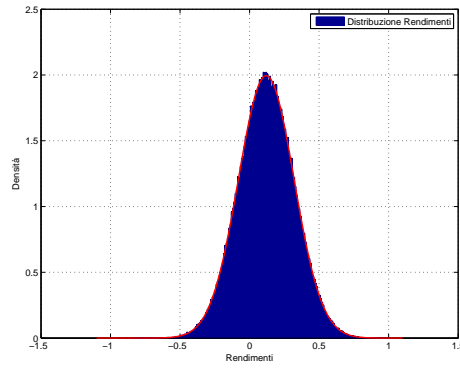
1. Rendimenti giornalieri non distribuiti normalmente.
2. Smile di volatilità.
3. Variabilità della varianza.
4. Bolle finanziarie e incompletezza di informazione.

Di seguito discuteremo in quale misura questi fatti stilizzati siano replicati dai vari modelli proposti.

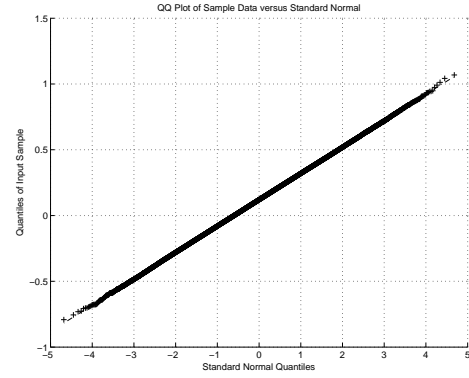
1. Analizzando vari indici azionari possiamo notare come in tutti i mercati i rendimenti siano caratterizzati da asimmetria negativa e da una forma leptocurtica della distribuzione con un'elevata probabilità che si verifichino eventi estremi (fenomeno detto delle code grasse o *fat tails*). In Figura 2.1 è riportata la distribuzione dei rendimenti nel caso di Black-Scholes, Black-Scholes con regime-switching, modello di Lévy e modello di Lévy con regime-switching. Come parametri per ottenere i grafici in Figura 2.1 e per il calcolo delle statistiche in Tabella 2.2 si sono scelti nel caso senza regime-switching $r = 0.14$, $\sigma = 0.2$, mentre per il caso regime-switching abbiamo scelto $r_1 = 0.04$, $r_2 = 0.14$, $\sigma_1 = 0.4$, $\sigma_2 = 0.9$, $q_{11} = -2$, $q_{22} = -1$. Per la parte di salto invece si sono scelti i parametri costanti $\lambda = 1$, $\mu = -0.3$ e $\delta = 0.2$ mentre come maturity abbiamo posto in tutti i casi $T = 1$.

Quello che possiamo osservare nelle figure è che, nel caso del modello di Black-Scholes, i rendimenti sono distribuiti come una normale: ciò si evince sia dal grafico della distribuzione, che molto assomiglia a quello della densità di una Normale, sia dal QQplot. Il grafico del QQplot compara il valore dei quantili di una distribuzione empirica (quella dei rendimenti della serie storica) standardizzata attraverso la deviazione standard empirica con i quantili di una variabile aleatoria Normale standardizzata. Nel caso in cui i rendimenti siano normali, il grafico assume la connotazione di una linea retta passante per l'origine. Nel caso del modello di Black-Scholes con regime-switching la differenza si osserva soprattutto nelle code, dove per un dato livello di probabilità, la distribuzione è caratterizzata da rendimenti inferiori o superiori in misura significativa rispetto a quelli di una variabile Normale: ciò indica una maggiore probabilità di eventi positivi e negativi estremi, di conseguenza, i quantili per livelli di bassa probabilità sono significativamente inferiori rispetto a quanto previsto dalla distribuzione Normale. Il modello di Lévy, invece, riesce a riprodurre una asimmetria della distribuzione con eventi estremi negativi con probabilità superiore al caso Normale. Combinando l'effetto del regime-switching e del modello di Lévy si riesce ad ottenere una distribuzione con un'asimmetria negativa che presenta inoltre l'effetto delle code grasse. Nella Tabella 2.2 sono riportati i valori di skewness, kurtosis, valore del test di Jarque Bera e p-value del test. Il test di Jarque Bera è un test che permette di verificare che il campione selezionato provenga da una distribuzione normale. La variabile testata è:

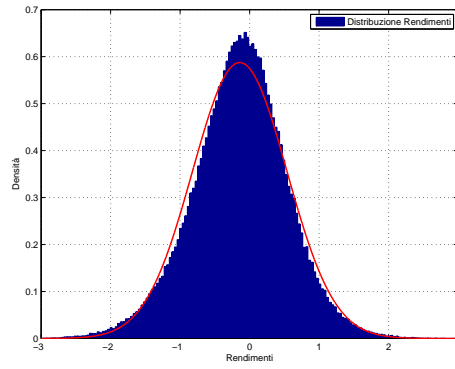
$$JB = \frac{n}{6} \left(S^2 + \frac{(K - 3)^2}{4} \right)$$



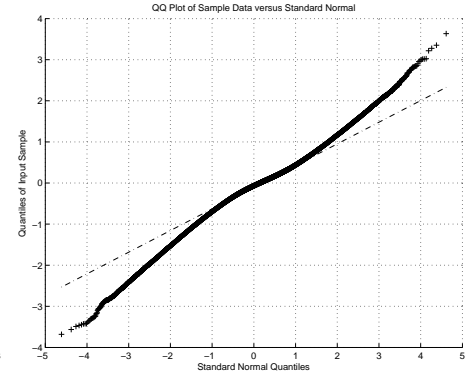
(a) Rendimenti Black-Scholes



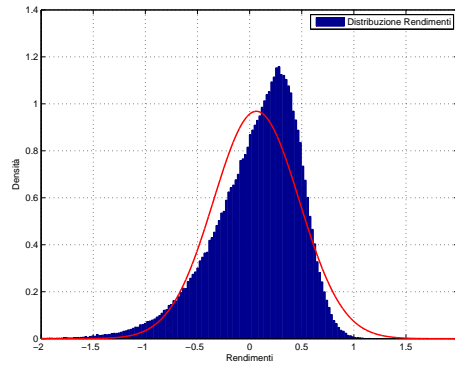
(b) QQplot dei rendimenti di Black-Scholes



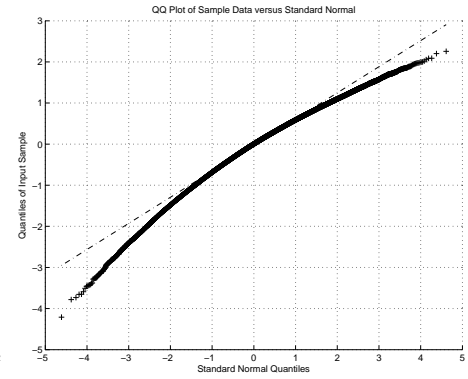
(c) Rendimenti Black-Scholes regime switching



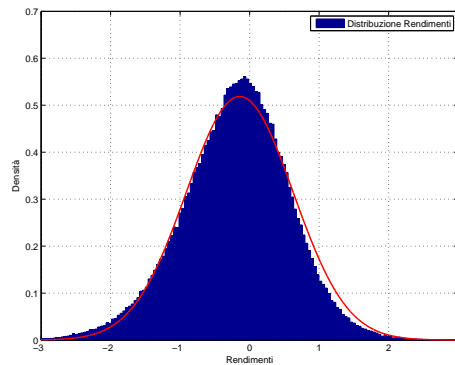
(d) QQplot dei rendimenti di Black-Scholes regime switching



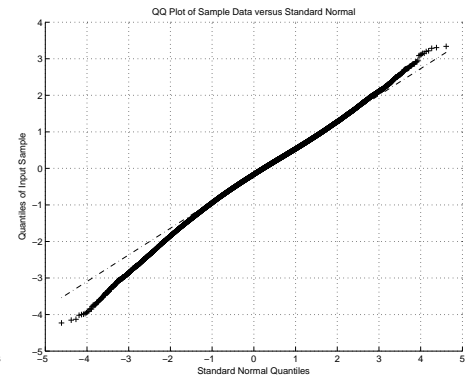
(e) Rendimenti Lévy jump diffusion



(f) QQplot dei rendimenti Lévy jump diffusion



(g) Rendimenti Lévy jump diffusion regime switching



(h) QQplot dei rendimenti Lévy jump diffusion regime switching

Figura 2.1: Distribuzione dei rendimenti e QQplot per i vari modelli

dove

$$S = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^3}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \right)^{3/2}},$$

$$K = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^4}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \right)^2},$$

dove n è il numero di osservazioni. Sotto l'ipotesi nulla di provenienza da una distribuzione Normale, la statistica di Jarque Bera si distribuisce come una χ^2 con due gradi di libertà: si ha che per elevati valori della statistica test si rigetta l'ipotesi nulla.

Modello	S	K	JB	χ^2 p-value
BS	-0.0065	3.0076	3.3231	0.1898
RS	-0.2512	4.0971	2.1232e+04	0.0000
Levy	-0.4598	3.4001	1.4670e+04	0.0000
LevyRS	-0.2342	3.4591	6.2744e+03	0.0000

Tabella 2.2: Valori della Skewness, Kurtosis, statistica di Jarque-Bera e p-value per i vari modelli. L'ipotesi nulla di normalità non viene rigettata solo nel caso del modello standard di Black-Scholes.

Da quando riportato in Tabella (2.2), si osserva come la skewness sia molto vicina allo zero nel caso di modello classico di Black-Scholes, mentre sia significativamente diversa da zero per gli altri tre modelli considerati. Dato che la skewness è un indice di simmetria della distribuzione possiamo dedurre che nel modello di Black-Scholes classico non sia presente asimmetria nei rendimenti: eventi che portano ad alti rendimenti e a bassi rendimenti sono equiprobabili. Osservando invece la kurtosis, indice di allontanamento dalla normalità distributiva (ricordiamo che per una normale si ha che $k = 3$), si può dire che effettivamente i rendimenti di Black-Scholes sono distribuiti normalmente mentre, negli altri modelli, ciò non si verifica: il tutto è avvalorato dai valori della statistica di Jarque-Bera e dal p-value ad essa associato: solo nel caso di Black-Scholes standard l'ipotesi nulla di normalità della distribuzione non è rifiutata. In conclusione osserviamo anche che sia il modello di Black-Scholes con regime-switching sia i modelli di Lévy catturino la distribuzione leptocurtica dei rendimenti: in Figura 2.1 vediamo, infatti, come la distribuzione dei rendimenti presenti un picco più elevato rispetto alla distribuzione Normale che ha come media la media e varianza la media e la varianza dei log-rendimenti.

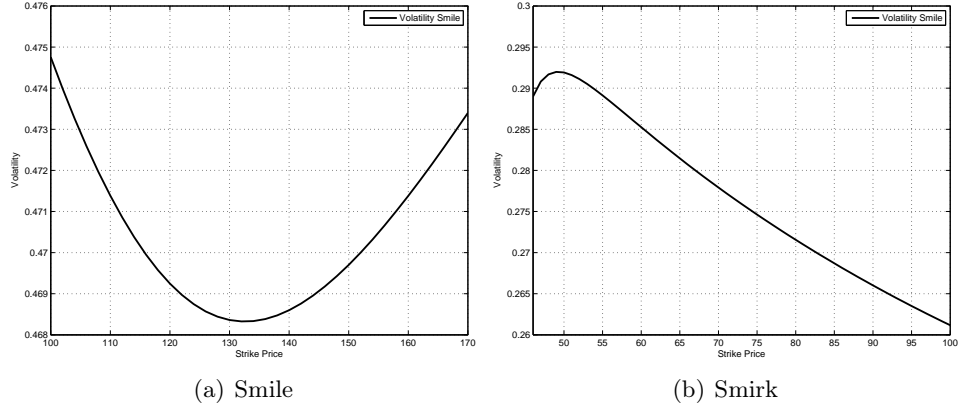


Figura 2.2: Smile e Smirk di volatilità nel modello di Black-Scholes con regime-switching

2. Un'ulteriore ipotesi del modello standard di Black-Scholes è che la volatilità sia costante nel tempo. Il valore di una Call di Black-Scholes è funzione crescente della volatilità tra $]0, +\infty[$ e $\left[(S_t - Ke^{-r(T-t)})^+, S_0 \right]$, dato un prezzo di mercato all'interno di questo intervallo si può trovare un unico valore per la volatilità $\Sigma_t(T, K)$ tale che il valore dell'opzione del modello di Black-Scholes coincida con il valore di mercato. Quindi

$$\exists! \Sigma_t(T, K) > 0, \quad C^{BS}(S_t, K, T, \Sigma(T, K)) = C_t^*(T, K).$$

Questo valore di $\Sigma_t(T, K)$ è detto volatilità implicita. Può essere considerato come “numero sbagliato che inserito nella formula sbagliata produce il prezzo giusto” [12]. Si può osservare come la volatilità implicita $\Sigma_t(T, K)$ dipenda dalle caratteristiche della maturity T e dello strike K . La funzione

$$\Sigma_t : (T, K) \mapsto \Sigma_t(T, K),$$

è detta *superficie di volatilità implicita*. Nel modello di Black-Scholes, il quale presuppone una profilo di volatilità piatto, la funzione $\Sigma_t(T, K)$ è costante: si ha che

$$\Sigma_t(T, K) = \sigma.$$

E' un fatto ben documentato empiricamente che la volatilità implicita non sia costante, anzi presenti degli “smile” oppure degli “smirk”. Inoltre, si presentano anche degli appiattimenti degli smile: per K fissato lo smile si appiattisce al crescere della maturità T . Si può notare come, anche nel caso semplice di switching tra due stati, il fenomeno degli smile di volatilità sia già presente come riportato in Figura 2.2, dove si è posto $S_0 = 100$, $r = 0.14$, $\sigma_1 = 0.2$, $\sigma_2 = 0.8$, $q_{11} = -2$, $q_{22} = -3$.

Anche i modelli di Lévy sono in grado di produrre smile di volatilità. Nella Figura 2.3 sono riportati gli smile di volatilità nel caso in cui il sottostante evolva secondo un modello di Lévy nell'ipotesi che ci sia o meno regime switching. I parametri sono gli stessi del caso precedente a cui abbiamo aggiunto i parametri del modello di Merton jump-diffusion $\lambda = 1$, $\mu = -0.2$ e $\delta = 0.2$.

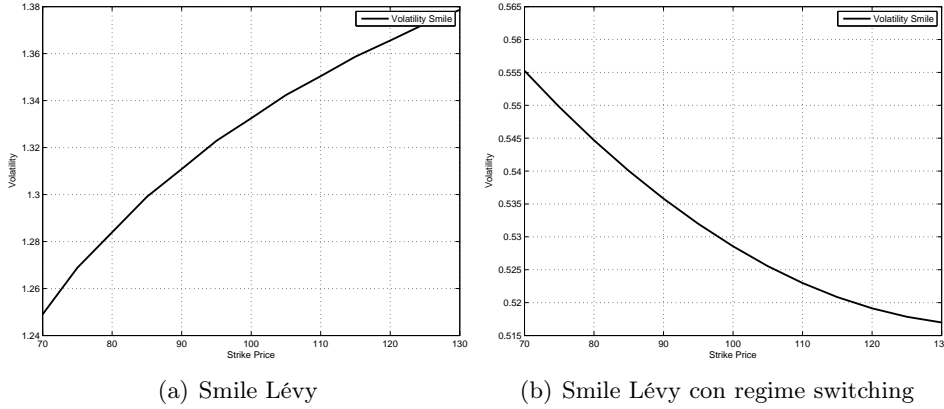
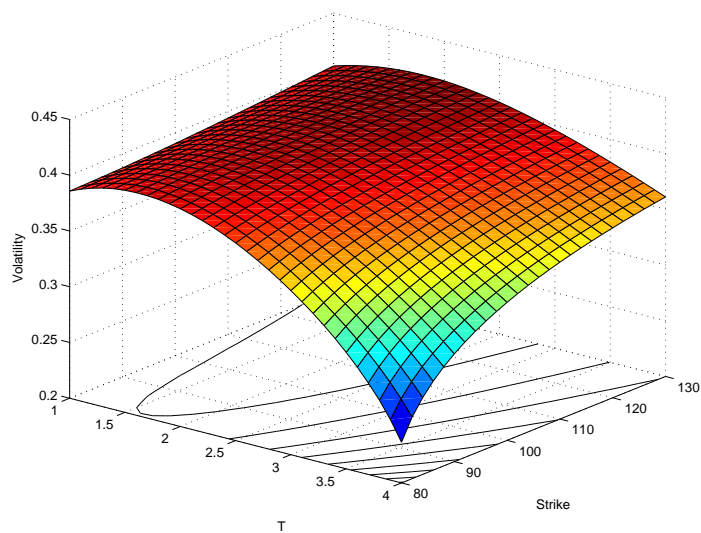


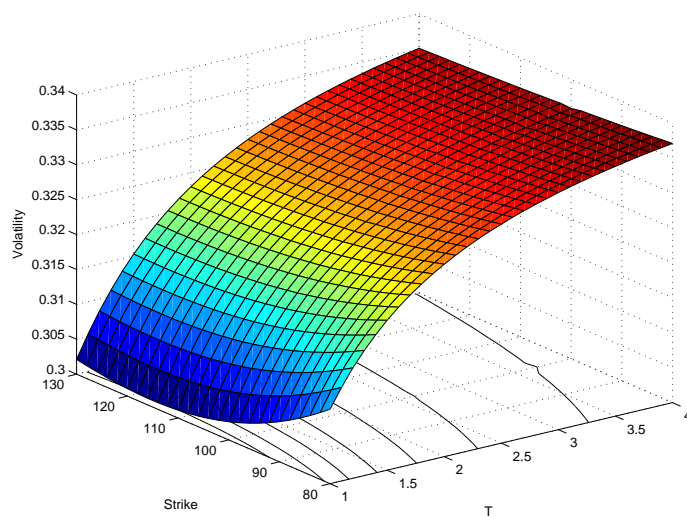
Figura 2.3: Smile di volatilità nei processi di Lévy

Nella Figura 2.4 sono riportate invece alcune superfici di volatilità implicita $\Sigma(T, K)$. Da tutte le considerazioni fatte sopra risulta evidente che sia i modelli di Lévy sia il modello di Black-Scholes con regime-switching riescano a riprodurre il fenomeno fondamentale degli smile di volatilità.

3. Un'altra evidenza empirica che si può riscontrare nei dati di mercato è che la varianza giornaliera dei rendimenti non è costante nel tempo: i rendimenti presentano periodi di volatilità elevata a cui si alternano periodi di volatilità più contenuta. Questa caratteristica è nota come *eteroschedasticità* dei rendimenti. Inoltre, questa variabilità dei rendimenti tende a persistere nel tempo: come osservato da Mandelbrot già nel 1963 [29] “grandi cambiamenti tendono ad essere seguiti da grandi cambiamenti, mentre piccoli cambiamenti tendono ad essere seguiti da piccoli cambiamenti”. I modelli di regime-switching catturano perfettamente questo fatto come possiamo osservare in Figura 2.5, a differenza del modello di Black-Scholes standard, che, assumendo come ipotesi volatilità costante, non può riprodurre questo fenomeno. In entrambe i casi la varianza campionaria al tempo t è calcolata sulle ultime 63 osservazioni.

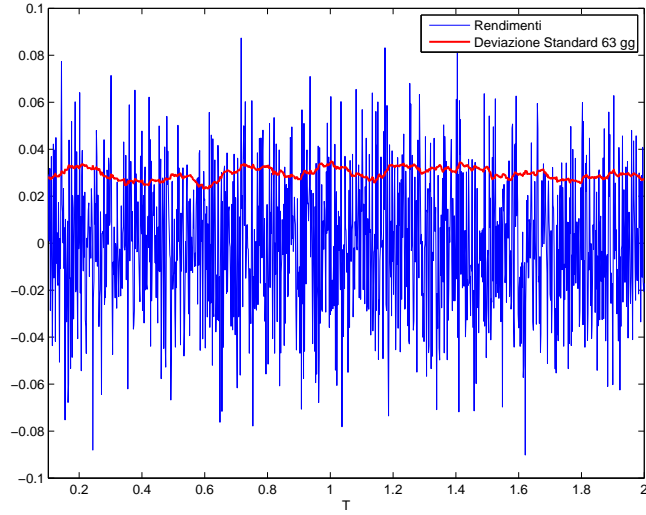


(a) Superficie di volatilità call modello di Lévy con regime switching

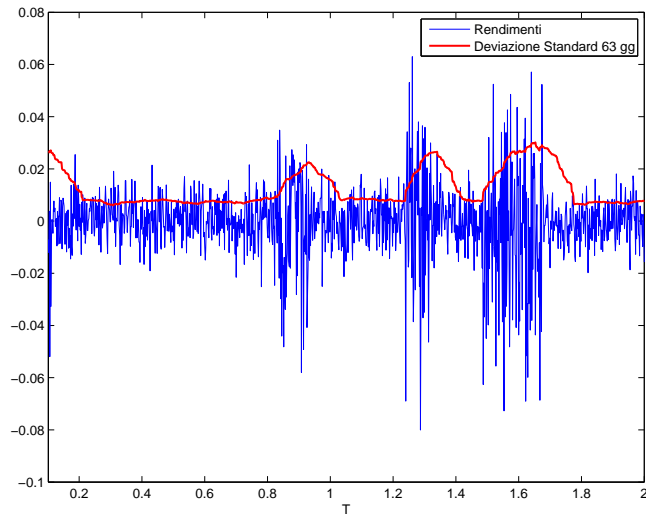


(b) Superficie di volatilità Black-Scholes con regime switching

Figura 2.4: Superfici di volatilità implicite. Parametri $r = 0.14$, $\sigma_1 = 0.1$, $\sigma_2 = 0.7$, $\lambda = 1$, $\mu = 0.1$, $\delta = 0.2$, $q_{11} = -1$, $q_{22} = -2.1$



(a) Modello di Black-Scholes



(b) Modello di Black-Scholes regime switching

Figura 2.5: Variabilità della varianza e volatility clustering. Parametri $r = 0.14$, $\sigma_1 = 0.2$, $\sigma_2 = 0.7$, $T = 2$, $S_0 =$, $q_{11} = -2$, $q_{22} = -3$.

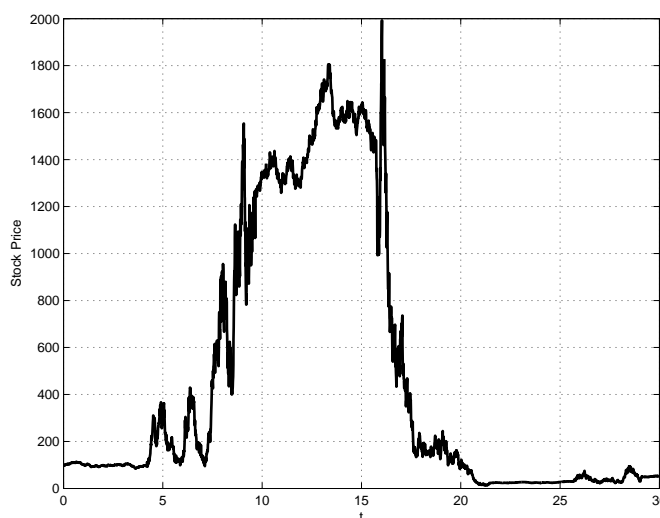


Figura 2.6: Simulazione bolla finanziaria. Parametri $S_0 = 100$, $r = 0.14$, $\sigma_1 = 0.2$, $\sigma_2 = 0.8$, $T = 20$, $q_{11} = -0.1$, $q_{22} = -0.3$.

4. Una caratteristica peculiare dei modelli di regime switching in campo finanziario è la capacità di poter prendere in considerazione il caso in cui si presentino bolle finanziarie. Una bolla finanziaria è una particolare fase di mercato in cui si osserva un aumento sconsiderato ed ingiustificato dei prezzi di un bene, dovuto ad una crescita repentina della domanda di un prodotto. Alla fase di bolla segue una fase di scoppio che tende a ripristinare il valore originario del bene. Lo scoppio di una bolla può verificarsi per vari motivi tra i quali il realizzare ad un certo punto che i guadagni non sono buoni quanto si pensasse. Un caso reale è quanto avvenne nella compagnia canadese di cercatori d'oro Bre-X. Nel 1997 fu annunciata la scoperta di un grande giacimento d'oro in indonesia. Di conseguenza il valore del sottostante si incrementò rapidamente. Così anche la volatilità aumentò di intensità a causa dell'elevata quantità di operazioni sul mercato. Tuttavia una minoranza di persone era consapevole che la notizia fosse in realtà errata. Si riscontrò difatti, poco dopo, che il giacimento d'oro era in realtà irrisorio. Non appena la notizia divenne pubblica i sogni di gloria degli investitori si infransero e la Bre-X fallì.

Un andamento di questo tipo può essere riprodotto agevolmente dai modelli di regime-switching. In Figura 2.6 una traiettoria del modello Black-Scholes con regime-switching che riproduce l'andamento possibile di un titolo in presenza di bolla finanziaria.

Capitolo 3

Fast Fourier Transform

Il metodo della *Fast Fourier Transform* per il pricing di derivati è un metodo sviluppato da Carr e Madan nel 1998 [9]. Tale metodo è applicabile a tutti quei problemi per i quali la funzione caratteristica del processo sottostante è ottenibile analiticamente, requisito soddisfatto da molti modelli. Uno dei punti forza di questo metodo, se non il principale, è che consente di calcolare in poco tempo il valore di una gran quantità di opzioni. In questo capitolo, estenderemo la tecnica della *FFT* proposta da Carr e Madan al caso di modelli regime-switching, seguendo l'impostazione data da Lui, Zhang e Yin in [27].

3.1 Approccio neutrale al rischio e option pricing

Supponiamo di essere nel modello di Black-Scholes in ipotesi di regime-switching. Consideriamo allora la solita dinamica per il sottostante sotto la misura storica di probabilità \mathbb{P}

$$dS_t = \mu_t S_t dt + \sigma_t S_t d\tilde{W}_t,$$

dove μ_t e σ_t sono definiti come nel Capitolo 2. Come già discusso in precedenza l'approccio standard per il pricing è la valutazione risk-neutral. E' possibile adattare il teorema di Girsanov per i modelli regime-switching [41] di modo da trovare una misura di martingala equivalente \mathbb{Q} tale che il sottostante abbia dinamica

$$dS_t = \alpha_t S_t dt + \sigma_t S_t dW_t, \quad (3.1)$$

dove α_t denota il tasso di interesse privo di rischio nella dinamica risk-neutral. Utilizziamo α_t e non r_t perchè non è detto che il drift, sotto la misura \mathbb{Q} , sia pari al tasso risk-free. Ciò accade, ad esempio, se il sottostante distribuisce dividendi. Considereremo, d'ora in avanti, l'approccio risk-neutral,

di conseguenza tutti i valori attesi sono da calcolati rispetto alla misura di martingala equivalente \mathbb{Q} . Sia

$$X_t = \int_0^t \left(\alpha_s - \frac{1}{2} \sigma_s^2 \right) ds + \int_0^t \sigma_s dW_s.$$

Allora la soluzione dell'Equazione (3.1) è data da

$$S_t = S_0 \exp [X(t)].$$

Se consideriamo la Call Europea ($C(K) = C(t, T, S_t, K)$) scritta su S_t e con strike price K e maturity T , grazie al principio di valutazione risk-neutral il prezzo dell'opzione $C(K)$ è dato da

$$C(K) = \mathbb{E} \left[\exp \left(- \int_0^T r_t dt \right) (S_0 e^{X(T)} - K)^+ \right]. \quad (3.2)$$

Posto ora $k = \log(K/S_0)$ allora l'Equazione (3.2) può essere riscritta come

$$C(k) = S_0 \mathbb{E} \left[\exp \left(- \int_0^T r_t dt \right) (e^{X(T)} - e^k)^+ \right].$$

Osserviamo che il prezzo dell'opzione $C(k)$ non decade a 0 per k che tende a ∞ . Per questo motivo non possiamo calcolarne direttamente la trasformata di Fourier. Come proposto da Carr e Madann introduciamo un termine esponenziale che renda quadrato integrabile la funzione per k che varia su $(-\infty, \infty)$. Definiamo allora la funzione di prezzo modificata $c(k)$ definita da

$$c(k) = e^{\rho k} \frac{C(K)}{S_0},$$

dove $\rho > 0$ è un fattore positivo specificato. Ora, è possibile derivare un'espressione in forma chiusa per la trasformata di Fourier di $c(k)$ ¹. Quindi posto

$$\psi(u) = \int_{-\infty}^{\infty} e^{iuk} c(k) dk = \frac{\mathbb{E} \left[e^{-R_T} \Phi_T^{(1)}(u - i(1 + \rho)) \right]}{\rho^2 + \rho - u^2 + i(1 + 2\rho)u},$$

dove $R_T = \int_0^T r_t dt$ e $\phi_T^{(1)}(u) = \mathbb{E} [e^{iuX(T)} | \mathcal{F}_T]$, si ottiene che, nel caso di processo di regime-switching con spazio degli stati $\mathcal{M} = \{1, \dots, m\}$, l'espressione della funzione caratteristica è data da

$$\psi(u) = \frac{\exp(B_1(u)T) \mathbb{E} \left[\exp \left(i \sum_{j=1}^{m-1} A_1(u, j) T_j \right) \right]}{\rho^2 + \rho - u^2 + i(1 + 2\rho)u}, \quad (3.3)$$

¹I dettagli possono essere trovati in Appendice A.

dove

$$T_j = \int_0^T I_{[t=j]} dt, \quad j \in \mathcal{M},$$

è il tempo di soggiorno della catena di Markov nello stato j nell'intervallo di tempo $[0, T]$. Quindi $\sum_{j=1}^m T_j = T$. Le espressioni di $A_1(u, j)$ e $B_1(u)$ sono date, per $j = 1, \dots, m$, da

$$\begin{aligned} A_1(u, j) = & \left[(\alpha(j) - \alpha(m)) + \left(\frac{1}{2} + \rho \right) (\sigma^2(j) - \sigma^2(m)) \right] u \\ & + \frac{1}{2} u^2 (\sigma^2(j) - \sigma^2(m)) i + i [(r(j) - r(m)) - (1 + \rho) (\alpha(j) - \alpha(m))] \\ & - \frac{1}{2} \rho (1 + \rho) (\sigma^2(j) - \sigma^2(m)) i, \end{aligned}$$

$$\begin{aligned} B_1(u) = & iu \left[\alpha(m) + \left(\frac{1}{2} + \rho \right) \sigma^2(m) \right] - \frac{1}{2} u^2 \sigma^2(m) + \\ & (1 + \rho) \alpha(m) - r(m) + \frac{1}{2} \rho (1 + \rho) \sigma^2(m). \end{aligned}$$

Si noti che, per determinare $\psi(u)$ è necessario calcolare la funzione caratteristica del vettore aleatorio $(T_1, \dots, T_{m-1})'$.

L'approccio di Carr e Madan può essere facilmente esteso nel caso in cui il prezzo del sottostante evolva seguendo un processo di Lévy, anche nell'ipotesi di regime-switching come dimostrato da Wang e Chen [39]. Consideriamo, ad esempio, un processo di Merton jump-diffusion con misura di Lévy data da

$$\nu_j(x) = \frac{\lambda_j}{\sqrt{2\pi\delta_j^2}} \exp \left\{ -\frac{(x - \mu_j)^2}{2\delta_j^2} \right\}, \quad j = 1, \dots, m.$$

Se indichiamo con $X_t = \log(S_t/S_0)$ si può dimostrare che la funzione caratteristica $\Phi_T(u)$ può essere scritta come prodotto di due funzioni caratteristiche, una riguardante la parte di diffusione ed una riguardante la parte di salto.

$$\Phi_T(u) = \mathbb{E} \left[e^{iuX_T} \middle| \mathcal{F}_T \right] = \mathbb{E} \left[e^{iuX_T^{(1)}} \middle| \mathcal{F}_T \right] \mathbb{E} \left[e^{iuX_T^{(2)}} \middle| \mathcal{F}_T \right] := \Phi_T^{(1)}(u) \Phi_T^{(2)}(u).$$

Poniamo ora $v = u - i(1 + \rho)$, $J_i = e^{\mu_i + \frac{1}{2}\delta_i^2} - 1$ e

$$\phi_j = -\lambda_j (1 + iuJ_j) + \lambda_j \exp \left(\frac{1}{2} u (2i\mu_i - \delta_j^2 u) \right),$$

per $j = 1, \dots, m-1$, otteniamo che

$$\begin{aligned}\Phi_T^{(2)}(u) &= \exp(\phi_m(u)T) \exp\left(\sum_{j=1}^m (\phi_j(u) - \phi_m(u))T_j\right), \\ &= \exp(B_2(u, m)T) \exp\left(i \sum_j^{m-1} A_2(u, j)T_j\right),\end{aligned}$$

dove per $j = 1, \dots, m-1$ si è posto

$$\begin{aligned}A_2(u, j) &= -i(\phi_j(v) - \phi_m(v)), \\ B_2(u, m) &= -\lambda_m(1 + ivJ_m) + \lambda_m \exp\left(\frac{1}{2}v(2i\mu_m - \delta_m^2 v)\right).\end{aligned}$$

Allora otteniamo che, nel caso di jump diffusion, $\psi(u)$ diventa

$$\psi(u) = \frac{\exp(B(u)T) \mathbb{E}\left[\exp\left(i \sum_{j=1}^{m-1} A(u, j)T_j\right)\right]}{\rho^2 + \rho - u^2 + i(1 + 2\rho)u},$$

dove

$$\begin{aligned}A(u, i) &= A_1(u, j) + A_2(u, j), \\ B(u, m) &= B_1(u, m) + B_2(u, m).\end{aligned}$$

Anche in questo caso risulta necessario calcolare la funzione caratteristica del vettore aleatorio $(T_1, \dots, T_{m-1})'$. Prenderemo in considerazione solo il caso con $m = 2$. Per dettagli sul caso $m > 2$ e sul calcolo della funzione caratteristica del vettore aleatorio dei tempi di soggiorno si rimanda a [27].

Switching tra due stati

Sia ora $m = 2$. Consideriamo il generatore della catena di Markov dato da

$$Q = \begin{bmatrix} -\pi_1 & \pi_1 \\ \pi_2 & -\pi_2 \end{bmatrix}, \quad \pi_1, \pi_2 > 0.$$

Risulta necessario trovare la funzione caratteristica di T_1 , tempo di soggiorno nello stato 1.

Assumiamo che lo stato iniziale $j(0)$ sia j_0 . Definiamo

$$\phi_{j_0}(\theta, T) = \mathbb{E}\left[e^{i\theta T} \middle| j(0) = j_0\right], \quad j_0 = 1, 2.$$

Allora, come è stato dimostrato in [27], $\phi_1(\theta, T)$ e $\phi_2(\theta, T)$ soddisfano il seguente sistema di equazioni integrali:

$$\begin{aligned}\phi_1(\theta, T) &= e^{i\theta T} e^{-\pi_1 T} + \int_0^T e^{i\theta t} \phi_2(\theta, T-t) \pi_1 e^{-\pi_1 t} dt, \\ \phi_2(\theta, T) &= e^{-\pi_2 T} + \int_0^T \phi_1(\theta, T-t) \pi_2 e^{-\pi_2 t} dt.\end{aligned}$$

Prendendo la trasformata di Laplace otteniamo il seguente sistema di equazioni algebriche

$$\begin{aligned}\mathcal{L}\{\phi_1(\theta, T)\} &= \frac{1}{s + \pi_1 - i\theta} + \frac{\pi_1}{s + \pi_1 - i\theta} \mathcal{L}\{\phi_2(\theta, T)\}, \\ \mathcal{L}\{\phi_2(\theta, T)\} &= \frac{1}{s + \pi_1} + \frac{\pi_1}{s + \pi_1} \mathcal{L}\{\phi_1(\theta, T)\}.\end{aligned}$$

Risolto tale sistema, si ottiene

$$\begin{aligned}\mathcal{L}\{\phi_1(\theta, T)\} &= \frac{s + \pi_1 + \pi_2}{s^2 + (\pi_1 + \pi_2 - i\theta)s - i\theta\pi_2}, \\ \mathcal{L}\{\phi_2(\theta, T)\} &= \frac{s + \pi_1 + \pi_2 - i\theta}{s^2 + (\pi_1 + \pi_2 - i\theta)s - i\theta\pi_2}.\end{aligned}$$

Prendendo l'inversa della trasformata di Laplace otteniamo

$$\begin{aligned}\phi_1(\theta, T) &= \frac{1}{s_1 - s_2} \left((s_1 + \pi_1 + \pi_2) e^{s_1 T} - (s_2 + \pi_1 + \pi_2) e^{s_2 T} \right), \\ \phi_2(\theta, T) &= \frac{1}{s_1 - s_2} \left((s_1 + \pi_1 + \pi_2 - i\theta) e^{s_1 T} - (s_2 + \pi_1 + \pi_2 - i\theta) e^{s_2 T} \right).\end{aligned}$$

dove s_1 ed s_2 sono le radici dell'equazione

$$s^2 + (\pi_1 + \pi_2 - i\theta)s - i\theta\pi_2 = 0.$$

In questo caso allora la (3.3) diventa

$$\psi(u) = \frac{\exp(B(u)T) \phi_{j_0}(A(u), T)}{\rho^2 + \rho - u^2 + i(1 + 2\rho)u},$$

dove $A(u) = A(u, 1)$. Essendo tutto noto in forma analitica è possibile ora calcolare numericamente la trasformata di Fourier e quindi il prezzo della Call Europea $C(K)$.

3.2 Algoritmo FFT per l'option pricing

In questo paragrafo introduciamo l'algoritmo della *Fast Fourier Trasform* per il pricing di opzioni nel modello di regime-switching. Ricordiamo innanzitutto che la trasformata di Fourier discreta di una sequenza $\{f_j\}_{j=0}^{N-1}$ è definita da

$$F_l = \sum_{j=0}^{N-1} e^{i(2\pi/N)lj} f_j \quad l = 0, 1, \dots, N-1. \quad (3.4)$$

Si è soliti in generale scegliere N come una potenza di 2 quando viene utilizzata la procedura FFT per il calcolo di $\{F_l\}_{l=0}^{N-1}$. Nota la trasformata $\psi(u)$, il prezzo modificato dell'opzione, indicato con $c(k)$, può essere ottenuto invertendo la trasformata di Fourier

$$c(k) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-iuk} \psi(u) du = \frac{1}{\pi} \int_0^{\infty} e^{-iuk} \psi(u) du. \quad (3.5)$$

A questo punto il prezzo dell'opzione risulta essere, come prima specificato, $C(K) = e^{-\rho k} S_0 c(k)$. Poniamo ora $u_j = j\Delta_u$, $j = 0, 1, \dots, N-1$, dove Δ_u è il passo di discretizzazione della variabile u . Allora (3.5) può essere approssimata dalla seguente sommatoria:

$$c(k) \simeq \frac{1}{\pi} \sum_{j=0}^{N-1} e^{-iu_j k} \psi(u_j) \Delta_u.$$

Sia ora Δ_k il passo della griglia in k e si scelta una griglia lungo il log strike modificato k nel seguente modo

$$k_l = \left(l - \frac{N}{2}\right) \Delta_k, \quad l = 0, 1, \dots, N-1.$$

Allora risulta

$$c(k_l) \simeq \frac{1}{\pi} \sum_{j=0}^{N-1} e^{-ijl\Delta_u} e^{ij(N/2)\Delta_u\Delta_k} \psi(j\Delta_u) \Delta_u \quad l = 0, 1, \dots, N-1.$$

Ponendo

$$\Delta_u \Delta_k = \frac{2\pi}{N},$$

abbiamo infine

$$c(k_l) \simeq \frac{\Delta_u}{\pi} \sum_{j=0}^{N-1} e^{-ijl(2\pi/N)} e^{ij\pi} \psi(j\Delta_u) \quad l = 0, 1, \dots, N-1.$$

Usando la regola di quadratura di Simpson, definiamo una sequenza di pesi

$$w(j) = \begin{cases} \frac{1}{3}, & \text{se } j = 0, \\ \frac{4}{3}, & \text{se } j \text{ è pari}, \\ \frac{2}{3}, & \text{se } j \text{ è dispari}. \end{cases}$$

Allora si ha

$$c(k_l) \simeq \frac{\Delta_u}{\pi} \sum_{j=0}^{N-1} e^{-ijl(2\pi/N)} e^{ij\pi\psi(j\Delta_u)} w(j), \quad l = 0, 1, \dots, N-1.$$

Confrontando quest'ultima espressione con (3.4) si può facilmente vedere che $\{c(k_l)\}_{l=0}^{N-1}$ possa essere ottenuta prendendo la trasformata di Fourier di $\{e^{ij\pi\psi(j\Delta_u)} w(j)\}_{j=0}^{N-1}$.

La cosa importante da notare è che il calcolo della trasformata di Fourier richiede, se fatto direttamente, un costo computazionale di $O(N^2)$. Tra questi algoritmi citiamo l'algoritmo di Cooley-Tuckey oppure l'algoritmo di Rader-Brenner. Tuttavia gli algoritmi del tipo *FFT* richiedono solo $O(N \log N)$ operazioni: generalmente si è solito fattorizzare N , ma esistono algoritmi per il calcolo della trasformata di Fourier per N primo. Il guadagno computazionale è così elevato che l'algoritmo *FFT* ha trovato applicazioni nei più svariati campi: dall'elaborazione dei segnali, alla risoluzione di equazioni alle derivate parziali alla finanza, appunto.

Altro vantaggio della trasformata di Fourier è che ogni volta che viene eseguita essa calcola esattamente N prezzi, uno per ogni k . Se ad esempio $N = 4096$, come proposto da Carr e Madan, otteniamo, in una sola esecuzione, 4096 prezzi. Inoltre, il tempo computazionale è nettamente inferiore a quello richiesto da altri metodi come i metodi Monte Carlo o dal metodo alle differenze finite. Questo metodo di pricing veloce ci permette, da ultimo, di calibrare rapidamente il nostro modello, cosa che non sarebbe possibile utilizzando altri metodi. Il problema della calibrazione è vasto e complesso: ricordiamo solo cosa significhi in breve: calibrare un modello significa ricavare dai prezzi di mercato il set di parametri che governa il modello proposto. Tipicamente, il problema della calibrazione, altro non è che un problema di minimizzazione. Un procedimento tipico è quello di osservare i prezzi di mercato e di andare a scegliere il set di parametri che minimizzi la differenza tra il prezzo di mercato e il prezzo del modello.

Analizziamo ora l'errore commesso dalla *FFT*, limitandoci al caso del modello di Black-Scholes con regime-switching perchè, siamo in possesso, in tal caso, di una formula di pricing diretta, con cui confronteremo i risultati ottenuti tramite *FFT*. Scelti come parametri $r = 0.1$, $\sigma_1 = 0.3$, $\sigma_2 = 0.2$,

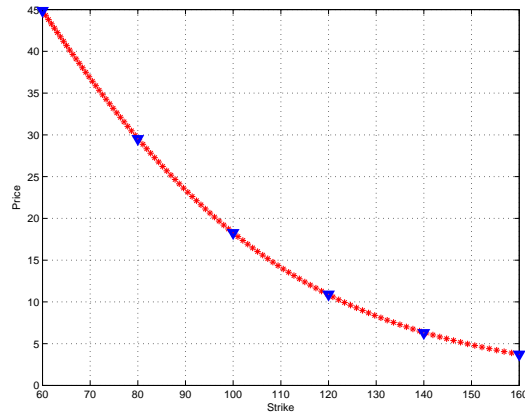


Figura 3.1: Prezzi della Call: in evidenza i prezzi richiesti

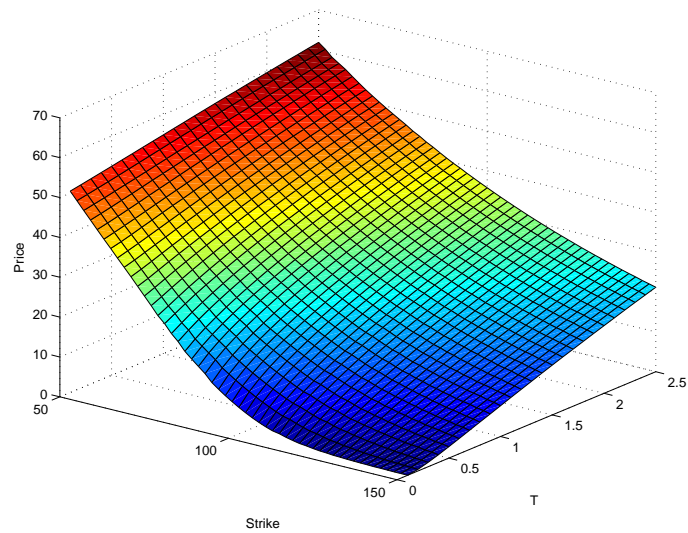


Figura 3.2: Superficie dei prezzi delle Call al variare di K e T

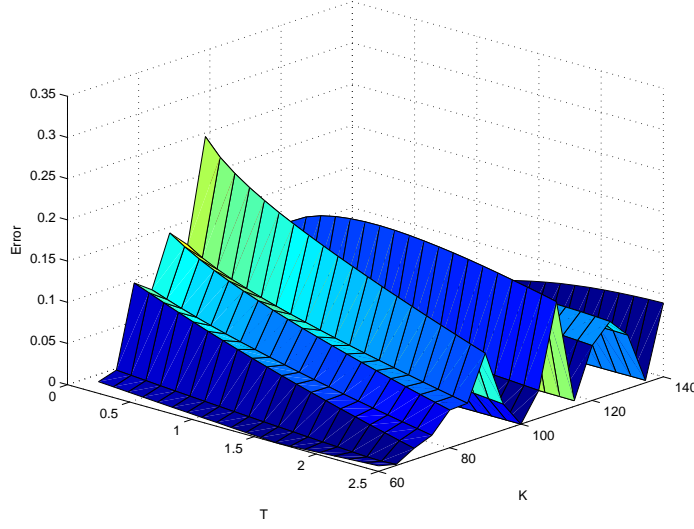


Figura 3.3: Errore al variare della maturità e dello strike.

$\pi_1 = 3$ e $\pi_2 = 1$, definiamo allora i seguenti tipi di errore

$$\begin{aligned}
 RMSE &= \sqrt{\frac{1}{L} \sum_{l=1}^L (C_l^D - C_l^{FFT})^2}, \\
 mean_{err} &= \frac{1}{L} \sum_{l=1}^L C_l^D - C_l^{FFT}, \\
 abs_{err} &= \max_{l=1, \dots, L} |C_l^D - C_l^{FFT}|, \\
 rel_{err} &= \max_{l=1, \dots, L} \left| \frac{C_l^D - C_l^{FFT}}{C_l^D} \right|.
 \end{aligned}$$

dove L è il numero di opzioni considerate mentre C_l^D è il prezzo dell'opzione Call calcolata con l'integrazione diretta. In Figura 3.3 possiamo osservare che l'errore assoluto più elevato per opzioni con scadenze a breve soprattutto se sono out-of-the money, come discusso da Carr e Madann in [9], mantenendosi comunque ragionevolmente "piccolo". Tale comportamento si evince anche dai grafici degli errori in Figura 3.4 sopra definiti.

Soffermiamoci ora un istante sulla scelta del fattore ρ . Lo scopo per cui era stato introdotto da Carr e Madan era rendere integrabile la funzione $C(K)$. Valori positivi di ρ consentono l'integrabilità della funzione modificata sulla parte negativa dell'asse dei log-strike, k , ma penalizzano l'integrabilità per $k > 0$. Come osservato in [9] è possibile dimostrare che la trasformata di Fourier di $c(k)$ esiste sempre se $\mathbb{E}[S_T^{\rho+1}] < \infty$. In pratica,

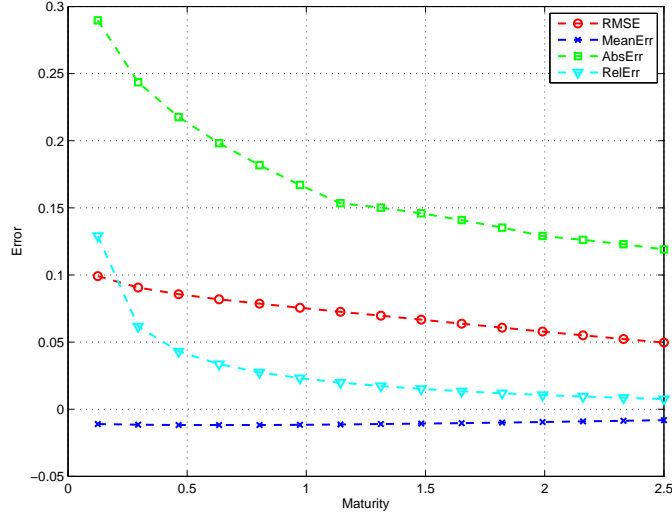


Figura 3.4: Andamento degli errori al variare della maturity.

si può ottenere un upper bound per il valore di ρ dall'espressione analitica della funzione caratteristica osservando che $\phi_{j_0}(-(\rho+1)i) = \mathbb{E}[S_T^{\rho+1}]$.

Il metodo della *FFT*, pur essendo rapido ed accurato, presenta tuttavia alcuni inconvenienti: esso, ad esempio, non è quasi per nulla estendibile al caso delle opzioni Americane. Tuttavia il problema principale è legato alla dipendenza che lega le due griglie di integrazione data da

$$\Delta_u = \frac{2\pi}{N\Delta_k}.$$

E' immediato osservare che una diminuzione del passo Δk porta inevitabilmente ad un aumento del passo Δu a causa proprio dell'inversa proporzionalità che lega queste due grandezze. Ecco perchè è opportuno scegliere adeguatamente i passi di discretizzazione onde poter mantenere sia Δk che Δu sufficientemente "piccoli".

Capitolo 4

Metodi Montecarlo

In questo capitolo presenteremo i metodi di pricing basati su simulazioni Monte Carlo. Tali metodi vennero formalizzati negli anni '40 da Enrico Fermi, John Von Neumann e Stanislaw Ulam ed hanno come padre Nicholas Metropolis [33]. L'idea di questi metodi è molto intuitiva: si cerca di trarre stime attraverso simulazioni. Il metodo Monte Carlo si basa su un algoritmo che genera una serie di numeri tra loro indipendenti, che seguono la distribuzione di probabilità che si suppone abbia il fenomeno da indagare. L'algoritmo Monte Carlo è un metodo numerico che viene utilizzato per trovare le soluzioni di problemi matematici, che possono avere molte variabili e che non possono essere risolti facilmente, per esempio il calcolo integrale. L'efficienza di questo metodo aumenta rispetto agli altri metodi quando la dimensione del problema cresce. Presenteremo, inizialmente, l'idea che sta alla base del metodo dopodichè forniremo gli algoritmi necessari per il pricing di opzioni Europee ed Americane.

4.1 Generalità sul metodo

Supponiamo di avere il consueto spazio di probabilità $(\Omega, \mathcal{F}, \mathbb{P})$ e che una certa quantità α possa essere rappresentata come valore atteso di una variabile aleatoria $X : (\Omega, \mathcal{F}) \rightarrow (\mathbb{R}^d, \mathcal{B}(\mathbb{R}))$, cioè

$$\alpha = \mathbb{E}(X).$$

Supponendo che X sia quadrato integrabile e abbia legge Λ , consideriamo una successione di v.a. indipendenti $\{X_n\}_{n=1,2,\dots,N}$ allora per la Legge dei Grandi Numeri si ha che

$$\bar{\alpha}_N = \frac{1}{N} \sum_{n=1}^N X_n \xrightarrow{N \rightarrow \infty} \mathbb{E}[X] = \alpha.$$

Quindi un buon modo per calcolare α è calcolare la media empirica di un elevato numero di variabili aleatorie indipendenti con stessa legge di X . La velocità di convergenza del metodo, invece, si può studiare utilizzando il Teorema Centrale del Limite, come è discusso in [38]. A questo scopo è utile enunciare il Teorema Centrale del Limite nella seguente formulazione

Teorema 3. *Date $\{X_n\}_{n=1,\dots,N}$ variabili aleatorie i.i.d., posto $\alpha = \mathbb{E}[X]$ e*

$$\bar{\alpha}_N = \frac{1}{N} \sum_{n=1}^N X_n \text{ si ha che}$$

$$\sqrt{N}(\bar{\alpha}_N - \alpha) \xrightarrow{\mathcal{L}} \mathcal{N}(0, C),$$

dove $\xrightarrow{\mathcal{L}}$ indica la convergenza in legge, $\mathcal{N}(0, C)$ è la legge Gaussiana su \mathbb{R}^d di media 0 e matrice di covarianza C .

In particolare, se N è sufficientemente grande, vale la seguente approssimazione.

$$\bar{\alpha}_N \simeq \alpha + \frac{1}{N} C^{1/2} \mathcal{Z}, \quad (4.1)$$

con \mathcal{Z} variabile Gaussiana standard su \mathbb{R}^d . Ciò significa che l'errore aleatorio che si commette approssimando α con $\bar{\alpha}_N$ è dell'ordine di $\frac{1}{N} C^{1/2} \mathcal{Z}$. In altre parole: l'ordine di convergenza del Metodo è $\frac{1}{\sqrt{N}}$, e ciò ne fa un metodo abbastanza lento. Tuttavia, la sua peculiarità è che la velocità di convergenza non dipende dalla dimensione d del problema. Inoltre, ricordiamo che, come osservato in [38], l'andamento dell'errore non è monotono al crescere del numero di simulazioni a causa dei numeri casuali generati, diversi in ogni esperimento.

E' anche possibile, grazie all'espressione (4.1) ottenere un'intervallo di confidenza per α di livello δ . Fissato δ è possibile dimostrare, come fatto in [8], che un'intervallo di confidenza bilatero per α è dato da

$$\bar{\alpha}_N \pm \frac{\bar{\sigma}}{\sqrt{N}} \phi_{\frac{1+\delta}{2}},$$

dove $\phi_{\frac{1+\delta}{2}}$ denota il quantile di ordine $(1 + \delta)/2$ della legge Gaussiana, mentre $\bar{\sigma}$ è una stima della deviazione standard di X tipicamente data dallo stimatore non distorto $\bar{\sigma}^2$ di σ^2

$$\bar{\sigma}^2 = \frac{1}{N-1} \sum_{n=1}^N (X_n - \bar{\alpha}_N)^2.$$

Da ultimo ricordiamo che il metodo Monte Carlo è molto sensibile al seme utilizzato per generare i numeri casuali, motivo per cui, ripetendo più volte l'esperimento, si ottengano risultati simili ma diversi tra loro: questo può essere dovuto ad un potenziale cluster di punti generati. Esistono tuttavia

sia ottimi generatori di numeri casuali che riducono questo fenomeno sia dei metodi, detti di riduzione della varianza, che permettono di ridurre la varianza dell'errore [38, p. 119-123].

Ora che abbiamo presentato le generalità del metodo Monte Carlo lo adatteremo per ricavare gli algoritmi di pricing per contratti Europei ed Americani.

4.2 Metodo Montecarlo per opzioni europee

Il fatto che il prezzo di non arbitraggio di un'opzione possa essere calcolato come

$$V(S_t, t) = \mathbb{E}^{\mathbb{Q}} \left[e^{-\int_t^T r(s) ds} V(S_T, T) \right],$$

suggerisce di poter sfruttare un approccio Monte Carlo. Lo schema che seguiremo è il seguente: considereremo N realizzazioni indipendenti del valore del sottostante nell'intervallo $[0, T]$, ognuno dei quali ha come valore iniziale S_0 . Una volta generate le N traiettorie del valore del sottostante, grazie alla Legge dei Grandi Numeri avremo che il valore dell'opzione è dato dalla media aritmetica, effettuata sulle N simulazioni, scontata per il fattore e^{-rT} del payoff $V(S_T, T)$. La difficoltà del Monte Carlo sta nel simulare adeguatamente il sottostante. A titolo esemplificativo consideriamo il caso in cui il sottostante evolva secondo un modello del tipo Merton jump-diffusion nell'ipotesi di regime-switching con generatore della catena di Markov Q . L'andamento di tale processo è riportato in Figura 4.1(a), mentre in figura 4.1(b) osserviamo il valore della volatilità nel tempo. Da ultimo in Figura 4.1(c) sono rappresentati gli istanti di tempo in cui si verifica un salto nella dinamica del sottostante e l'ampiezza del rispettivo salto. In Figura 4.2 sono rappresentate invece 100 simulazioni della dinamica del sottostante nel modello di Black-Scholes con regime-switching.

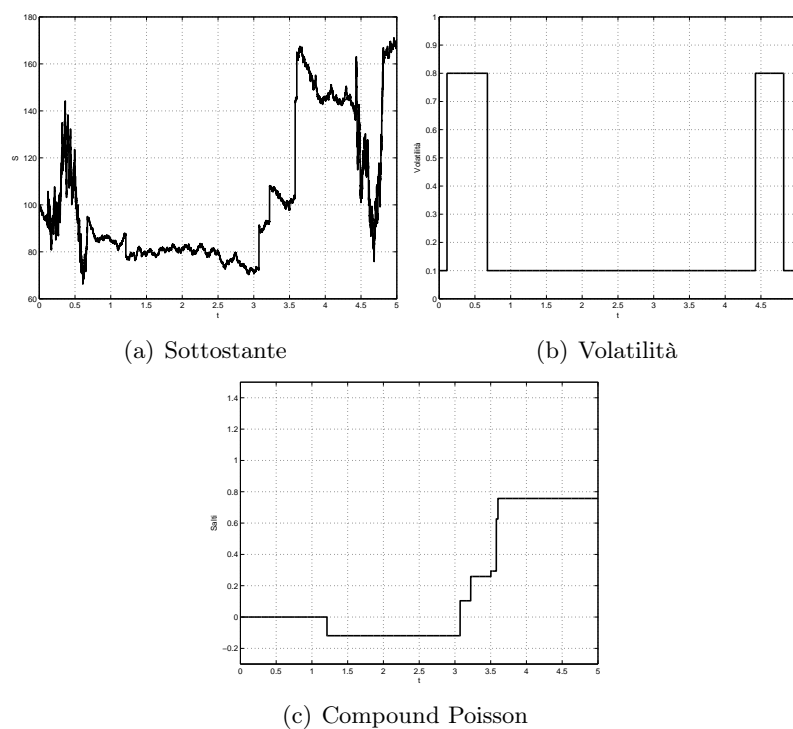


Figura 4.1: Sottostante in presenza di regime-switching e jump-diffusion di tipo Merton. Parametri utilizzati $r_1 = 0.01, r_2 = 0.15, \sigma_1 = 0.1, \sigma_2 = 0.8, \mu = 0.1, \delta = 0.2, \lambda = 1, Q_{11} = -0.3, Q_{22} = -1.1$

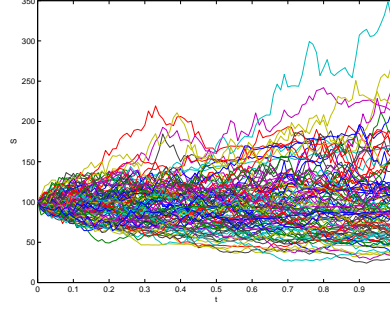


Figura 4.2: Cento traiettorie possibili di un sottostante nel modello di Black-Scholes con regime switching.

La dinamica del sottostante sotto la misura neutrale al rischio \mathbb{Q} è regolata dalla seguente equazione differenziale stocastica

$$dS_t = S_t \left(\left(r_t - \frac{\sigma_t^2}{2} - \lambda \left(e^{\frac{\delta^2}{2} + \mu} - 1 \right) \right) dt + \sigma_t dW_t + dJ_t \right), \quad (4.2)$$

dove r_t è il tasso risk-free, σ_t è la volatilità, μ e δ la media e la varianza dei salti, λ la densità dei salti, mentre dJ_t è la parte dovuta al processo di salto che qui viene assunto della forma

$$J(t) = \sum_{i=1}^{N_t} Y_i.$$

N_t è il numero di salti del processo nell'intervallo di tempo considerato mentre le Y_i sono variabili aleatorie i.i.d e distribuite, nel caso di Merton jump-diffusion, come una normale di media μ e varianza pari a δ^2 . Per semplicità di esposizione abbiamo considerato interessati al cambio di regime solo r e σ : torneremo sul caso in cui varino anche λ , μ e δ alla fine del capitolo.

Innanzitutto osserviamo che è necessario introdurre una griglia sul tempo del tipo $t_0 < \dots < t_N$ discretizzando l'intervallo temporale $[0, T]$: istante per istante andremo a simulare l'avanzamento del sottostante nel passaggio da t_i a t_{i+1} . Per poter simulare correttamente la dinamica del sottostante è necessario, in primo luogo, calcolare i tempi aleatori in cui il processo presenta dei salti. Il risultato di cui ci serviamo afferma che se λ , ovvero l'intensità dei salti, è costante allora N_t è distribuita secondo una legge di Poisson con parametro λt . Di conseguenza gli istanti di salto sono distribuiti con legge esponenziale di parametro λ . Un algoritmo per generare i tempi di salto è il countdown simulation schematizzato nell'Algoritmo 1. Fissato un intervallo di tempo $[0, T]$ si pone come istante iniziale $\tau_0 = 0$ e numero di salti $i = 0$. Si genera una variabile aleatoria con legge esponenziale estraendo prima un numero casuale con distribuzione uniforme tra $[0, 1]$ e poi utilizzandolo nell'inversa della cumulata della distribuzione esponenziale. Questo è il tempo del prossimo salto che va sommato all'istante di salto

Algorithm 1 Countdown simulation

-
- 1: Let $\tau_0 = 0$ and $i = 0$
 - 2: Set $i = i + 1$
 - 3: Generate $U \sim \text{uniform}[0, 1]$
 - 4: set $T_i = -\lambda^{-1} \log U$ and $\tau_i = \tau_{i-1} + T_i$
 - 5: **if** $\tau_i \leq T$ **then**
 - 6: go to step 2
 - 7: **end if**
 - 8: return jump times $\tau_1, \dots, \tau_i - 1$ and the Poisson sample $N_t = i - 1$
-

precedente: raggiunto T , l'algoritmo si ferma e restituendo i tempi di salto ed il numero di salti. I tempi così generati, essendo aleatori, potrebbero non coincidere con quelli della griglia temporale predefinita e deterministica. Quindi, semplicemente, si va a cercare qual è il tempo deterministico più vicino ad ognuno dei τ_i generati, nel seguente modo: si definisca $\tau_i^* = t_j$ tale che $|\tau_i - t_j| = \min_{k=1 \dots N} |\tau_i - t_k|$.

Il passo successivo è quello di considerare gli istanti di tempo in cui si ha il passaggio da uno stato dell'economia all'altro. I tempi di switch seguono una distribuzione esponenziale di parametro $Q_{\theta\theta}$ con $\theta = 1, 2$. Iniziamo a generare il primo tempo di switch estraendo da un'esponenziale di parametro $Q_{\theta\theta}$. Anche qui, come nel caso dei tempi di salto, il tempo di switch generato potrebbe non coincidere con i nodi temporali: analogamente a quanto già fatto prima individuiamo il tempo deterministico più vicino a quello generato. Inizia poi un ciclo sul tempo in cui, in ogni istante, si verifica innanzitutto se il tempo considerato è un tempo di switch: se è così si cambia lo stato θ , dopodichè si genera il successivo tempo di switch. In seconda battuta si simula la parte di diffusione del sottostante in accordo con l'Equazione (4.2) ed infine si controlla se il tempo considerato è un tempo di salto. Se è così allora si genera l'ampiezza del salto estraendo un numero dalla normale di media μ e varianza δ^2 . Fatto ciò è possibile passare al tempo successivo. Tale procedura è formalizzata nell'Algoritmo 2. L'Algoritmo 2 va ripetuto N volte calcolando il payoff scontato per ognuna delle traiettorie, effettuando poi la media aritmetica ottenendo così il prezzo dell'opzione. Se il tasso risk-free è costante l'operazione di sconto per il fattore $e^{-r(T-t)}$ può essere fatta un'unica volta al termine dell'algoritmo sulla media dei payoff. Per quanto riguarda i modelli regime-switching questo approccio non è applicabile perchè r non è costante. Bisognerà, quindi, scontare ogni singola traiettoria calcolandone poi la media. Al termine della singola simulazione Monte Carlo si memorizza lo stato dell'economia finale e si registrano i vari tempi di switch. Dato il vettore dei tempi di switch $T_{\text{switch}} = (T_1, \dots, T_n)$ a partire dall'ultimo elemento del vettore si sconta il payoff moltiplicando per $e^{-r_\theta \Delta T}$ con $\Delta T = T - T_n$. A questo punto si cambia il valore dello stato θ e si passa a scontare il payoff utilizzando $e^{-r_\theta \Delta T}$ dove $\Delta T = T_n - T_{n-1}$. Si

Algorithm 2 Jump Diffusion Path

```

1: Generate m random jump time  $\{\tau_i\}_{i=1,\dots,m}$ 
2: Chose  $M$ , define  $\Delta t = T/M$  and compute  $\{t_k\}_{k=1,\dots,N}$ 
3: Compute  $\{\tau_i^*\}$  and define  $i_k$  such that  $t_{i_k} = \tau_k^*$   $k = 1, \dots, m$ 
4: Generate the first time switch  $\tilde{\tau} \sim \exp\{|Q_{\theta\theta}|\}$ 
5: Define  $i_l$  such that  $t_l = \tilde{\tau}_l^*$ 
6: for  $j=0 : M-1$  do
7:   if  $t_j = t_l$  then
8:     if  $\theta = 2$  then
9:        $\theta = 1$ 
10:    else
11:       $\theta = 2$ 
12:    end if
13:    Generate next time switch  $\tilde{\tau} \sim \exp\{|Q_{\theta\theta}|\}$  and save it
14:    Define  $i_l$  such that  $t_l = t_l + \tilde{\tau}_l^*$ 
15:  end if
16:  simulate  $g_j \sim N(0, 1)$ 
17:   $S_{j+1} = S_j * \exp\left\{\left(r_\theta - \frac{\sigma_\theta^2}{2}\right) \Delta t + \sigma_\theta * \sqrt{2} \Delta t * g_j\right\}$ 
18:  if exist a  $k \in \{1, \dots, m\}$  such that  $j+1 = i_k$  then
19:    Simulate jump size  $Y \sim N(\mu, \delta^2)$ 
20:     $S_{j+1} = S_{j+1} * \exp\{Y\}$ 
21:  end if
22: end for

```

ripete il procedimento fino al primo tempo di switch ottenendo così il valore del payoff scontato. Tale procedura è descritta nell'Algoritmo 3.

Algorithm 3 Discounted payoff

```

1: given a final state  $\theta$ , the vector of switching time  $T_{switch} = (T_1, \dots, T_n)$ ,
    $\Delta t = T/M$  and setted  $C = (S_T - K)^+$  the payoff for a call option
2: for  $j=0 : n-1$  do
3:    $C = C * e^{(-r_\theta(T_{switch}(n-j) - T_{switch}(n-j-1)) * \Delta t)}$ 
4:   if  $\theta = 1$  then
5:      $\theta = 2$ 
6:   else
7:      $\theta = 1$ 
8:   end if
9: end for
  
```

Dopo aver scontato ognuno degli N payoff calcolati con l'Algoritmo 2 si può mediare sui vari payoff ed ottenere il prezzo dell'opzione. In Figura 4.3 è rappresentato il prezzo di una Call Europea nel modello di Merton jump-diffusion in ipotesi di regime-switching al variare dello strike K e della maturity T .

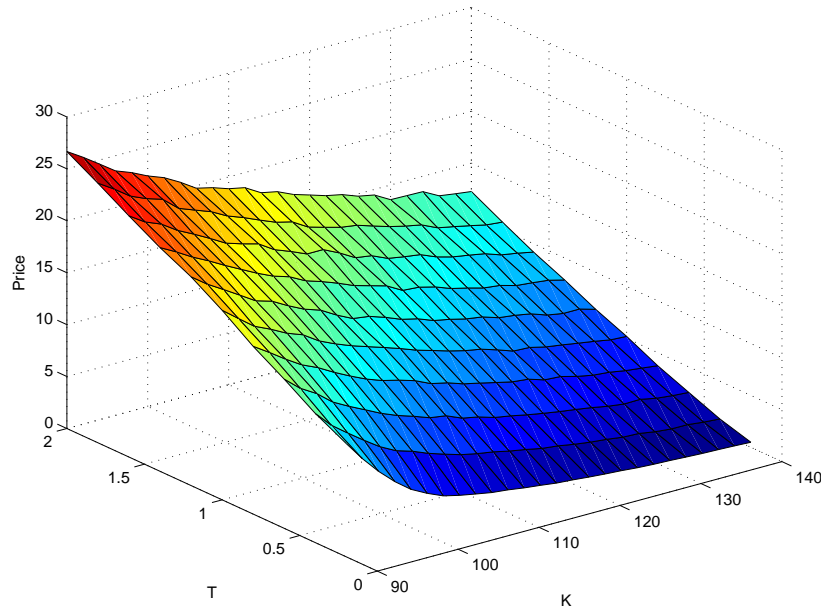


Figura 4.3: Prezzo della Call europea in ipotesi di *regime-switching* al variare di K e di T con parametri $S_0 = 100, r_1 = 0.01, r_2 = 0.15, \sigma_1 = 0.1, \sigma_2 = 0.8, \mu = 0.1, \delta = 0.2, \lambda = 1, Q_{1,1} = -0.3, Q_{2,2} = -1.1$

Verifichiamo numericamente la correttezza dell'algoritmo. Considerando il modello di Black-Scholes con regime-switching ed il set di parametri

$S_0 = 100$, $K = 100$, $r = 0.1$, $\sigma_1 = 0.5$, $\sigma_2 = 0.2$, $Q_{11} = -1$, $Q_{22} = -2$ e facciamo variare il numero di simulazioni, N diminuendo via via lo step temporale aumentando M . Consideriamo il prezzo esatto di una Call Europea con stato di partenza $\theta = 1$, calcolato con la formula di integrazione del Capitolo 2. Dai risultati riportati in Tabella 4.1 possiamo osservare come il prezzo calcolato con il metodo Monte Carlo si avvicini sempre di più al prezzo esatto. La convergenza alla soluzione esatta non è monotona, come precedentemente osservato, mentre l'intervallo di confidenza, di livello $\alpha = 0.5$, si rimpicciolisce all'aumentare di N ed M .

Primo stato						
N	M	DI	MC	IC		Err
1000	100	21.9193	21.1077	[18.9570 23.2585]		0.8116
5000	100	21.9193	21.6342	[20.6066 22.6619]		0.2850
10000	100	21.9193	22.0531	[21.2946 22.8115]		-0.1337
50000	100	21.9193	22.1657	[21.8303 22.5012]		-0.2464
200000	150	21.9193	21.9290	[21.7633 22.0948]		-0.0097
500000	200	21.9193	21.9773	[21.8718 22.0828]		-0.0580

Tabella 4.1: Convergenza del Metodo Monte Carlo.

L'algoritmo qui presentato è l'algoritmo base e si può utilizzare per prezzare tutte le opzione di tipo Europeo in caso di regime-switching variando semplicemente il payoff. Per poter prezzare contratti di tipo Americano è necessario utilizzare un algoritmo più raffinato presentato da Francis Longstaff ed Eduardo Schwartz nel 2001 [28].

4.3 Algoritmo di Longstaff-Schwartz

L'algoritmo di Longstaff e Schwarz non è solo un efficiente algoritmo per prezzare le opzioni Americane con il metodo Monte Carlo ma si rivela necessario per affrontare questo genere di problema. Un approccio ingenuo è quello di pensare che si potrebbero generare per ogni nodo temporale alcune traiettorie, calcolare il payoff e poi scontare all'istante precedente e prendere il massimo tra il payoff e il valore dell'opzione nel nodo. Questo approccio si rivela fallimentare perchè il numero di simulazioni Monte Carlo esplode rapidamente rendendo questo algoritmo ben presto inutilizzabile. L'approccio proposto da Longstaff e Schwartz in [28] richiede invece un numero molto minore di simulazioni Montecarlo ma è meno intuitivo del metodo classico. Il punto chiave dell'algoritmo consiste nell'approssimare il valore atteso del payoff con la tecnica dei minimi quadrati a partire dal suo valore di continuazione.

4.3.1 L'algoritmo

Generiamo, come abbiamo già fatto per le opzioni Europee, con un Monte Carlo S_0, S_1, \dots, S_M dove j è un indice discreto tale che $S_j = S(j\Delta t)$ e $\Delta t = T/M$ dove T è la maturity. Definiamo il valore intrinseco:

$$IV_j = \max(K - S_j)^+,$$

nel caso di una Put e definiamo il valore di continuazione come

$$CV_j = \mathbb{E} [e^{-r\Delta t} V_{j+1} | S_j].$$

A questo punto la programmazione dinamica ricorsiva ci permette di dire che il valore dell'opzione è dato da

$$\begin{aligned} V_M &= \max\{K - S_M, 0\}, \\ V_j &= \max\{IV_j, CV_j\}, \quad j = 1, \dots, M. \end{aligned}$$

Si osservi che l'unica quantità che non è calcolabile al tempo j è CV_j perchè il valore di V_{j+1} è ignoto. L'idea è di approssimare il valore di V_{j+1} con il valore di S_j che al tempo j è noto, tramite polinomi nel seguente modo:

$$CV_j = \mathbb{E} [e^{-r\Delta t} V_{j+1} | S_j] \simeq \sum_{k=1}^L \alpha_{k,j} \phi_k(S_j) = \sum_{k=1}^L \alpha_{k,j} S_j^{k-1},$$

dove si può considerare per ϕ_j la base dei monomi oppure dei polinomi ortogonali. E' importante osservare come ad ogni istante temporale si stia usando una base che dipende dal tempo S_j mentre i pesi sono indipendenti dal tempo. I pesi $\alpha_{k,j}$ possono essere trovati tramite regressione lineare andando indietro nel tempo.

Supponendo di aver generato N traiettorie, si denoti con $S_{i,j}$ il valore del sottostante al tempo j della traiettoria i . Quando $j = M$ si ha che

$$e^{-r\Delta t} \max(K - S_{M,i}, 0) = \sum_{k=1}^L \alpha_{k,M-1} S_{M-1,i}^{k-1} + \epsilon_i.$$

Il problema da risolvere allora è un problema ai minimi quadrati

$$\min_{\alpha_{1,M-1}, \dots, \alpha_{L,M-1}} \sum_{i=1}^N \epsilon_i^2.$$

Supponiamo di essere in un generico istante di tempo j per ogni traiettoria i tale che

$$IV_{j,i} = \max(K - S_{j,i}) > 0,$$

ci sarà un generico istante di tempo $j_e > j$ che inizialmente viene posto pari a M . Il generico problema da risolvere all'istante j sarà:

$$\min_{\alpha_{1,j}, \dots, \alpha_{L,j}} \sum_{i=1}^N \epsilon_i^2,$$

con ϵ_i dato da

$$\epsilon_i = e^{-r(j_e-j)\Delta t} \max(K - S_{j_e,i}, 0) - \sum_{k=1}^L \alpha_{k,j} S_{j,i}^{k-1}$$

A questo punto allora CV_j si può approssimare con

$$CV_j \simeq \sum_{k=1}^L \alpha_{k,j} S_j^{k-1},$$

e calcolare infine

$$V_{j,i} = \max \{IV_{j,i}, CV_{j,i}\}.$$

Una precisazione è dovuta: ai fini dell'implementazione è utile considerare solo i sottoinsiemi delle traiettorie che al tempo j si presentano *in the money*. Difatti se un'opzione non fosse *in the money* non avremmo ragione di esercitare l'opzione.

L'algoritmo è dunque il seguente

Algorithm 4 Longstaff-Schwartz

- 1: Generate N path $S_{i,j}$ with $j = 1, \dots, M$ and $i = 1, \dots, N$
 - 2: **for** $j = 1 : M$ **do**
 - 3: **for each** path such that $IV_{j,i} = \max(K - S_{j,i}) > 0$ **do**
 - 4: Solve the linear regression problem $\min_{\alpha_{1,j}, \dots, \alpha_{L,j}} \sum_{i=1}^N \epsilon_i^2$
 - 5: Approximate $CV_j \simeq \sum_{k=1}^L \alpha_{k,j} S_j^{k-1}$
 - 6: Set $V_{j,i} = \max \{IV_{j,i}, CV_{j,i}\}$
 - 7: **end for**
 - 8: **end for**
 - 9: $V(t, S_t) = \max(K - S_0, \text{mean}(Ve^{-r\Delta t j_e}))$
-

4.3.2 Adattamento dell'algoritmo nell'ipotesi regime-switching

L'algoritmo sopra presentato è estendibile al caso in cui i parametri del modello cambino aleatoriamente secondo una dinamica dettata da una catena di Markov a tempo continuo come precedentemente discusso. Particolare

attenzione va posta quando si deve calcolare il valore di continuazione. Infatti bisognerà considerare il tasso risk-free r_θ dello stato θ in cui ci si trova. Per lo stesso motivo non sarà possibile scontare per $e^{-r\Delta t_{j_e}}$ perchè il tasso risk-free non è deterministico e di conseguenza non può essere portato fuori dal valore atteso. L'approccio da seguire è abbastanza simile a quello adottato per il metodo Monte Carlo per opzioni Europee. Effettuando N simulazioni è opportuno, per ogni simulazione, salvare, in un vettore ad esempio, lo stato in cui ci si trova al tempo t_j . La trattazione è, da questo punto in poi, identica al caso non regime-switching fatto salvo che il problema di regressione da affrontare risulta essere

$$\min_{\alpha_{1,j}, \dots, \alpha_{L,j}} \sum_{i=1}^N \epsilon_i^2,$$

dove però ϵ_i dato da

$$\epsilon_i = e^{-r_\theta(j_e-j)\Delta t} \max(K - S_{j_e,i}, 0) - \sum_{k=1}^L \alpha_{k,j} S_{j,i}^{k-1}.$$

Come detto, l'altro punto da considerare è il passo in cui si sconta il payoff: l'operazione andrà fatta traiettoria per traiettoria considerando il tasso risk-free dello stato θ dell'economia in cui ci si trova. In pratica è sufficiente sostituire la riga 9 dell'Algoritmo 4 con la procedura descritta nell'Algoritmo 5.

Algorithm 5 Discounted payoff

- 1: Given a matrix Υ , where $\Upsilon_{i,j}$ is the state of the path i at time j
 - 2: **for each** path $i = 1, \dots, N$ **do**
 - 3: **for** $j = j_e : -1 : 0$ **do**
 - 4: $\theta = \Upsilon_{i,j}$
 - 5: $V_i = e^{r_\theta j \Delta t} V_i$
 - 6: **end for**
 - 7: **end for**
-

Supponendo di avere la matrice Υ , dove $\Upsilon_{i,j}$ rappresenta lo stato della simulazione i al tempo j , per ognuna delle traiettorie si sconta il payoff a partire dall'istante di esercizio j_e fino all'istante iniziale, utilizzando come fattore di sconto $e^{r_\theta j \Delta t}$, dove r_θ è il tasso risk-free associato allo stato $\theta = 1, 2$.

In Figura 4.4 è riportato il prezzo di Put Americana al variare dello strike e della maturity, nel caso in cui il sottostante evolva secondo il modello di Merton jump-diffusion con switching su r e σ .

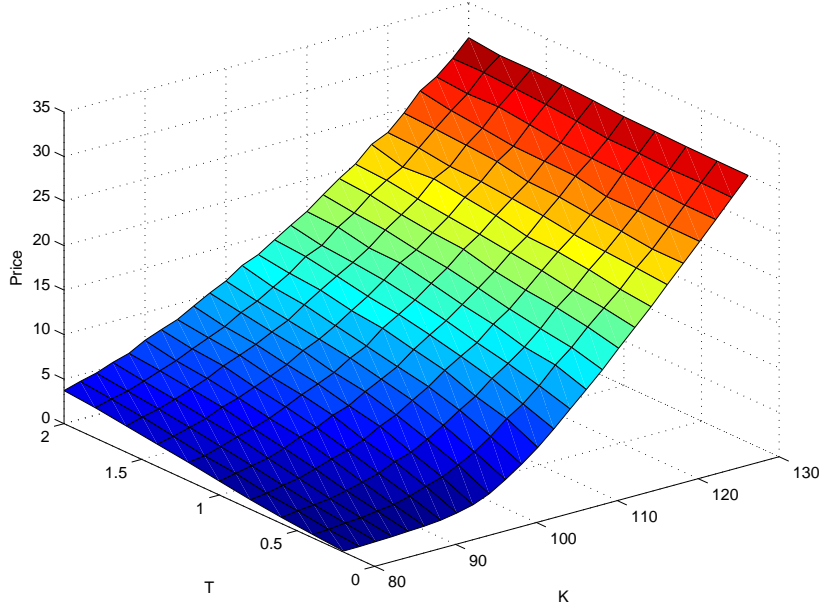


Figura 4.4: Prezzo della Put Americana in ipotesi di *regime-switching* al variare di K e di T con parametri $S_0 = 100, r_1 = 0.05, r_2 = 0.15, \sigma_1 = 0.1, \sigma_2 = 0.4, \mu = 0.1, \delta = 0.2, \lambda = 1, Q_{11} = -0.5, Q_{22} = -1.1$

4.4 Un'ulteriore estensione

Abbiamo, finora, trattato solo il caso in cui i parametri della parte di salto, cioè λ, μ, δ , siano costanti. Il procedimento può essere esteso anche nel caso in cui tali parametri siano soggetti a switch in maniera analoga a quanto avviene per r e σ . L'unica difficoltà che si presenta è la seguente: come è possibile adattare l'algoritmo di countdown simulation nel caso in cui λ non sia costante? In questo caso si presentano due approcci. Una prima idea può essere quella di ricordarsi al caso con $\lambda_t = \bar{\lambda}$, con $\bar{\lambda}$ costante, attraverso una stima. Limitandoci al caso in cui sia possibile il passaggio tra due stati, cerchiamo di stimare $\bar{\lambda}$ facendo N_{sim} simulazioni in ognuna delle quali generiamo i tempi di switch e poi calcoliamo un λ^m , facendo una media pesata sui tempi di soggiorno nel primo e nel secondo stato. L'espressione per $\bar{\lambda}$ è la seguente e la procedura è riassunta nell'Algoritmo 6:

$$\bar{\lambda} \simeq \frac{1}{N_{sim}} \sum_{i=1}^{N_{sim}} \lambda_i^m.$$

Stimato $\bar{\lambda}$, possiamo passare a prezzare le opzioni con gli algoritmi presentati per il caso λ costante. Avendo cura di sostituire μ e δ , media e ampiezza dei salti, con μ_i e δ_i che competono allo stato i .

Algorithm 6 Stima di $\bar{\lambda}$

```

1: for  $i = 1 : N_{sim}$  do
2:   Generate switch-time according to  $Q$ 
3:   Calculate  $\lambda_i^m = \frac{\lambda_1 * \tau_1 + \lambda_2 * \tau_2}{T}$ 
4:   where  $\tau_i$  is the sojourn time in the state  $i$ 
5: end for
6:  $\bar{\lambda} = \frac{1}{N_{sim}} \sum_{i=1}^{N_{sim}} \lambda_i^m$ 

```

Una soluzione alternativa si può ricavare utilizzando il fatto che, come già osservato, i tempi di salto e i tempi di passaggio di stato siano variabili aleatorie tra loro indipendenti. Questo ci autorizza ad adottare un approccio che prevede la generazione dei tempi di salto con parametri λ_1 e λ_2 prima della generazione della dinamica del sottostante. Quest'ultima si ottiene attraversando gli algoritmi presentati precedentemente. L'accorgimento da adottare riguarda la scelta dei tempi di salto generati. Valutando ogni istante temporale è opportuno stabilire se questo sia tempo di salto nello stato in cui ci troviamo, osservando i tempi generati all'inizio per gli stati 1 e 2, e, in caso affermativo, generare un salto estraendo da una Normale $\mathcal{N}(\mu_i, \delta_i^2)$. Questo procedimento è motivato dal fatto che il processo di salto è privo di memoria quindi l'eventualità ci sia un salto dipende unicamente da "quanto tempo è passato dal tempo iniziale" e non dalla sua storia precedente. Il processo di salto "non deve curarsi" del fatto che in qualche tratto della sua storia precedente il valore del parametro λ è stato diverso da quello attuale. Per una verifica numerica della correttezza di tali approcci consideriamo un'opzione Call Europea nel modello di Merton jump-diffusion in cui tutti i parametri siano soggetti a switching. Non essendo in possesso di una formula chiusa per il pricing di opzioni, consideriamo corretto il prezzo calcolato con la *FFT*. Scegliamo come parametri $S_0 = 100$, $K = 100$, $r_1 = 0.05$, $r_2 = 0.09$, $\sigma_1 = 0.4$, $\sigma_2 = 0.2$, $\lambda_1 = 1$, $\lambda_2 = 2$, $\mu_1 = -0.1$, $\mu_2 = 0.1$, $\delta_1 = 0.2$, $\delta_2 = 0.1$, $T = 1$. Come parametri per il metodo Monte Carlo scegliamo $M = 200$ ed $N_{sim} = 300000$, mentre per la *FFT* si è scelto di porre il numero di nodi $N = 4096$, $\rho = 1$ e $\Delta k = 0.01$. Per il calcolo di $\bar{\lambda}$ si è scelto di effettuare 20000 simulazioni. Dai risultati riportati in Tabella 4.2, possiamo dedurre la correttezza dei due approcci. Nel caso in cui si approssimi λ con un certo $\bar{\lambda}$ l'errore commesso è più elevato e l'intervallo di confidenza per il prezzo dell'opzione più ampio rispetto al caso in cui si sfrutti l'indipendenza dei tempi di salto da quelli di switch. Questo perché utilizzando $\bar{\lambda}$ si introduce un'ulteriore approssimazione che rende meno accurato il risultato. Possiamo, quindi, affermare che l'idea migliore sembrerebbe essere quella di sfruttare l'indipendenza dei tempi di switch dai tempi di salto del sottostante.

Primo approccio			Secondo approccio		
MC	$IC_{\alpha=0.5}$	Err	MC	$IC_{\alpha=0.5}$	Err
19.4187	[19.2969 19.5404]	-0.1845	19.2127	[19.0924 19.3330]	0.0215

Tabella 4.2: Confronto tra i due approcci nel caso di switching anche nei parametri della parte di salto. Il prezzo calcolato con la *FFT* è pari a 19.2342.

Una verifica ulteriore della correttezza di entrambe gli approcci e la conferma che sfruttare l'indipendenza dei processi del sottostante e dei passaggi di stato sia meglio che approssimare λ numericamente con $\bar{\lambda}$ costante, sarà oggetto di discussione nel Capitolo 7.

Capitolo 5

Metodo delle differenze finite

Il metodo delle differenze finite è un metodo numerico per risolvere le equazioni alle derivate parziali basato sull'idea di sostituire le derivate con differenze finite.

Per la costruzione di uno schema alle differenze finite, nel caso in cui si abbia a che fare con una *PIDE*, è necessario considerare i seguenti fatti:

- Localizzazione: il dominio della *PIDE* è inizialmente illimitato. Operando in aritmetica finita, introdurremo un dominio limitato che porterà ad un errore di approssimazione.
- Discretizzazione in spazio: il dominio spaziale è sostituito da una griglia discreta di nodi, gli operatori differenziali da una matrice, e la soluzione da un vettore che rappresenta la soluzione calcolata nei nodi della griglia spaziale.
- Discretizzazione in tempo: la derivata in tempo viene sostituita dalle differenze finite scegliendo tra diversi tipi di approssimazioni che conducono a schemi numerici differenti.
- Approssimazione dei piccoli salti: quando il processo di Lévy è ad attività infinita, esso presenta una discontinuità nell'origine. E' necessario prestare attenzione quando si calcola numericamente il termine integrale poichè esso potrebbe divergere.

Nella prima parte del capitolo ci occuperemo di trattare queste problematiche e di derivare uno schema numerico per prezzare le opzioni Europee, dopodiché presenteremo gli algoritmi necessari per il pricing di opzioni Americane.

5.1 Generalità del metodo ed opzioni Europee

Come sappiamo, il prezzo di un'opzione Europea soddisfa, sotto determinate ipotesi ed opportune condizioni agli estremi del dominio, la seguente equazione integro-differenziale alle derivate parziali

$$\begin{aligned} \frac{\partial C}{\partial t} + rS \frac{\partial C}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 C}{\partial S^2} - rC \\ + \int_{-\infty}^{\infty} \nu(dy) \left[C(t, Se^y) - C - S(e^y - 1) \frac{\partial C}{\partial S} \right] = 0. \end{aligned} \quad (5.1)$$

Una volta risolta tale equazione, in unione con le condizioni agli estremi del dominio, ci fornirà il prezzo dell'opzione Europea. Descriviamo ora i passaggi necessari per la derivazione di uno schema numerico alle differenze finite.

5.1.1 Localizzazione a dominio limitato

Poichè un calcolatore può operare solo con una griglia finita di punti è necessario troncare il dominio della *PIDE*. Il problema del troncamento del dominio è piuttosto complesso e richiede elementi di analisi funzionale. In particolare si può dimostrare che l'errore dovuto al troncamento del dominio decresce esponenzialmente con l'aumentare dell'ampiezza dell'intervallo $[-A, A]$ come dimostrato in [12, Proposizione 12.10] e come si può vedere in Figura 5.1. Un'approccio meno formale, che però può essere utile per avere un'idea di quanto ampio debba essere l'intervallo, è il seguente: tramite il metodo Monte Carlo si simulano vari andamenti per il sottostante e si vede qual è il valore massimo che viene raggiunto, indicato con S_{max} , e quale il minimo, indicato con S_{min} . Il dominio della *PIDE* diventa allora $[S_{min}, S_{max}]$. Ovviamente un'approccio del genere non ha un fondamento teorico, ma può essere un buon punto di partenza per la soluzione del problema. Perchè il problema risulti ben posto è necessario aggiungere condizioni al bordo "artificiali" per la soluzione anche fuori dal dominio $]S_{min}, S_{max}[$ e non solamente in S_{min} ed S_{max} . Poichè il valore dell'opzione $C(S, t)$ si comporta asintoticamente come il payoff $h(S)$ per grandi valori di $|S|$, una scelta adeguata è porre

$$C_A(S, t) = h(S), \quad \forall S \notin [S_{min}, S_{max}].$$

come proposto in [12, Sezione 12.4.1].

Analogamente, anche la riduzione ad un dominio finito per il termine integrale si rivela necessaria. Questo si può fare troncando l'integrale in prossimità di certi upper/lower bounds indicati con B_r, B_l : questo procedimento corrisponde a troncare i grandi salti del processo di Lévy considerato, cioè sostituendo $\nu(x)$ con $\nu(x) \mathbb{I}_{x \in [B_l, B_r]}$. L'analisi dell'errore dovuto a questa approssimazione può essere trovata in [12, Sezione 12.4.1].

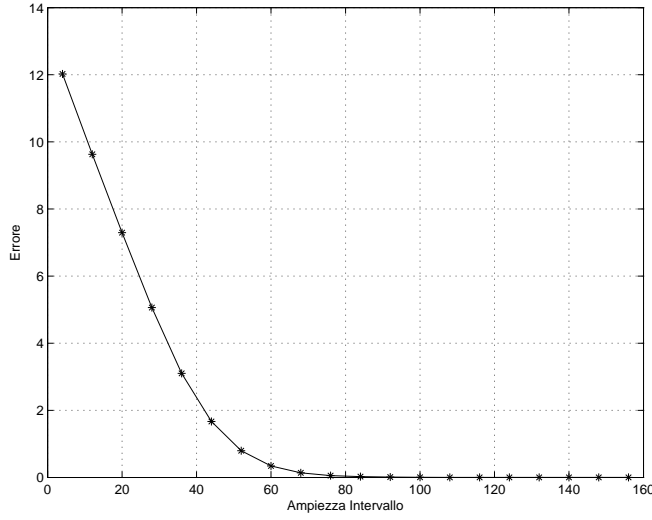


Figura 5.1: Andamento dell'errore di troncamento all'aumentare dell'ampiezza dell'intervallo $[S_{min}, S_{max}]$.

5.1.2 Discretizzazione in spazio

Il passo successivo è sostituire il dominio $]S_{min}, S_{max}[$ con una griglia discreta. Consideriamo una griglia uniforme su $[0, T] \times [S_{min}, S_{max}]$:

$$\begin{aligned} S_i &= S_{min} + i\Delta t, & i &= 0 \dots N, & \Delta S &= (S_{max} - S_{min})/N, \\ \tau_n &= n\Delta t, & n &= 0 \dots M, & \Delta t &= T/M. \end{aligned}$$

Sia C_i^n la soluzione al tempo n nel nodo i estesa, ponendola uguale al payoff, fuori dal dominio $[S_{min}, S_{max}]$.

Consideriamo il caso in cui il processo di Lévy sia ad attività finita cioè $\nu(\mathbb{R}) = \lambda < \infty$ e poniamo $\alpha = \int_{-\infty}^{\infty} (e^y - 1) \nu(dy)$. L'Equazione (5.1) può essere riscritta come

$$\begin{aligned} \frac{\partial C}{\partial t} + (r - \alpha) \frac{\partial C}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 C}{\partial S^2} - (r + \lambda) C \\ + \int_{-\infty}^{\infty} \nu(dy) C(t, S e^y) = 0. \end{aligned} \tag{5.2}$$

Per approssimare i termini integro-differenziali si può procedere ponendo :

$$\begin{aligned}
\lambda &\simeq \tilde{\lambda} = \sum_{j=K_l}^{K_r} \nu_j, \\
\alpha &\simeq \tilde{\alpha} = \sum_{j=K_l}^{K_r} (e^{y_j} - 1) \nu_j, \\
JC &= \int_{-\infty}^{\infty} \nu(dy) C(t, Se^y) \stackrel{z=Se^y}{=} \int_0^{\infty} C(t, z) \nu\left(\frac{dz}{z}\right) \\
&\simeq \int_0^{L_r} C(t, z) \nu\left(\frac{dz}{z}\right) \simeq JC^n = \sum_{j=0}^L \nu\left(\log\left(\frac{z_j}{S}\right)\right) C(t_n, z_j) \frac{\Delta z}{z_j},
\end{aligned} \tag{5.3}$$

posti K_l, K_r tali che $[B_l, B_r] \subset [(K_l - 1/2) \Delta S, (K_r + 1/2) \Delta S]$, con L_r da scegliersi sufficientemente grande così che il contributo dell'integrale in $[L_r, \infty)$ sia trascurabile e con $\Delta z = L_r/L$ dove L è il numero di nodi di discretizzazione dell'intervallo $[0, L_r]$.

Per quanto riguarda le derivate in spazio possono essere approssimate con:

$$\begin{aligned}
\left(\frac{\partial^2 u}{\partial x^2}\right) &\simeq \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}, \\
\left(\frac{\partial u}{\partial x}\right) &\simeq \frac{u_{i+1} - u_{i-1}}{2\Delta x}.
\end{aligned}$$

E' da tenere presente il fatto che lo schema che nasce con l'approssimazione della derivata prima con una differenza centrata può presentare instabilità nel caso in cui il drift sia superiore alla diffusione. Per ovviare a questo problema si può utilizzare lo schema di tipo *up-wind*, presentato in [12, Sezione 12.4.2], sostituendo l'approssimazione della derivata prima con:

$$\left(\frac{\partial u}{\partial x}\right) \simeq \begin{cases} \frac{u_{i+1} - u_i}{\Delta x}, & \text{se } r - \tilde{\alpha} < 0 \\ \frac{u_i - u_{i-1}}{\Delta x}, & \text{se } r - \tilde{\alpha} \geq 0 \end{cases}.$$

Pur garantendo la stabilità del metodo, questo schema consente un'approssimazione solo al primo ordine contro un'approssimazione al secondo ordine per le differenze centrate.

Un altro fattore a cui occorre prestare attenzione è il calcolo del termine integrale. Se il processo di Lévy è ad attività finita non si verificano problemi perché l'integrale converge. Se invece il processo di Lévy fosse ad attività infinita lo schema presentato sopra non potrebbe essere applicato direttamente perché la misura $\nu(x)$ presenterebbe una singolarità nell'origine. Quello che si può fare, come proposto in [12, Sezione 12.4.3], è troncare la misura di

Lévy vicino all'origine, sostituendola con $\nu(x) \mathbb{I}_{|x|>\epsilon}$, il che corrisponde a ignorare i salti inferiori ad un certo $\epsilon > 0$. Per migliorare ulteriormente l'approssimazione si possono sostituire i piccoli salti con un moto browniano come proposto da Asmussen e Rosiński in [1].

5.1.3 Discretizzazione in tempo

Vediamo ora come trattare la discretizzazione della dimensione temporale. Indichiamo con D e J rispettivamente le matrici che rappresentano la parte differenziale e la parte integrale. Si può ora scegliere di utilizzare uno schema completamente esplicito, completamente implicito o un θ -metodo. Lo schema esplicito ha la caratteristica di essere molto veloce computazionalmente, non richiedendo inversioni di matrici, ma ha il grosso svantaggio di essere instabile non appena il rapporto $\frac{\Delta t}{\Delta x^2}$ cresce oltre una certa soglia. Quindi, se volessimo una griglia fitta, dovremmo aumentare vertiginosamente il numero di punti nella griglia temporale, come mostrato in [38, Sezione 4.2.4], e quindi il tempo di computazione crescerebbe eccessivamente. Al contrario, un metodo completamente implicito non ha problemi di stabilità ma è molto oneroso da utilizzare poichè ad ogni passo è richiesta l'inversione di una matrice piena, operazione di per sè computazionalmente molto costosa.

I θ -metodi combinano le due precedenti soluzioni. Indichiamo con u^n la soluzione dell'Equazione (5.2) al tempo generico n . Scegliendo $\theta \in [0, 1]$ e lo schema che nasce è il seguente:

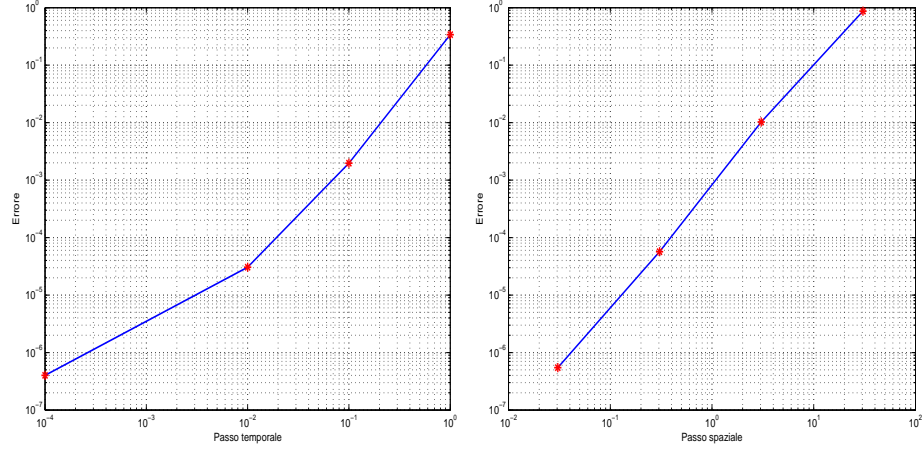
$$\frac{u^{n+1} - u^n}{\Delta t} = \theta (Du^n + Ju^n) + (1 - \theta) [Du^{n+1} + Ju^{n+1}]. \quad (5.4)$$

Se $\theta = \frac{1}{2}$ lo schema è detto di Crank-Nicholson ed è particolarmente vantaggioso perché consente un ordine di convergenza in tempo di $O(\Delta t^2)$. Tale peculiarità è visibile in Figura 5.2 dove è riportato, su un grafico logaritmico, l'andamento dell'errore in funzione dell'ampiezza dell'intervallo di discretizzazione. La pendenza della retta è circa 2 e questo permette di affermare che l'ordine di convergenza è quello ipotizzato.

Per risolvere il problema useremo un metodo *esplicito-implicito*, descritto nell'Algoritmo 7. Decidiamo di trattare l'operatore differenziale con il θ -metodo mentre ed in modo esplicito il termine integrale. Quest'ultima scelta permette di velocizzare l'algoritmo, non prevedendo l'inversione di una matrice piena ad ogni iterazione, ma comporta la perdita di un ordine di convergenza in tempo: tale metodo è noto come *operator-splitting* e può essere utilizzato per risolvere il problema presentato nella Sezione 1.4.2.

5.1.4 Approccio per il regime-switching

Nel Capitolo 2 abbiamo visto come nell'ipotesi di regime-switching si debba risolvere un sistema di n equazioni alle derivate parziali accoppiate,



(a) Andamento dell'errore rispetto al passo temporale. (b) Andamento dell'errore rispetto al passo spaziale.

Figura 5.2: Errore schema di Crank-Nicholson

Algorithm 7 Pide Solving Algorithm

- 1: Initialize $u_i^0 = h(x_i)$
- 2: **for** $n=0, \dots, M-1$ **do**
- 3: solve $(I - (1 - \theta)\Delta t D) u^{n+1} = (I + \theta\Delta t D + \Delta t J) u^n$
- 4: **end for**
- 5: Impose boundary condition
- 6: $u_i^n = h(x_i)$ per $i \notin [0, N]$
 where

$$(Du^{n+1})_i = \vartheta \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2} - \varrho \frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2\Delta x} - (r + \tilde{\lambda}) u_i^{n+1},$$

and ϑ is the diffusion term while ϱ is the drift term, obtained comparing 5.4 with the θ -method discretized version of Equation 5.1 with explicit integral term.

dove n è il numero degli stati. Abbiamo già affermato in precedenza che il termine di accoppiamento non influisce sui termini della *PIDE* ma aggiunge ad essa solo dei termini lineari dati da $\langle Q, C \rangle$. Il prezzo di una Call Europea si ottiene risolvendo la seguente equazione integro-differenziale:

$$\begin{aligned} \frac{\partial C}{\partial t} + rS \frac{\partial C}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 C}{\partial S^2} - rC \\ + \int_{-\infty}^{\infty} \nu(dy) \left[C(t, Se^y) - C - S(e^y - 1) \frac{\partial C}{\partial S} \right] + \langle C, Q \rangle = 0. \end{aligned} \quad (5.5)$$

Affrontiamo il problema seguendo quanto presentato nella sezione precedente con l'accorgimento che ora abbiamo n termini in più da trattare cioè $q_{i1}C_1 + \dots + q_{in}C_n$ per l'equazione i . Se non considerassimo questi termini avremmo come risultato che le equazioni sarebbero disaccoppiate: quello che troveremmo alla fine sarebbero i prezzi di n opzioni ciascuna delle quali sarebbe il prezzo di un derivato con differenti parametri del modello i quali però non cambiano nel corso del tempo.

Indicata con u^n la soluzione dell'Equazione (5.5) al tempo n , possiamo decidere di trattare i termini aggiuntivi, $\langle Q, u \rangle$, in maniera implicita, esplicita o con θ -metodo. Per semplicità e per ottenere una maggiore velocità di esecuzione nello schema numerico abbiamo deciso di trattare esplicitamente il termine di accoppiamento così da evitare l'inversione ad ogni passo di matrici piene (nel caso del modello di Lévy). Quindi, nel metodo illustrato nella precedente sezione, sarà sufficiente sostituire il termine $(Ju^n)_i$ con:

$$(Yu^n)_i = (Ju^n)_i + \langle Q, u^n \rangle_i, \quad (5.6)$$

dove Q è il generatore della catena di Markov che regola lo switch e u^n è la soluzione al passo n .

Verifichiamo ora che la scelta di trattare esplicitamente i termini di accoppiamento tra le equazioni, pur diminuendo di molto il tempo di esecuzione, non produca risultati dissimili da quelli ottenuti utilizzando un metodo completamente implicito o un θ -metodo. Consideriamo il modello classico di Black-Scholes con switching tra due stati e supponiamo di voler risolvere il sistema di PDE con il θ -metodo. Discretizziamo il sistema di equazioni seguente

$$-r_i C_i + \frac{\partial C_i}{\partial t} + r_i S \frac{\partial C_i}{\partial S} + \frac{1}{2} \sigma_i^2 S^2 \frac{\partial^2 C_i}{\partial S^2} + \langle C, Qe_i \rangle = 0, \quad i = 1, 2,$$

approssimando le derivate con differenze finite. Posto $\bar{C} = C_1$ e $\tilde{C} = C_2$ otteniamo allora

$$\begin{aligned} & \frac{\bar{C}_i^{n+1} - \bar{C}_i^n}{\Delta t} + \theta \left(r_1 S \frac{\bar{C}_{i+1}^{n+1} - \bar{C}_{i-1}^{n+1}}{2\Delta S} \right. \\ & \quad + \frac{1}{2} \sigma_1^2 S_i^2 \frac{\bar{C}_{i+1}^{n+1} - 2\bar{C}_i^{n+1} + \bar{C}_{i-1}^{n+1}}{\Delta S^2} - r_1 \bar{C}_i^{n+1} + q_{11} \bar{C}_i^{n+1} + q_{12}^{n+1} \tilde{C}_i^{n+1} \\ & \quad + (1 - \theta) \left(r_1 S \frac{\bar{C}_{i+1}^n - \bar{C}_{i-1}^n}{2\Delta S} + \frac{1}{2} \sigma_1^2 S_i^2 \frac{\bar{C}_{i+1}^n - 2\bar{C}_i^n + \bar{C}_{i-1}^n}{\Delta S^2} \right. \\ & \quad \left. \left. - r_1 \bar{C}_i^n + q_{11} \bar{C}_i^n + q_{12}^n \tilde{C}_i^n \right) \right) = 0. \end{aligned} \quad (5.7)$$

Detto allora N il numero di nodi e posto

$$\bar{\mathbf{C}} = \begin{bmatrix} \bar{C}_1 \\ \vdots \\ \bar{C}_{N-1} \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} \tilde{C}_1 \\ \vdots \\ \tilde{C}_{N-1} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \bar{\mathbf{C}} \\ \tilde{\mathbf{C}} \end{bmatrix},$$

otteniamo che il sistema da risolvere ad ogni passo è dato da

$$A\mathbf{C}^{n+1} = B\mathbf{C}^n + \mathbf{b}, \quad (5.8)$$

con A e B matrici di dimensione $2(N-1) \times 2(N-1)$ della forma

$$\begin{bmatrix} c_1 & u_1 & 0 & 0 & \cdots & -q_{12} & 0 & 0 & \cdots & 0 \\ d_1 & c_1 & u_1 & 0 & \cdots & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & d_1 & \ddots & \ddots & \ddots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & 0 & \ddots & c_1 & u_1 & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & d_1 & c_1 & 0 & \cdots & \cdots & 0 & -q_{12} \\ -q_{21} & 0 & \cdots & \cdots & 0 & c_2 & u_2 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots & d_2 & c_2 & u_2 & 0 & \cdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & 0 & d_2 & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & 0 & \ddots & c_2 & u_2 \\ 0 & \cdots & \cdots & 0 & -q_{21} & 0 & \cdots & 0 & d_2 & c_2 \end{bmatrix}$$

dove i coefficienti d_i, c_i, u_i sono rispettivamente gli elementi della diagonale inferiore, principale e superiore della matrice tridiagonale che si ottiene scrivendo in forma matriciale il Problema (5.7). Il vettore \mathbf{b} è il vettore della condizioni al bordo.

Osserviamo che la risoluzione del Problema (5.8) richiede, ad ogni passo, la risoluzione di un sistema lineare di dimensione $2(N-1)$ mentre, trattare il termine di accoppiamento come proposto in (5.6), considerandolo cioè esplicito per ogni valore di θ , riduce il problema alla risoluzione di un sistema lineare $(N-1) \times (N-1)$. In Figura 5.3 notiamo che il tempo computazionale cresce molto più lentamente, al crescere del numero di nodi spaziali, qualora trattassimo i termini di accoppiamento sempre in maniera esplicita piuttosto che con un θ -metodo generico.

Possiamo inoltre notare come i prezzi calcolati considerando esplicitamente i termini di accoppiamento si discostino poco da quelli calcolati come proposto in (5.8), come si evince dalla Tabella 5.1 dove si sono usati $N = 2000$ nodi per la griglia spaziale e $M = 100$ nodi per la griglia temporale. Per verificare la correttezza dei metodi usati riportiamo anche i prezzi ottenuti con la *FFT*.

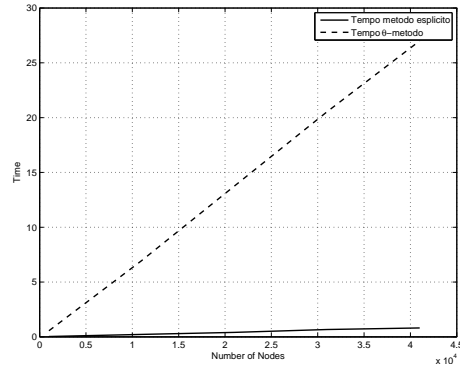


Figura 5.3: Confronto temporale θ -metodo ($\theta = 0.5$) e metodo esplicito sui termini di accoppiamento.

r_1	r_2	σ_1	σ_2	FFT	$Expl$	$Impl$
0.10	0.05	0.50	0.20	19.9258	19.9240	19.9140
0.11	0.05	0.40	0.20	17.5897	17.5901	17.5815
0.80	0.11	0.40	0.30	43.0394	43.0664	43.0348
0.10	0.10	0.40	0.20	18.1572	18.1519	18.1474

Tabella 5.1: Termine accoppiamento: esplicito vs θ -metodo. Modello Black-Scholes.

A conclusioni analoghe si arriva considerando un modello di Lévy, ad esempio il modello di Merton jump-diffusion: i risultati sono riportati in Tabella 5.2, in cui si è scelto $N = 1000$ ed $M = 100$. In entrambi i casi si sono scelti come parametri $S_t = 100$, $K = 90$, $T = 1$, $Q_{11} = -1$, $Q_{22} = -0.3$, $\theta = 0.5$.

r_1	r_2	σ_1	σ_2	λ	μ	δ	FFT	$Expl$	$Impl$
0.10	0.05	0.40	0.30	1	0.10	0.20	20.7426	20.6290	20.6244
0.08	0.10	0.80	0.50	2	-0.10	0.10	31.6942	31.5578	31.5521
0.05	0.15	0.20	0.20	1.5	0.05	0.20	16.5670	16.4220	16.4282
0.03	0.10	0.40	0.30	0.5	0.20	0.15	18.6570	18.5404	18.5421

Tabella 5.2: Termine accoppiamento: esplicito vs θ -metodo. Modello di Merton jump-diffusion.

Questa trattazione ci autorizza a considerare d'ora in avanti, in tutti gli approcci numerici, il termine di accoppiamento in maniera esplicita così da velocizzare la computazione.

La verifica della correttezza e della convergenza del metodo all'infittirsi del numero di nodi spaziali e temporali, può essere fatta scegliendo il modello di Black-Scholes in ipotesi di regime-switching con r costante ed il seguente set di parametri $S_0 = 100$, $K = 100$, $r = 0.1$, $\sigma_1 = 0.5$, $\sigma_2 = 0.2$, $Q_{11} = -1$,

$Q_{22} = -2$, prezziamo una Call Europea con il metodo delle differenze finite ed assumiamo che il prezzo esatto sia quello calcolato con la formula esplicita del Capitolo 2. Consideriamo i vari tipi di errori introdotti nel Capitolo 3. Osserviamo in Figura 5.4 come, all'aumentare dei nodi spaziali, l'errore si riduca sempre di più.

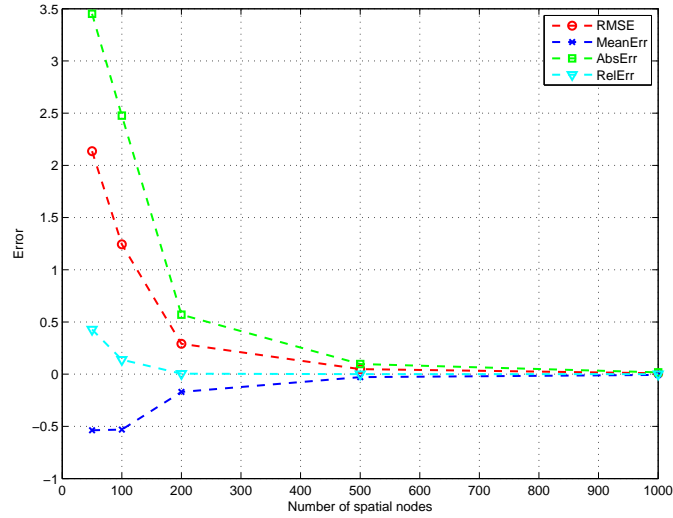


Figura 5.4: Andamento degli errori al variare del numero di nodi spaziali.

In Figura 5.5 mostriamo invece che all'aumentare del numero di nodi spaziali e temporali il prezzo calcolato con il metodo delle differenze finite converge alla soluzione esatta. Si osserva, inoltre, come la convergenza alla soluzione esatta sia molto più rapida all'aumentare dei nodi della griglia temporale che non all'aumentare dei nodi della griglia spaziale. Convergenza e correttezza del metodo sono avvalorate dai risultati riportati in Tabella 5.3.

		Primo stato			Secondo stato		
N	M	DI	DF	Err	DI	DF	Err
10	10	21.9193	17.3531	4.5662	18.7597	14.7425	4.0172
100	50	21.9193	21.9168	0.0025	18.7597	18.7499	0.0098
300	70	21.9193	21.9173	0.0019	18.7597	18.7586	0.0011
1000	100	21.9193	21.9186	0.0007	18.7597	18.7605	-0.0008
2000	200	21.9193	21.9189	0.0005	18.7597	18.7600	-0.0003

Tabella 5.3: Convergenza del metodo alle differenze finite modello di Black-Scholes regime-switching.

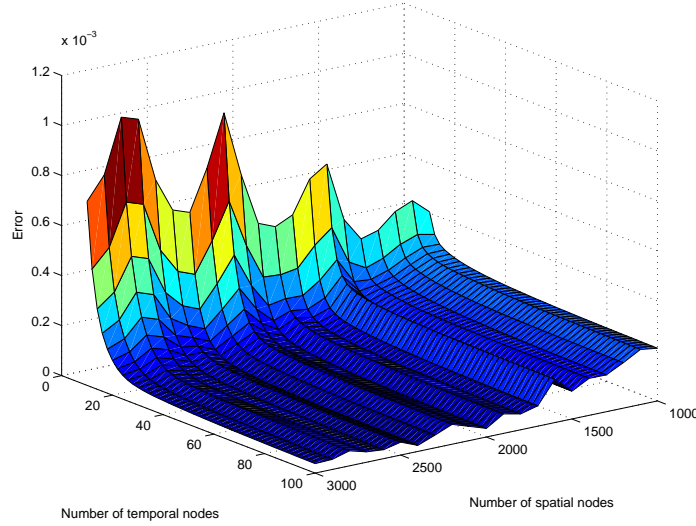


Figura 5.5: Andamento dell' errore medio al variare del numero di nodi spaziali e temporali.

Un cenno sull'accuratezza

Sebbene sia possibile risolvere direttamente l'equazione (5.1) scrivendo lo schema numerico presentato nella sezione precedente, con un semplice cambio di variabile ci si può ridurre a studiare un'equazione i cui coefficienti siano costanti. Assumendo che

$$\int_{|y| \geq 1} e^{2y} \nu(y) < \infty, \quad (5.9)$$

e che,

$$\exists \beta \in]0, 2[, \quad \liminf_{\epsilon \rightarrow 0} \epsilon^{-\beta} \int_{-\epsilon}^{\epsilon} |x|^2 \nu(x) > 0, \quad (5.10)$$

si può introdurre il cambio di variabili definendo la log-forward moneyness x ed il time to maturity τ

$$\tau = T - t \quad x = (S/K) + r_\tau. \quad (5.11)$$

Riscrivendo il payoff ed il valore dell'opzione Call Europea in funzione di queste due variabili otteniamo

$$w(\tau, x) = \frac{e^{r\tau}}{K} C(T - \tau, K e^{x - r\tau}), \quad h(x) = (e^x - 1)^+. \quad (5.12)$$

E' possibile quindi dimostrare che

Teorema 4. [12, Corollario 12.2] Sotto le condizioni (5.9) e (5.10) la funzione

$$w : [0, T] \times \mathbb{R} \rightarrow \mathbb{R},$$

definita da (5.11)-(5.12) è continua su $[0, T] \times [0, \infty[$, $C^{1,2}$ su $]0, T[\times]0, \infty[$ e verifica la seguente equazione integro-differenziale

$$\begin{aligned} \frac{\partial w}{\partial t}(\tau, x) &= \frac{\sigma^2}{2} \left[\frac{\partial^2 w}{\partial x^2}(\tau, x) - \frac{\partial w}{\partial x}(\tau, x) \right] \\ &= \int_{-\infty}^{\infty} \nu(dy) \left[w(\tau, x+y) - w(\tau, x) - (e^y - 1) \frac{\partial w}{\partial x}(\tau, x) \right]. \end{aligned} \quad (5.13)$$

con condizione iniziale $\forall x \in \mathbb{R}, w(0, t) = (e^x - 1)^+$.

Per completezza ricordiamo che la presenza di regime-switching comporta l'aggiunta del termine usuale $\langle \mathbf{w}, Q \rangle$ al membro di sinistra dell'equazione precedente.

Per quanto riguarda la risoluzione numerica dell'Equazione (5.13) lo schema presentato per la *PIDE* originaria è facilmente adattabile. Sostituendo il dominio spaziale con $[-A, A]$ e considerando la griglia discreta

$$x_i = -A + i\Delta t, \quad i = 0 \dots N, \quad \Delta x = 2A/N,$$

l'unica accortezza riguarda il termine

$$\int_{-\infty}^{\infty} \nu(dy) w(\tau, x_i + y) \int_{B_l}^{B_r} \nu(dy) w(\tau, x_i + y) \simeq \sum_{j=K_l}^{K_r} \nu_j w_{i+j}^n, \quad (5.14)$$

con K_l, K_r tali che $[B_l, B_r] \subset [(K_l - 1/2)\Delta x, (K_r + 1/2)\Delta x]$. Siccome è necessaria la soluzione su $[-A + B_l, A + B_r]$ per calcolare il termine integrale, la vera griglia di computazione si estende da $i = K_l$ a $i = N + K_r$ quindi poniamo $w_i^n = h(x_i)$ per $i \notin [0, N]$ dove $h(x_i)$ è il payoff dell'opzione. Scrivendo il θ -metodo con $\theta = 1$ per il problema (5.13) e trattando il termine integrale in maniera esplicita è possibile dimostrare che tale schema numerico è incondizionatamente stabile e consistente con la *PIDE* per $(\Delta t, \Delta x) \rightarrow 0$. Per una dimostrazione formale della convergenza dell'algoritmo si rimanda a [12, Proposizione 12.13].

Nel corso delle prove numeriche effettuate durante lo svolgimento di questo lavoro abbiamo riscontrato una scarsa accuratezza nel metodo scritto senza passare in coordinate log-moneyness nel caso in cui λ , cioè l'intensità dei salti del processo di Poisson, sia troppo elevato e $\mu > 0$: l'errore commesso inizia ad essere intollerabile con $\lambda \geq 8$. Non entreremo nei dettagli tecnici della questione poichè essi comporterebbero una lunga discussione e la trattazione esulerebbe dallo scopo di questa tesi. Passando in log-moneyness si

nota un aumento considerevole dell'accuratezza anche nel caso in cui λ sia elevato. In questo secondo caso, però, è rischioso il calcolo del termine (5.14) che va eseguito numericamente per interpolazione: ne consegue un maggiore tempo di calcolo computazionale rispetto allo schema numerico scritto per l'Equazione (5.2). Possiamo concludere che, siccome in ambito finanziario raramente il parametro λ è maggiore di 1 è più conveniente il pricing risolvendo numericamente direttamente l'Equazione (5.1). Qualora invece λ dovesse diventare troppo grande si consiglia di risolvere l'Equazione (5.13) così da ottenere prezzi più accurati. In Tabella 5.4 riportiamo i tempi di computazione necessari per i due schemi numerici all'aumentare dei nodi di discretizzazione spaziale: per mostrare la correttezza dello schema numerico scritto per l'Equazione (5.13) abbiamo riportato anche i prezzi calcolati da entrambi i metodi.

N	$PIDE$	$PIDE_{log-money}$	$TPIDE$	$T_{log-money}$
400	21.2122	21.1773	5.7915	23.5320
800	21.1729	21.1758	19.9734	61.5303
1000	21.1649	21.1755	30.2156	87.0644
2000	21.1489	21.1750	116.5361	272.9403

Tabella 5.4: Tempi di computazione per la risoluzione dell'Equazione che determina il prezzo di un'opzione Call Europea. $S_0 = 100$, $K = 100$, $r_1 = 0.08$, $r_2 = 0.02$, $\sigma_1 = 0.4$, $\sigma_2 = 0.2$, $\lambda = 1$, $\delta = 0.2$, $\mu = -0.1$, $q_{11} = 0$, $q_{22} = 0$, $M = 100$.

5.2 Metodi numerici per le opzioni Americane

La risoluzione della $PIDE$ per le opzioni Americane richiede metodi più complessi rispetto a quelli usati per le Europee. In accordo con quanto detto nel Capitolo 1 il prezzo di un'opzione Americana $V(t, S)$ è data dall'unica soluzione continua e limitata del seguente *linear complementary problem*:

$$\begin{aligned}
& \frac{\partial V}{\partial t} + L^S V - rV \leq 0, \\
& V - h(S) \geq 0, \\
& \left\{ \frac{\partial V}{\partial t} + L^S V - rV \right\} \{V - h(S)\} = 0 \\
& su [0, T] \times]0, \infty[, \quad e \quad \forall S \in]0, \infty[, \\
& V(T, S) = h(S),
\end{aligned} \tag{5.15}$$

dove si è posto

$$L^S V(t, S) = rS \frac{\partial V}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 V}{\partial S^2} + \int_{-\infty}^{+\infty} \nu(dy) \left[V(t, Se^y) - V - S(e^y - 1) \frac{\partial V}{\partial S} \right],$$

e dove $h(S)$ è il payoff dell'opzione. Ponendo, come nella sezione precedente,

$$\nu(\mathbb{R}) = \lambda < \infty \quad \alpha = \int_{-\infty}^{\infty} (e^y - 1) \nu(dy),$$

$L^S V$ assume la forma

$$L^S V(t, S) = (r - \alpha) S \frac{\partial V}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 V}{\partial S^2} - \lambda V + \int_{-\infty}^{+\infty} \nu(dy) V(t, Se^y). \quad (5.16)$$

Per poter risolvere questo problema possiamo ricorrere ad un metodo iterativo per la soluzione di sistemi lineari chiamato *SOR* proiettato. Il *SOR* proiettato è una versione leggermente modificata del più noto *SOR* che a sua volta è un raffinamento del metodo di Gauss-Seidel. Nel prossimo paragrafo faremo una panoramica dei tre metodi e si vedrà come utilizzare il *SOR* proiettato per risolvere il Problema (5.15).

5.2.1 I metodi di Jacobi, di Gauss-Seidel e del rilassamento

In questa sezione presentiamo alcuni metodi iterativi per la risoluzione di sistemi lineari, seguendo la linea proposta in [36]. Consideriamo il seguente sistema lineare:

$$A\mathbf{x} = \mathbf{b},$$

dove A è una matrice e \mathbf{x} è il vettore delle incognite e \mathbf{b} è il vettore dei termini noti. L'idea è basata su una decomposizione della matrice A della forma $A = P - N$. Fatto ciò ed assegnato $\mathbf{x}^{(0)}$ si ottiene $\mathbf{x}^{(k+1)}$ risolvendo i sistemi

$$P\mathbf{x}^{(k+1)} = N\mathbf{x}^{(k)} + \mathbf{b}.$$

La matrice di iterazione è $P^{(-1)}N$ e quindi il costo computazionale per invertire P dovrà essere basso se si vuole trarre qualche vantaggio dal metodo. Supponiamo che gli elementi diagonali della matrice A siano non nulli. Allora possiamo mettere in evidenza in ogni equazione la corrispondente incognita e ottenere il seguente metodo detto di **Jacobi**

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right].$$

Ciò equivale allo splitting $A = P - N$ con

$$P = D, \quad N = D - A = E + F,$$

dove D è la matrice diagonale che rappresenta gli elementi diagonali di A , E è la matrice triangolare inferiore con coefficienti $e_{ij} = -a_{ij}$ se $i > j$, $e_{ij} = 0$ se $i \leq j$ ed F è la matrice triangolare superiore con $f_{ij} = -a_{ij}$ se $j > i$ e $f_{ij} = 0$ se $j \leq i$. Il metodo di **Gauss-Seidel** segue la stessa idea del metodo di Jacobi con la differenza che al passo $k + 1$ si usano i valori di $x_i^{(k+1)}$ se disponibili. Si ottiene allora

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right],$$

dove si è quindi usato lo splitting $A = P - N$ e

$$P = D - E, \quad N = F.$$

Introducendo un parametro di rilassamento $\omega \in (0, 2)$ si ottiene il metodo del rilassamento successivo (**SOR**)

$$x_i^{(k+1)} = \frac{\omega}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right] + (1 - \omega) x_i^{(k)}.$$

Il parametro ω così scelto garantisce la convergenza del metodo SOR e, se opportunamente selezionato, può velocizzarne la convergenza [36, p. 114-119].

5.2.2 SOR proiettato

Per risolvere il Problema (5.15) è necessario introdurre il **SOR proiettato** che indicheremo con **PSOR**. Se utilizzassimo il **SOR** così come presentato nella precedente sezione, non terremmo conto in alcun modo del vincolo $\mathbf{V}^{n+1} \geq h(S)$.

Denotiamo per semplicità \mathbf{V}^n il vettore delle soluzioni approssimate al tempo n e con \mathbf{g}^n il vettore che rappresenta il vincolo. Considerando $N + 1$ nodi abbiamo:

$$\mathbf{V}^n = \begin{bmatrix} V_1^n \\ \vdots \\ V_{N-1}^n \end{bmatrix}, \quad \mathbf{g}^n = \begin{bmatrix} g_1^n \\ \vdots \\ g_{N-1}^n \end{bmatrix}.$$

Non includiamo i termini V_0^n e V_N^n perché sono valori al bordo ed ad ogni passo temporale sono noti. Sia il vettore \mathbf{b}^n dato da

$$\mathbf{b}^n = \begin{bmatrix} b_1^n \\ \vdots \\ b_{N-1}^n \end{bmatrix},$$

cioè il vettore dei termini noti. Esso assumerà una forma diversa a seconda del problema trattato. Ad esempio, nel caso si usi una discretizzazione del tipo operator-splitting, presentata per le opzioni Europee, nel caso di modello di Lévy si avrà che

$$\mathbf{b}^n = \theta D \mathbf{V}^n + \begin{bmatrix} Y_1^n \\ \vdots \\ Y_{N-1}^n \end{bmatrix} + \boldsymbol{\beta},$$

con $\boldsymbol{\beta}$ vettore delle condizioni al contorno e con \mathbf{Y}^n , vettore rappresenta il valore numerico del termine integrale nell'Equazione (5.16) calcolato come in (5.3). Introduciamo ora la matrice \mathbf{C} di dimensione $(N-1) \times (N-1)$ tridiagonale e simmetrica

$$\begin{bmatrix} \varphi & \vartheta & 0 & \cdots & 0 \\ \varsigma & \varphi & \vartheta & 0 & \cdots \\ 0 & \varsigma & \ddots & \ddots & 0 \\ \vdots & 0 & \ddots & \varphi & \vartheta \\ 0 & \cdots & 0 & \varsigma & \varphi \end{bmatrix},$$

dove, ad esempio nel caso di θ -metodo e discretizzazione non *up-wind*, si ha che

$$\begin{aligned} \varsigma &= \frac{(r - \tilde{\alpha})}{2\Delta S} S_i + \frac{\sigma^2}{2\Delta S^2} S_i^2, \\ \varphi &= \frac{\sigma^2}{\Delta S^2} S_i^2 + (r + \lambda) S_i, \\ \vartheta &= -\frac{(r - \tilde{\alpha})}{2\Delta S} S_i + \frac{\sigma^2}{2\Delta S^2} S_i^2. \end{aligned}$$

In questo modo il *linear complementary problem* assume la forma

$$\begin{aligned} \mathbf{C} \mathbf{V}^{n+1} &\geq \mathbf{b}^n, \quad \mathbf{V}^{n+1} \geq \mathbf{g}^{n+1}, \\ (\mathbf{V}^{n+1} - \mathbf{g}^{n+1}) \cdot (\mathbf{C} \mathbf{V}^{n+1} - \mathbf{b}^n) &= 0. \end{aligned} \quad (5.17)$$

L'unica differenza rispetto alla risoluzione di un sistema lineare è la presenza di una disequaglianza. Tuttavia il Problema (5.17) può essere agevolmente risolto tramite il metodo *SOR* che prevede di porre

$$\begin{aligned} y_i^{n+1,k+1} &= \frac{1}{\varphi} \left(b_i^n - \varsigma V_{i-1}^{n+1,k+1} - \vartheta V_{i+1}^{n+1,k} \right), \\ V_i^{n+1,k+1} &= V_i^{n+1,k} + \omega \left(y_i^{n+1,k+1} - y_i^{n+1,k} \right), \end{aligned} \quad (5.18)$$

ed apportargli una modifica in modo da soddisfare il vincolo secondo il quale $\mathbf{V}^{n+1} \geq \mathbf{g}^n + 1$; A tal proposito è sufficiente sostituire, come proposto da Wilmott in [40], la (5.18) con la seguente

$$V_i^{n+1,k+1} = \max \left(V_i^{n+1,k} + \omega \left(y_i^{n+1,k+1} - V_i^{n+1,k} \right), g_i^{n+1} \right).$$

Il vincolo è imposto nello stesso momento in cui $V_i^{n+1,k+1}$ è calcolato. Così facendo l'effetto del vincolo si ripercuote direttamente sul calcolo di $V_{i+1}^{n+1,k+1}$, $V_{i+1}^{n+2,k+1}$ etc. Il *SOR* proiettato viene ripetuto fino a che la differenza $\|\mathbf{V}^{n+1,k+1} - \mathbf{V}^{n+1,k}\|$ è più piccola di una certa soglia. A quel punto il metodo si arresta e si prende $\mathbf{V}^{n+1} = \mathbf{V}^{n+1,k+1}$.

L'algoritmo del *SOR* proiettato è costruito in modo tale da essere consistente, da garantire l'unicità della soluzione ed il rispetto dei vincoli. Per il calcolo della soluzione del Problema (5.15) è necessario l'utilizzo del *SOR* proiettato se si vuole operare con un metodo implicito, perché le componenti di \mathbf{V}^{n+1} sono tutte relazionate tra loro. Se noi ad ogni iterazione temporale risolvessimo $\mathbf{C}\mathbf{V}^{n+1} = \mathbf{b}^n$ e poi imponessimo il vincolo a ogni componente di \mathbf{V}^{n+1} non sarebbe più garantito in alcun modo che $\mathbf{C}\mathbf{V}^{n+1} \geq \mathbf{b}^n$ o che $(\mathbf{V}^{n+1} - \mathbf{g}^{n+1}) \cdot (\mathbf{C}\mathbf{V}^{n+1} - \mathbf{b}^n) = 0$. L'effetto sarebbe di produrre soluzioni che potrebbero fallire nell'incontrare la frontiera libera oppure che non soddisferebbero la disuguaglianza di Black-Sholes, producendo dei valori della soluzione per i quali sarebbero presenti opportunità di arbitraggio.

La prova rigorosa della convergenza del *SOR* proiettato è complessa e richiede elementi di analisi funzionale: per una discussione più approfondita del *SOR* proiettato si rimanda a [13] mentre una prova della convergenza è fornita in [14]. Vediamo ora come adattare il metodo appena proposto nel caso di regime-switching.

5.2.3 Approccio per il regime-switching

Come abbiamo già detto in precedenza per il caso Europeo, in aggiunta ai termini usuali della *PIDE* della Proposizione 1.13 sono presenti per ogni equazione i i termini derivanti da $\langle Q, \mathbf{V} \rangle = q_{i1}V_1 + \dots + q_{i1}V_n$. Il problema si affronta esattamente come abbiamo fatto nella Sezione 5.1.4. Abbiamo allora che il termine \mathbf{Y}^n , che rappresenta il valore numerico del termine integrale nell'Equazione (5.16), ovvero la parte che andrà trattata esplicitamente, va modificato come segue:

$$(YV^n)_i = (YV^n)_i + \langle Q, \mathbf{V}^n \rangle_i, \quad (5.19)$$

dove Q è il generatore della catena di Markov che regola lo switch e \mathbf{V}^n è la soluzione al passo n .

Anche in questo caso, ricordando che, in assenza di dividendi, il prezzo di una Call Europea e di una Call Americana coincidono, effettuiamo un

esperimento numerico per verificare la correttezza del *SOR* proiettato nel caso in cui il modello sia quello di Black-Scholes e sia presente switching sulla volatilità. Scegliamo come parametri $S_t = 100$, $K = 100$, $r = 0.1$, $\sigma_1 = 0.4$, $\sigma_2 = 0.2$, $Q_{11} = -2$, $Q_{22} = -1$, $T = 1$ e consideriamo come soluzione esatta il prezzo ottenuto con la formula esplicita presentata nel Capitolo 2. Come tolleranza per il *PSOR* scegliamo $TOLL = 10^{-3}$ e $\omega = 1.2$. Da quanto riportato in Tabella 5.5 possiamo osservare come, all’infittirsi delle griglie spaziale e temporale, la soluzione calcolata con il metodo *PSOR* converga alla soluzione esatta.

		Primo stato			Secondo stato		
N	M	DI	$PSOR$	Err	DI	$PSOR$	Err
50	10	17.4307	17.5938	-0.1631	15.1089	15.2952	-0.1863
100	30	17.4307	17.4503	-0.0196	15.1089	15.1285	-0.0196
200	50	17.4307	17.4416	-0.0108	15.1089	15.1216	-0.0127
500	70	17.4307	17.4312	-0.0005	15.1089	15.1099	-0.0010
1000	100	17.4307	17.4308	-0.0001	15.1089	15.1094	-0.0005

Tabella 5.5: Convergenza del *SOR* proiettato modello di Black-Scholes regime-switching.

Esperimenti del tutto analoghi, i cui risultati sono riportati in Tabella 5.6, mostrano che il metodo del *SOR* proiettato fornisce risultati corretti anche ponendosi nel modello di Merton jump-diffusion regime switching e prezzando una Call Americana senza dividendi. Come prezzo esatto si è considerato quello fornito dalla *FFT* mentre i parametri sono i medesimi all’esperimento precedente a cui si aggiungono i parametri del modello jump-diffusion cioè $\lambda = 1$, $\mu = 0.1$ e $\delta = 0.2$.

		Primo stato			Secondo stato		
N	M	FFT	$PSOR$	Err	FFT	$PSOR$	Err
50	10	20.0249	19.7932	0.2317	18.0695	17.8738	0.1958
100	30	20.0249	19.9263	0.0986	18.0695	17.9804	0.0891
200	50	20.0249	19.9253	0.0997	18.0695	17.9686	0.1009
500	70	20.0249	19.9439	0.0811	18.0695	17.9862	0.0834
1000	100	20.0249	19.9684	0.0566	18.0695	18.0119	0.0577

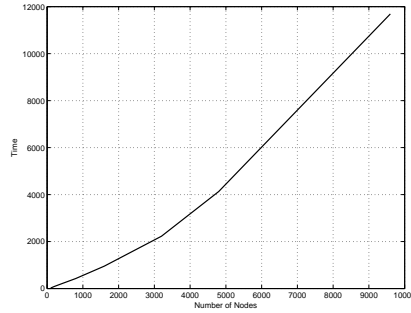
Tabella 5.6: Convergenza del *SOR* proiettato modello Merton jump-diffusion.

Nel Capitolo 7 approfondiremo gli effetti del regime-switching e condurremo ulteriori prove numeriche: analizzeremo dettagliatamente anche il tempo computazionale necessario per l’esecuzione dell’algoritmo, spiegando come esso vari al variare del numero di nodi di discretizzazione.

Capitolo 6

Metodo delle linee

Come abbiamo osservato più volte nella nostra trattazione, non esistono soluzioni analitiche per il problema di frontiera libera che descrive il prezzo di un'opzione Americana Put o Call (scritta su un sottostante che distribuisca dividendi). Nel corso degli anni sono state proposte diverse procedure numeriche per risolvere questo problema. Ci siamo soffermati sul metodo delle differenze finite che ci ha portato a derivare lo schema numerico del *SOR* proiettato, tuttavia tale metodo presenta alcuni svantaggi. Ad esempio il metodo delle differenze finite necessita di una gran quantità di punti di discretizzazione per produrre un risultato accurato, è difficilmente parallelizzabile a causa della sua natura sequenziale ed, inoltre, la matrice potrebbe risultare mal condizionata e quindi di difficile inversione. Tuttavia, in un mondo come quello della finanza in cui la rapidità di decisione ed azione è fondamentale, il più grande svantaggio che si presenta è la quantità di tempo necessaria per prezzare una singola opzione come si evince dalla Tabella 6.1. Da qui la necessità di avere a disposizione uno schema numerico che richieda un tempo di calcolo accettabile e che sia sufficientemente accurato. Un metodo numerico che risponde a queste esigenze è il metodo delle linee (*MOL*). Il metodo delle linee è un metodo numerico per la risoluzione di equazioni alle derivate parziali introdotto per la prima volta nei primi anni Sessanta [37]. Esso consiste nel discretizzare tutte le dimensioni tranne una, riducendosi a risolvere un'equazione alle differenze ordinarie (*ODE*), problema per il quale, nel corso degli anni, sono stati sviluppati una vasta gamma di metodi numerici accurati disponibili in molti pacchetti di calcolo, anche open source. In questo capitolo presenteremo il metodo generale, dopodiché lo applicheremo al caso in cui il sottostante evolva secondo un modello di Lévy in ipotesi di regime-switching. Una trattazione completa del problema può essere trovata in [31], [32], [11], [30].



N	$PSOR$	$Tempo(sec.)$
100	19.8271	5.8846
200	19.8385	34.8518
400	19.8386	227.2980
800	19.8377	957.4258
1600	19.8381	6708.0449

Figura 6.1: Tempi $PSOR$ con $M = 100$. Put Americana nel modello di Black-Scholes. Parametri: $r = 0.14, \sigma = 0.5, T = 1, S_t = 100, K = 110$.

6.1 Il metodo

Come già anticipato, il metodo delle linee consiste nell'approssimazione di una o più equazioni alle derivate parziali con una o più equazioni alle derivate ordinarie in una delle variabili indipendenti. Questo approccio è motivato dall'assunzione che le equazioni alle derivate ordinarie sono più facili da analizzare e da risolvere. L'approssimazione delle derivate può avvenire con espansioni in serie di Fourier, elementi finiti oppure con differenze finite: in linea con quanto fatto nel Capitolo 5, adotteremo quest'ultimo approccio.

Per introdurre i concetti fondamentali per le applicazioni successive consideriamo un'equazione parabolica del tipo

$$a(x, t) u_{xx} + b(x, t) u_x - c(x, t) u - d(x, t) u_t = f(x, t), \quad (6.1)$$

con condizioni iniziali e al bordo date da

$$\begin{aligned} u(x, 0) &= u_0(x), \\ u(0, t) &= \alpha(t), & u(L, t) &= \beta(t). \end{aligned}$$

dove

$$\begin{aligned} u_{xx} &= \frac{\partial^2 u}{\partial x^2}, & u_x &= \frac{\partial u}{\partial x}, \\ u_t &= \frac{\partial u}{\partial t}. \end{aligned}$$

Consistentemente con le nostre applicazioni supporremo che

$$\begin{aligned} a(x, t) &> 0, & 0 < x < L, \quad \forall t, \\ d(x, t) &\geq 0, & 0 < x < L, \quad \forall t. \end{aligned}$$

Generalmente nella trattazione di problemi ingegneristici, il metodo delle linee si riferisce ad un'equazione alle differenze ordinaria in τ (tale metodo è detto anche *vertical MOL*). Tuttavia in finanza non si hanno casi di discretizzazione spaziale: per opzioni con payoff discontinuo, salti nella dinamica del sottostante, possibilità di esercizio anticipato la soluzione dell'equazioni di Black-Scholes può variare molto per ogni x ad un tempo fissato [30]. Per questo motivo è opportuno discretizzare il tempo e di risolvere equazioni alle derivate ordinarie in spazio (*orizental MOL*).

Supponiamo che il Problema (6.1) debba essere risolto sull'intervallo temporale $[0, T]$, dove T è fissato arbitrariamente. Introduciamo una partizione sull'asse temporale

$$0 = t_0 < t_1 < \dots < t_N = T,$$

e poniamo $\Delta t_n = t_n - t_{n-1}$ che verrà, di solito, assunto costante. Per il metodo delle linee applicato al Problema (6.1) sostituiamo semplicemente u_t con una differenza finita implicita. Generalmente è possibile utilizzare due tipi di approssimazione. Se $u_n(x)$ denota l'approssimazione della soluzione $u(t_n, x)$ allora si può utilizzare un'approssimazione di Eulero all'indietro

$$u_t(x, t_n) \approx Du_n(x) = \frac{u_n(x) - u_{n-1}(x)}{\Delta t_n}, \quad (6.2)$$

oppure un'approssimazione a tre punti del tipo

$$u_t(x, t_n) \approx Du_n(x) = \bar{c}_n \frac{u_n(x) - u_{n-1}(x)}{\Delta t_n} + \bar{d}_n \frac{u_{n-1}(x) - u_{n-2}(x)}{\Delta t_{n-1}}, \quad (6.3)$$

dove \bar{c}_n e \bar{d}_n sono opportuni pesi che, nel caso lo step temporale sia costante assumono i valori $\bar{c}_n = \frac{3}{2}$ e $\bar{d}_n = -\frac{1}{2}$. In particolare si ha che il metodo di Eulero costituisce un'approssimazione in tempo del primo ordine mentre il metodo a tre punti garantisce che l'approssimazione sia di secondo ordine in tempo. Il metodo delle linee applicato al Problema (6.1) al tempo $t = t_n$ porta a

$$a(t_n, x) u_n''(x) + b(t_n, x) u_n'(x) - c(t_n, x) u_n(x) - d(t_n, x) Du_n(x) = f(x, t_n), \quad (6.4)$$

con condizione iniziale e condizioni al bordo seguenti

$$\begin{aligned} u_0(x) &= g(x), \\ u_n(0) &= \alpha(t_n), & u_n(L) &= \beta(t_n). \end{aligned}$$

E' un fatto ben noto che un metodo numerico basato su differenze finite di Eulero all'indietro o su uno schema di discretizzazione in tempo a tre livelli è incondizionatamente stabile. Quindi possiamo dedurre che sia (6.2) che (6.3)

conducono ad un metodo numerico stabile per la soluzione dell'equazione (6.1). Nelle implementazioni una possibile scelta è di usare un'approssimazione di Eulero all'indietro per $n = 1$ ed $n = 2$ e poi passare ad una approssimazione a tre nodi per $n \geq 3$.

Ricordiamo che è possibile scrivere un'equazione differenziale ordinaria di ordine N come sistema lineare di N equazioni differenziali del primo ordine. Ponendo

$$\begin{aligned} u(x) &= u_n(x), \\ v(x) &= u'_n(x), \end{aligned}$$

allora il Problema (6.4) può essere riscritto, scelta ad esempio un'approssimazione al primo ordine, come

$$\begin{aligned} u'(x) &= v, \\ v'(x) &= \frac{1}{a(t_n, x)} \left[\left(c(t_n, x) + \frac{1}{\Delta t_n} \right) u - b(t_n, x) v + f(t_n, x) - \frac{1}{\Delta t_n} u_{n-1}(x) \right], \\ u(0) &= \alpha(t_n), \quad u(L) = \beta(t_n). \end{aligned} \tag{6.5}$$

L'analoga formulazione per lo schema di secondo ordine può essere ricavata utilizzando (6.3) in (6.1).

Algoritmi per la soluzione di questi problemi sono stati derivati da vari autori. Il primo ad affrontare questo tipo di questioni fu il conte Vincenzo Riccati (1707-1775) motivo per cui il metodo seguente prende il nome di *metodo di trasformazione di Riccati*. Tale idea può essere estesa ad un sistema di N equazioni differenziali ordinarie del primo ordine. Tuttavia, dato che in tutta la trattazione avremo a che fare con sistemi di due *ODE*, introdurremo la trasformazione di Riccati solo per sistemi di due equazioni differenziali del primo ordine scalari.

6.1.1 La trasformazione di Riccati su un intervallo fissato

Le Equazioni (6.5) sono un caso particolare del seguente problema più generale

$$\begin{aligned} u'(x) &= A(x)u + B(x)v + f(x), & u(s) &= \Gamma v(s) + \alpha, \\ v'(x) &= C(x)u + D(x)v + g(x), & G(u(S), v(S)) &= 0, \end{aligned} \tag{6.6}$$

dove Γ e α sono costanti e G è una funzione di due variabili generale, che può essere, eventualmente, non lineare. Non assumiamo che $s < S$ perchè questo può non essere vero: ciò che è essenziale, però, è che una delle condizioni al bordo sia lineare affine e nella nostra trattazione richiediamo che ciò si verifichi per $x = s$. Il metodo che presentiamo è valido per il Problema (6.6) e quindi funzionerà per la risoluzione con il *MOL* in x per una generica

equazione di diffusione lineare.

E' nell'essenza del metodo, perchè possa essere applicato, che le funzioni $\{u(x), v(x)\}$ siano legate dalla trasformazione di Riccati

$$u(x) = R(x) v(x) + w(x), \quad (6.7)$$

dove le funzioni R e w sono soluzioni dei ben definiti problema ai valori iniziali

$$\begin{aligned} R' &= B(x) + A(x)R - D(x)R - C(x)R^2, & R(s) &= \Gamma, \\ w' &= [A(x) - C(x)R(x)]w - R(x)g(x) + f(x), & w(s) &= \alpha. \end{aligned} \quad (6.8)$$

Grazie alle ipotesi introdotte sui coefficienti di (6.1) i coefficienti sono funzioni continue di x . La teoria dei problemi ai valori iniziali assicura che l'equazione per R , la così detta equazioni di Riccati, abbia un'unica soluzione locale e che l'equazione lineare non omogenea per w abbia una soluzione finchè $R(x)$ esiste. Per il Problema (6.6) derivato dal *MOL* per l'equazione di diffusione, l'equazione di Riccati ammette una soluzione uniformemente limitata e quindi il problema ai valori iniziali (6.8) ha un'unica soluzione su tutto l'intervallo considerato. Chiameremo l'integrazione da $x = s$ a $x = S$ delle Equazioni (6.8) *forward sweep*.

Assumiamo ora che $w(S)$ e $R(S)$ siano state trovate. Usando la trasformazione di Riccati (6.7) nella condizione al contorno $G(u(S), v(S)) = 0$, osserviamo che $v = v(S)$ deve essere scelta in modo che

$$G(R(S)v + w(S), v) = 0.$$

Questa è un'equazione scalare. Ogni soluzione v^* di questa equazione definisce una soluzione di (6.6) ottenuta ponendo

$$v(s) = v^*,$$

e integrando tra $x = S$ e $x = s$ l'*ODE* seguente

$$v' = [C(x)R(x) + D(x)]v + C(x)w(x) + g(x), \quad v(S) = v^*. \quad (6.9)$$

Quest'ultima integrazione da $x = S$ a $x = s$ è detta *reverse sweep*. Una volta nota $v(x)$ allora $u(x) = R(x)v(x) + w(x)$ è una soluzione del problema (6.6).

Fondamentalmente la trasformazione di Riccati consta in tre passi: il *forward sweep*, la determinazione dei valori al contorno $u(S), v(S)$ ed infine il *reverse sweep*. In sostanza, il problema iniziale con due valori al bordo è stato trasformato in due problemi ben definiti ai valori iniziali. Il successo della trasformazione di Riccati per problemi di diffusione discreti in tempo è dovuto alle seguenti proprietà generali.

- La soluzione dell'equazione di Riccati si comporta come $\tanh(x)$ e quindi è continua e limitata.
- La soluzione fondamentale dell'equazione differenziale per w è esponenzialmente decrescente e quindi facilmente calcolabile numericamente.
- La seconda condizione al contorno è scorrelata dal *forward sweep*. Solo i valori numerici di $R(S)$ e $w(S)$ sono richiesti.
- Una volta che la condizione al bordo per la seconda equazione è stata trovata, l'Equazione (6.9) ammette ancora una volta una soluzione fondamentale esponenzialmente decrescente nel verso di integrazione.

6.1.2 La trasformazione di Riccati per un problema di frontiera libera

Quello che rende attrattiva la trasformazione di Riccati per i problemi di frontiera libera è l'indipendenza di quello che abbiamo chiamato *forward sweep* dalla condizione al bordo per $x = S$. Consideriamo ancora il Problema (6.6) con la condizione al bordo lineare affine

$$u(s) = \Gamma v(s) + \alpha,$$

e le condizioni di frontiera libera

$$\begin{aligned} G_1(u(S), v(S), S) &= 0, \\ G_2(u(S), v(S), S) &= 0, \end{aligned}$$

per un certo S non noto a priori. Dal problema in esame è noto se S è maggiore o minore di s . Nella trattazione assumeremo che $S > s$. Tre condizioni al contorno sono necessarie per un'equazione differenziale del secondo ordine (e per il sistema equivalente di *ODE* del primo ordine), per essere in grado di determinare le costanti e la frontiera libera. Le Equazioni (6.8) sono indipendenti dalla frontiera libera quindi vengono integrate come nel caso precedente. Assumiamo che $R(X)$ e $w(X)$ siano note su un certo intervallo (s, X) . Se X fosse la frontiera efficiente allora deve soddisfare

$$\begin{aligned} u(X) &= R(X)v(X) + w(X), \\ G_1(u(X), v(X), X) &= 0, \\ G_2(u(X), v(X), X) &= 0. \end{aligned}$$

Si hanno tre equazioni in tre incognite u, v e X . Possiamo eliminare u e ridurre il sistema a

$$\begin{aligned} G_1(R(X)v + w(X), v, X) &= 0, \\ G_2(R(X)v + w(X), v, X) &= 0. \end{aligned}$$

Nella maggior parte delle applicazioni è possibile eliminare anche v e ridursi ad una sola equazione scalare della forma

$$\phi(X, R(X), w(X)) = 0.$$

Integriamo quindi in avanti le equazioni (6.8) da $x = s$ e teniamo sotto controllo il segno di ϕ . Quando si è in presenza di uno zero lì si trova la frontiera libera. Nota S , possiamo integrare anche (6.9) da $x = S$ a $x = s$. La sua soluzione $v(x)$, insieme a $R(x)$ e $w(x)$, consente di calcolare $u(x)$. E' possibile che ϕ abbia più di una radice, ognuna delle quali conduce ad una diversa soluzione del problema di frontiera libera. Quale sia quella corretta da scegliere sarà brevemente discusso nel seguito della trattazione.

6.2 Soluzione numerica delle equazioni

Introduciamo ora i metodi numerici per la risoluzione delle equazioni alle derivate ordinarie per R , u e v viste in precedenza.

Il problema che dobbiamo affrontare è della forma

$$u' = F(x, u), \quad u(x_0) = u_0. \quad (6.10)$$

Potremmo essere tentati di risolvere le *ODE* del metodo delle linee utilizzando metodi adattativi di grande precisione e già disponibili in molte librerie matematiche. Oppure, cosa ancora più invitante, usare dei software che permettono la risoluzione di sistemi di equazioni differenziali del primo ordine, esattamente quello che dobbiamo fare. Tale approccio, seppur invitante, non risula essere, almeno per i problemi che vogliamo affrontare, il migliore possibile. Il problema è che le soluzioni numeriche sono disponibili solo in alcuni punti ed inoltre i solver per le *ODE* richiedono una funzione che valuti $F(x, u)$ e quindi i nostri dati dovranno essere interpolati in accordo alla precisione dell'integratore. Interpolare più e più volte un insieme discreto di punti richiede un'elevata complessità computazionale e fa perdere velocità al metodo proposto.

Una possibile strada da percorrere per ottenere risultati sufficientemente accurati, guadagnando considerevolmente in tempo, è applicare la regola dei trapezi come descritto in [30]. Il metodo è semplice, implicito, stabile ed inoltre è possibile ottenere qualsiasi ordine di accuratezza semplicemente infittendo la griglia spaziale. Ricordiamo il metodo dei trapezi per un sistema della forma (6.10). Definiamo una partizione non necessariamente equispaziata

$$x_0 < \cdots < x_J,$$

sull'intervallo di integrazione dove

$$x_0 = \min\{s, S\}, \quad x_J = \max\{s, S\}.$$

Data allora una soluzione u_{j_0} nel punto x_{j_0} possiamo trovare la soluzione u_j nel punto adiacente x_j da

$$\frac{u_j - u_{j_0}}{x_j - x_{j_0}} = \frac{1}{2} [F(x_j, u_j) + F(x_{j_0}, u_{j_0})].$$

Si noti che x_j può essere sia a destra che a sinistra di x_{j_0} . Quando la regola dei trapezi è applicata all'equazione di Riccati si ottiene

$$\begin{aligned} \frac{R_j - R_{j_0}}{x_j - x_{j_0}} = \frac{1}{2} \{ [B_j + (A_j - D_j) R_j - C_j R_j^2] \\ + [B_{j_0} + (A_{j_0} - D_{j_0}) R_{j_0} - C_{j_0} R_{j_0}^2] \}, \end{aligned}$$

dove $A_j = A(x_j)$. Questa è un'equazione di secondo grado delle forma

$$\alpha_j R_j + \beta_j R_j + \gamma_j = 0,$$

dove

$$\begin{aligned} \alpha_j &= \frac{C_j}{2}, \\ \beta_j &= \frac{1}{x_j - x_{j_0}} - \frac{(A_j - D_j)}{2}, \\ \gamma_j &= \frac{C_{j_0}}{2} R_{j_0}^2 - \left[\frac{1}{x_j - x_{j_0}} + \frac{(A_{j_0} - D_{j_0})}{2} \right] R_{j_0} - \frac{B_j + B_{j_0}}{2}. \end{aligned} \quad (6.11)$$

L'equazione è risolta dalla formula

$$R_j = \frac{-\beta_j \pm \sqrt{\beta_j^2 - 4\alpha_j\gamma_j}}{2\alpha_j}. \quad (6.12)$$

Fino ad ora il segno algebrico di $x_j - x_{j_0}$, ovvero il verso di integrazione, non è stato considerato: tuttavia esso è determinante per la scelta di R_j in (6.12). In tutte le seguenti applicazioni risulterà $B(x) > 0$ per ogni x . Segue, dall'equazione di Riccati, che $R'(x_0) = B(x_0) > 0$ ogni volta che $R(x_0) = 0$ e quindi $R(x) > 0$ per ogni $x > x_0$ e $R(x) < 0$ per ogni $x < x_0$. Se $0 < x_j - x_{j_0} \rightarrow 0$ allora $\beta_j \rightarrow \infty$ e $\gamma_j < 0$. Segue che $R_j > 0$ se e solo se in (6.12) viene scelto il ramo positivo. Similmente se $0 > x_j - x_{j_0}$ allora $\beta_j \rightarrow -\infty$ e $\gamma_j < 0$. In questo caso $R_j < 0$ richiede il ramo negativo in (6.12). Osserviamo, inoltre, che per piccoli valori del passo temporale (6.12) richiede la sottrazione di numeri molto grandi e simili tra loro: questo potrebbe portare ad una perdita di accuratezza dovuta alla loro sottrazione ed è per questo motivo che è preferibile riscrivere la (6.12) come

$$R_j = \frac{-2\gamma_j}{\beta_j \pm \sqrt{\beta_j^2 - 4\alpha_j\gamma_j}}.$$

Nota R è possibile ricavare w dalla seguente equazione lineare algebrica:

$$\frac{w_j - w_{j_0}}{x_j - x_{j_0}} = \frac{1}{2} \{ [A_j - C_j R_j] w_j + [A_{j_0} - C_{j_0} R_{j_0}] w_{j_0} - (R_j g_j + R_{j_0} g_{j_0}) + f_j + f_{j_0} \}. \quad (6.13)$$

Similmente, noti R e w , otteniamo v da:

$$\frac{v_j - v_{j_0}}{x_j - x_{j_0}} = \frac{1}{2} \{ [C_j R_j + D_j] v + [C_{j_0} R_{j_0} + D_{j_0}] v_{j_0} + (C_j w_j + C_{j_0} w_{j_0}) + g_j + g_{j_0} \}. \quad (6.14)$$

Sebbene tale metodo sia stabile c'è di fatto una limitazione: il passo di discretizzazione spaziale va scelto in modo tale che (6.12) abbia radici reali. Inoltre, esperimenti numerici condotti mostrano come sia necessario che il passo di discretizzazione spaziale sia sufficientemente piccolo di modo che le soluzioni analitiche di

$$\begin{aligned} w' &= [A(x) - R(x) C(x)] w \quad \text{per} \quad A(x) - R(x) C(x) \neq 0, \\ v' &= [C(x) R(x) + D(x)] v \quad \text{per} \quad C(x) R(x) + D(x) \neq 0, \end{aligned}$$

siano approssimate da soluzione numeriche $\{w_j\}, \{v_j\}$ non oscillanti. In particolare da (6.13) si può osservare che

$$w_j = \frac{1 + \frac{x_j - x_{j_0}}{2} (A_{j_0} - C_{j_0} R_{j_0})}{1 - \frac{x_j - x_{j_0}}{2} (A_j - C_j R_j)} w_{j_0} \equiv \mathcal{M}_j w_{j_0},$$

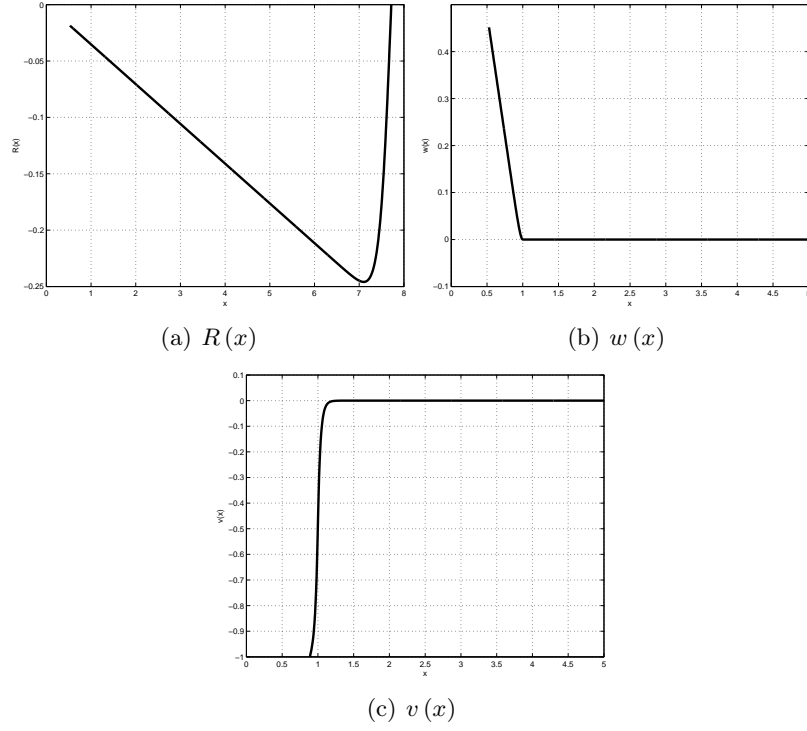
w_j è monotono in j se $\mathcal{M}_j > 0$. Analogamente per il *reverse sweep* abbiamo che

$$v_j = \frac{1 + \frac{x_j - x_{j_0}}{2} (C_{j_0} R_{j_0} + D_{j_0})}{1 - \frac{x_j - x_{j_0}}{2} (C_j R_j + D_j)} v_{j_0} \equiv \mathcal{N}_j v_{j_0}.$$

v_j non oscillerà se $\mathcal{N}_j > 0$. Se uno di questi due vincoli dovesse essere violato potrebbe accadere che nelle soluzioni numeriche di w e v si presentino oscillazioni con una conseguente perdita di accuratezza nel metodo. Nelle applicazioni numeriche faremo in modo che tali vincoli siano rispettati e rimandiamo a [30] per i dettagli. In Figura 6.2 sono rappresentate le soluzioni numeriche di R, w, v .

Concludiamo la sezione discutendo su come localizzare la frontiera efficiente. Dalla trattazione svolta sappiamo che la frontiera efficiente è una radice dell'equazione scalare

$$\phi(x, R(x), w(x)) = 0.$$

Figura 6.2: Soluzione numerica delle equazioni di R , w , v .

Assumendo che la frontiera libera sia localizzata da qualche parte in $[x_0, x_J]$, una volta noti numericamente R e w , monitoriamo il segno di $\phi(x_j, R_j, w_j)$. Se cambia tra x_j ed x_{j+1} allora $S_f(t)$ è tra questi due punti. Siccome il *reverse sweep* deve partire da $S_f(t)$ allora devono essere noti i valori di R , w nel punto $S_f(t)$: è quindi opportuno che il punto sia uno dei punti della griglia. Si può attuare come scelta quella di scegliere il punto x_{j+1} come frontiera libera. Se la griglia è sufficientemente fitta questa approssimazione non porta ad errori rilevanti: alternativamente si può trovare $S_f(t)$ con uno qualsiasi dei metodi di interpolazione ed in seguito fornire, sempre tramite interpolazione, il valori di w ed R in $S_f(t)$.

Ad ogni passo temporale è quindi richiesta la soluzione di due equazioni differenziali ordinarie e la ricerca dello zero di una funzione definita in un numero discreto di punti. L'equazione di Riccati, invece, è integrata una sola volta e memorizzata. Generalmente può capitare, per un problema di frontiera libera qualsiasi, che siano presenti più zeri nella funzione $\phi(x)$: tuttavia, nelle nostre applicazioni, riscontreremo che tale funzione cambia segno una sola volta. Nella prossima sezione l'algoritmo appena presentato verrà presentato nel caso specifico in cui il sottostante evolva seguendo una dinamica di tipo Merton jump-diffusion.

6.3 L'algoritmo

Consideriamo il caso in cui il sottostante evolva secondo un modello di Lévy di tipo jump-diffusion. Presenteremo l'algoritmo per il caso in cui non vi sia regime-switching per poi estenderlo facilmente al caso in cui il passaggio da uno stato dell'economia ad un altro sia possibile. Una trattazione più approfondita può essere trovata in [30]. Prendiamo in esame un'opzione Put Americana, indicata con $V(S, t)$, con maturity T e strike price K . Come detto in precedenza per ogni $t > 0$, esiste un unico $S_f(t)$ tale che

$$\begin{aligned} \frac{\partial V}{\partial t} + L^S V(S, t) &= 0, \quad S > S_f(t), \quad t \in (0, T], \\ V(S, t) &\rightarrow (K - S)^+, \quad t \rightarrow T, \\ V(S, t) &\rightarrow 0, \quad S \rightarrow \infty, \\ V(S, t) &> (K - S)^+, \quad S > S_f(t), \\ V(S, t) &\rightarrow (K - S)^+, \quad S \rightarrow S_f(t)^+, \\ \frac{\partial V}{\partial S}(S, t) &\rightarrow -1, \quad S \rightarrow S_f(t)^+, \end{aligned}$$

dove si è posto

$$\begin{aligned} L^S &= (r - \alpha) S \frac{\partial V}{\partial S} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} - (r + \lambda) V + \int_{-\infty}^{\infty} \nu(dy) V(t, S e^y), \\ \alpha &= \int_{-\infty}^{+\infty} (e^y - 1) \nu(dy), \\ \lambda &= \int_{-\infty}^{+\infty} \nu(dy). \end{aligned}$$

In questo caso risulta utile introdurre il cambio di variabile seguente

$$u = \frac{V}{K}, \quad x = \frac{S}{K}, \quad \tau = T - t.$$

Poniamo, inoltre, per comodità di notazione

$$J = \int_{-\infty}^{\infty} \nu(dy) \frac{V(t, S e^y)}{K}.$$

Otteniamo così il problema di frontiera libera

$$\sigma(x, \tau) u_{xx} + b(x, \tau) u_x - r(x, \tau) u + J - u_\tau = 0, \quad (6.15)$$

se $x > S_f(\tau)$ e $0 < \tau \leq T$, in unione a

$$\begin{aligned} u(S_f(\tau) +, \tau) &= 1 - S_f(\tau), & 0 < \tau \leq T, \\ u_x(S_f(\tau) +, \tau) &= -1, & 0 < \tau \leq T, \\ u(x, \tau) &\rightarrow 0, \quad x \rightarrow \infty, & 0 < \tau \leq T, \\ u(x, 0) &= 0, \quad x \geq 1. \end{aligned}$$

I coefficienti $\sigma(x, \tau)$, $b(x, \tau)$, $r(x, \tau)$, nel nostro modello, sono dati da

$$\begin{aligned} \sigma(x, \tau) &= \frac{\sigma^2}{2} x^2, \\ b(x, \tau) &= (r - \alpha)x, \\ r(x, \tau) &= r + \lambda. \end{aligned}$$

A questo punto si può procedere con l'algoritmo di soluzione. Tronchiamo innanzi tutto il dominio, lavorando con un intervallo finito della forma $[S_\infty, X]$, dove la scelta di S_∞ sarà discussa tra poco. La condizione per $S \rightarrow \infty$ verrà sostituita da

$$u(X, \tau) = 0.$$

A questo punto siamo pronti per introdurre la discretizzazione in tempo ad ogni istante di tempo τ_n . Se scegliessimo un'approssimazione al primo ordine l'equazione alle derivate parziali diventerebbe

$$\sigma(x, \tau_n) u'' + b(x, \tau_n) u' - r(x, \tau_n) u + J - \frac{u - u_{n-1}}{\Delta\tau} = 0, \quad (6.16)$$

dove $u \equiv u_n$ è l'approssimazione di $u(x, \tau_n)$. Altrimenti, è possibile scegliere uno schema di approssimazione del secondo ordine: in tal caso l'equazione diventa

$$\begin{aligned} \sigma(x, \tau_n) u'' + b(x, \tau_n) u' - r(x, \tau_n) u + J \\ - \frac{3}{2\Delta\tau} (u - u_{n-1}) + \frac{1}{2\Delta\tau} (u_{n-1} - u_{n-2}) = 0. \end{aligned} \quad (6.17)$$

In entrambi i casi il problema iniziale dipendente dal tempo è sostituito da una serie di problemi di frontiera libera per equazioni alle differenze ordinarie. Al fine di evitare problemi numerici per $x = 0$ è più opportuno considerare $\sigma(x, \tau_n) = \max\{10^{-6}, \sigma(x, \tau_n)\}$. Per semplicità di notazione riscriviamo (6.16) e (6.17) genericamente come

$$u'' + d(x, \tau_n) u' - c(x, \tau_n) u = g(x, \tau_n) \quad (6.18)$$

dove c, d, g possono essere ottenuti comparando l'Equazione (6.18) con (6.16) e (6.17). Come abbiamo già fatto per nell'approccio alle differenze finite decidiamo di trattare in maniera esplicita il termine integrale J calcolato come nel Capitolo 5. Procedendo ora come nella precedente sessione riscriviamo (6.18) in forma di sistema del primo ordine

$$\begin{aligned} u' &= v, \\ v' &= c(x, \tau_n) u - d(x, \tau_n) v + g(x, \tau_n). \end{aligned}$$

Come abbiamo già osservato è possibile operare una trasformazione di Riccati ottenendo così

$$u(x) = R(x) v(x) + w(x), \quad (6.19)$$

dove R e w sono le soluzioni dei seguenti problemi ai valori iniziali ben definiti

$$\begin{aligned} R' &= 1 + d(x, \tau_n) R - c(x, \tau_n) R^2, \quad R(X) = 0, \\ w' &= -c(x, \tau_n) R(x) w - R(x) g(x, \tau_n), \quad w(X) = 0. \end{aligned}$$

Una volta note R e w su (S_∞, X) è possibile calcolare la frontiera libera $S_f(t) = S_n$ al tempo τ_n . S_∞ è la posizione iniziale della frontiera libera: il modo in cui può essere scelta varia a seconda del problema trattato. Se si ha a che fare con il modello di Black-Scholes una buona idea è sceglierla di modo che per $\tau \rightarrow \infty$ lo schema numerico riproduca il comportamento di una Put America Perpetua la cui frontiera libera è data da

$$s_\infty = \frac{\gamma}{\gamma - 1},$$

dove

$$\gamma = \frac{-(b - \sigma^2/2) - \sqrt{(b - \sigma^2/2)^2 + 2r\sigma^2}}{\sigma^2}.$$

Non essendo note formule chiuse per la frontiera efficiente nel caso di modello di Lévy decidiamo di porre $S_\infty = S_{min}/K$ come proposto in [31]. Esperimenti numerici dimostrano che questa assunzione è valida nella maggior parte dei casi. Come abbiamo osservato nella sessione precedente la frontiera libera dovrà essere scelta in modo da essere una radice della funzione

$$\phi(x) = R(x) - w(x)(1 - x).$$

Ogni punto S_f tale che $\phi(S_f) = 0$ è una frontiera libera ammissibile al tempo τ_n . Qualora vi fossero più frontiere libere, tenendo conto che generalmente la frontiera libera è monotona crescente, S_{f_n} sarà la prima radice a sinistra di $S_{f_{n-1}}$. Nota S_{f_n} la soluzione verrà trovata dopo aver integrato numericamente

$$v' = c(x, \tau_n) (R(x) v + w(x)) - d(x, \tau_n) v + g(x, \tau_n),$$

con $v(S_{f_n}) = -1$ e sostituendo il risultato in (6.19) ottenendo così $u(x)$ al tempo τ_n . La soluzione viene poi estesa linearmente su $[S_\infty, S_{f_n}]$. Fatto tutto ciò è possibile avanzare al prossimo step temporale.

Analizziamo ora il caso in cui sia presente regime-switching. Fortunatamente l'estensione dell'algoritmo è quasi immediata. Ricordiamo che, in presenza di switching tra N stati, si deve risolvere un sistema di equazioni alle derivate parziali integro-differenziali del tipo:

$$\begin{aligned} \frac{\partial V_i}{\partial t} + L^S V_i(S, t) + \langle \mathbf{V}, Q \rangle &= 0, \quad S > S_f(t), \quad t \in (0, T], \\ V_i(S, t) &\rightarrow (K - S)^+, \quad t \rightarrow T, \\ V_i(S, t) &\rightarrow 0, \quad s \rightarrow \infty, \\ V_i(S, t) &> (K - S)^+, \quad S > S_f(t), \\ V_i(S, t) &\rightarrow (K - S)^+, \quad S \rightarrow S_f(t) +, \\ \frac{\partial V_i}{\partial S}(S, t) &\rightarrow -1, \quad S \rightarrow S_f(t) +, \quad i = 1, \dots, N, \end{aligned}$$

dove Q è il generatore della catena di Markov, mentre \mathbf{V} è il vettore delle soluzioni. Tutto resta invariato fatta eccezione per il termine J dell'Equazione (6.15) che va sostituito con

$$\tilde{J}_i = \int_{-\infty}^{\infty} \nu(dy) \frac{V_i(t, Se^y)}{K} + q_{i,1} \frac{V_1}{K} + \dots + q_{i,N} \frac{V_N}{K}, \quad i = 1, \dots, N.$$

Il termine integrale, qualora si decida di trattarlo in maniera esplicita, può essere calcolato numericamente come mostrato nel Capitolo 5. Notiamo che, anche i termini di accoppiamento, scelto un'approccio esplicito, sono semplici da trattare: infatti, supponendo di essere al tempo $n+1$ abbiamo la soluzione \mathbf{V} al passo n è nota e quindi lo è anche $q_{i,1} \frac{V_1^n}{K} + \dots + q_{i,N} \frac{V_N^n}{K}$ permettendoci di calcolare \tilde{J} .

Verifichiamo ora che il metodo delle linee è di gran lunga più veloce del SOR proiettato. In Tabella 6.1 sono riportati i tempi necessari per il pricing di un'opzione Put Americana nel modello di Black-Scholes standard. Si può osservare come, a parità di discretizzazione spaziale, il metodo delle linee sia più rapido del SOR proiettato, soprattutto se il numero di punti di discretizzazione diventa alto, mentre i risultati si mantengono simili tra loro, differendo all'ordine dei centesimi. I parametri utilizzati sono gli stessi di inizio capitolo riportati nella didascalia della Figura 6.1.

Verifichiamo ora numericamente, seguendo l'approccio del Capitolo 5, la correttezza del metodo delle linee e la convergenza alla soluzione esatta all'infittirsi delle griglie di discretizzazione. Prezziemo a tal scopo una Call Americana senza dividendi nel modello di Black-Scholes regime-switching, confrontando il prezzo ottenuto con quello calcolato con la formula esatta

(a) Tempo SOR			(b) Tempo SOR		
N	MOL	$Tempo(sec.)$	N	SOR	$Tempo(sec.)$
100	19.9507	3.4969	100	19.8271	5.8846
200	19.8196	3.8646	200	19.8385	34.8518
400	19.8068	4.5797	400	19.8386	227.2980
800	19.8043	6.0080	800	19.8377	957.4258
1600	19.8056	8.8767	1600	19.8381	6708.0449

Tabella 6.1: Metodi $PSOR$ e MOL a confronto.

Primo stato					Secondo stato		
N	M	DI	MOL	Err	DI	MOL	Err
500	10	22.3023	22.0256	0.2767	19.9737	19.7259	0.2478
1000	50	22.3023	22.2491	0.0532	19.9737	19.9243	0.0494
2000	50	22.3023	22.2485	0.0538	19.9737	19.9235	0.0502
4000	100	22.3023	22.2753	0.0270	19.9737	19.9484	0.0253
10000	200	22.3023	22.2888	0.0135	19.9737	19.9610	0.0127
40000	400	22.3023	22.2955	0.0068	19.9737	19.9674	0.0063

Tabella 6.2: Convergenza del metodo delle linee. Parametri $r = 0.1$, $\sigma_1 = 0.5$, $\sigma_2 = 0.2$, $S_0 = 100$, $K = 100$, $Q_{11} = -1$, $Q_{22} = -3$.

presentata nel Capitolo 2. Dai risultati riportati in Tabella 6.2 vediamo che, al crescere dei nodi di discretizzazione spaziale e temporale la soluzione ottenuta con il metodo delle linee converge alla soluzione esatta.

Da ultimo notiamo che, anche in questo capitolo, abbiamo sempre considerato per chiarezza di notazione i parametri legati alla parte di salto, cioè λ , μ e δ costanti. Nulla cambierebbe nella trattazione se ammettessimo che possano assumere valori diversi a seconda dell'economia in cui ci troviamo. Come nel Capitolo 2 sarà sufficiente considerare λ_i , μ_i e δ_i , ciascuno relativo allo stato i -esimo considerato.

Capitolo 7

Risultati numerici

In questa parte del lavoro considereremo un modello di economia a due stati conducendo un'analisi più approfondita dei vari metodi, la cui correttezza è già stata mostrata nei capitoli precedenti, individuandone il migliore. Inizialmente, analizzeremo gli algoritmi presentati per il pricing di opzioni Europee: *FFT*, schema alle differenze finite e Monte Carlo. Nella seconda parte del capitolo, invece, confronteremo gli algoritmi proposti per il pricing di opzioni Americane, cioè l'algoritmo di Longstaff e Schwartz, il *SOR* proiettato, ed il metodo delle linee. In tutta la trattazione, per semplicità di notazione, considereremo i parametri del modello di Lévy relativi alla parte di salto costanti nel tempo, quindi non affetti da switching. Presenteremo, per completezza, un caso in cui tali parametri possano cambiare nel passaggio da uno stato dell'economia all'altro. In conclusione analizzeremo come il passaggio da uno stato all'altro influenzi il prezzo di un'opzione Europea o Americana e come si possa utilizzare il modello di regime-switching, fornendo un esempio di applicazione all'economia reale. Tutti gli esperimenti presentati di seguito sono stati condotti su un PC desktop con processore AMD Athlon(tm2)X4 620 CPU 2.60 GHz con RAM 3.25 GB.

7.1 Opzioni europee

Cominciamo analizzando gli algoritmi per il pricing di opzioni Europee. Consideriamo il caso in cui sia il drift che la volatilità siano soggetti a switching nelle modalità descritte nel Capitolo 2, analizzando il metodo delle differenze finite, la *FFT* ed il metodo Monte Carlo. Consideriamo, inizialmente, il modello standard di Black-Scholes e utilizziamo il seguente set di parametri: $r_1 = 0.1$, $r_2 = 0.05$, $\sigma_1 = 0.5$, $\sigma_2 = 0.3$, $q_{11} = -1$, $q_{22} = -2$, $T = 1$, $S_0 = 100$ e $K = 40, 60, \dots, 160, 180$. Utilizzeremo questo set di parametri per ogni esperimento di questa sezione, salvo dove diversamente specificato. Per le simulazioni Monte Carlo si considerino $N_{sim} = 400000$

simulazioni ed $N_{dates} = 200$ punti di discretizzazione per simulare la traiettoria del sottostante in $[0, T]$. Per il metodo delle differenze finite si è deciso di utilizzare $N = 2500$ nodi sulla griglia del sottostante e $M = 100$ nodi sulla griglia temporale e $\theta = 0.5$. In Tabella 7.1 sono riportati i risultati e gli errori relativi commessi dal metodo Monte Carlo e dal metodo alle differenze finite rispetto alla FFT , per la quale in tutto il capitolo si è scelto di porre, come proposto in [10], $N = 4096$, $\Delta_k = 0.01$, $\rho = 1.0$. Possiamo osservare come tutti i metodi di pricing portino a risultati simili, a prova della loro correttezza.

K	FFT	MC	DF
40	63.3827	63.4736 (-0.0910)	63.5460 (-0.1633)
60	46.6326	46.6501 (-0.0174)	46.6244 (0.0083)
80	32.4935	32.6617 (-0.1682)	32.5217 (-0.0283)
100	21.9375	21.9622 (-0.0248)	21.9234 (0.0141)
120	14.5876	14.5229 (0.0648)	14.5461 (0.0415)
140	9.5365	9.5589 (-0.0223)	9.5939 (-0.0573)
160	6.3395	6.3681 (-0.0286)	6.3411 (-0.0017)
180	4.1938	4.1812 (0.0126)	4.2183 (-0.0245)

Tabella 7.1: Prezzi Call europea Black-Scholes Regime-Switching con regime di partenza $i = 1$. Tra parentesi è indicato l'errore del metodo rispetto al prezzo calcolato con al FFT .

Soffermiamoci per un momento sul tempo computazionale di calcolo del prezzo di una singola opzione. Osserviamo che, eseguendo una sola volta l'algoritmo FFT otteniamo 4096 prezzi. Calcoliamo tutti i 4096 prezzi utilizzando il metodo DF e il metodo Monte Carlo, con i parametri numerici utilizzati nel caso precedente: in Tabella 7.2 sono riportati i tempi necessari per il calcolo di 4096 opzioni con i diversi metodi.

<i>Metodo</i>	$T(s)$
FFT	0.2407
DF	258.5778
MC	6.5679e+5

Tabella 7.2: Tempo necessario per il calcolo di 4096 opzioni con i tre metodi di pricing. $N_{sim} = 400000$, $N_{dates} = 200$, $N = 1000$, $M = 2500$.

Notiamo subito come il metodo FFT sia di gran lunga più veloce sia dell'approccio PDE sia del metodo Monte Carlo. In Tabella 7.2(a) si mostra ancora la bontà dei metodi proposti riportando i prezzi delle Call con regime di partenza $i = 1$ ed $i = 2$. I parametri, numerici ed economici, restano invariati eccetto il generatore della catena di Markov che diventa, da qui in

poi,

$$Q = \begin{bmatrix} -3 & 3 \\ 0.5 & -0.5 \end{bmatrix}.$$

Un confronto sull'efficienza dei metodi di pricing può essere effettuato nel modo seguente: definiamo l'*RMSRD* (root mean-square relative difference) in relazione al prezzo vero calcolato. Dato non esiste una formula chiusa assumeremo che il prezzo giusto sia quello calcolato con il metodo *FFT*. L'*RMSRD* è dato dalla seguente formula

$$RMSRD = \sqrt{\frac{1}{N} \sum_{n=1}^N \left(\frac{C_n^F - C_n}{C_n^F} \right)^2},$$

dove C_n^F è il prezzo calcolato con la trasformata di Fourier mentre C_n è il prezzo calcolato con il metodo. In Figura 7.1 viene mostrato l'andamento dell'errore all'aumentare del tempo di computazione. Si è scelto $K = 60, 80, \dots, 160$.

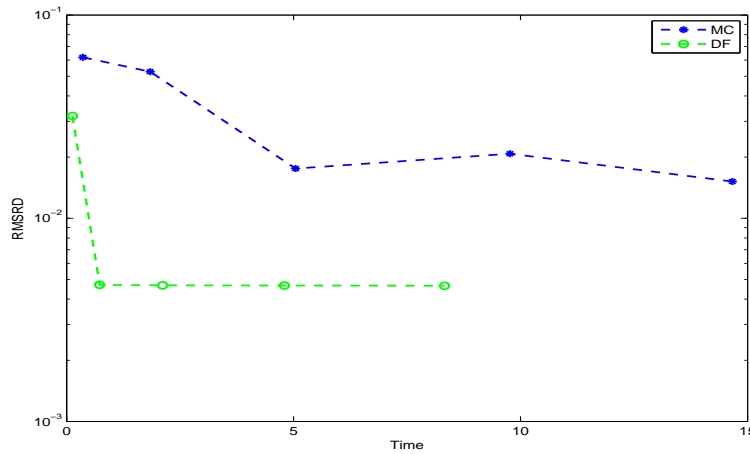


Figura 7.1: Efficienza dei metodi delle differenze finite e MonteCarlo per Call Europea.

Osserviamo che, a parità di tempo, il metodo delle differenze finite è più accurato rispetto al metodo Monte Carlo. Ad esempio, per raggiungere un errore dell'ordine di 10^{-2} il metodo alle differenze finite impiega circa un secondo mentre, ad un tale livello di errore, il metodo Monte Carlo non riesce ad arrivare nemmeno dopo dodici secondi.

Spostiamo ora la nostra attenzione sul caso in cui il sottostante segua una dinamica di tipo Merton jump-diffusion. In Tabella 7.2(b) riportiamo i risultati del metodo alle differenze finite ed i risultati del metodo Monte Carlo con i relativi intervalli di confidenza posto $\alpha = 0.5$. Lasciamo invariati tutti i parametri rispetto all'esperimento precedente e poniamo $\lambda = 1$, media dei

(a) Call Europee Black-Scholes						
Primo stato				Secondo stato		
K	<i>FFT</i>	<i>MC</i>	<i>DF</i>	<i>FFT</i>	<i>MC</i>	<i>DF</i>
60	44.8704	44.8224 (0.0480)	44.8679 (0.0025)	43.6460	43.5629 (0.0831)	43.6448 (0.0012)
80	29.5214	29.5034 (0.0181)	29.5750 (-0.0536)	27.2535	27.3211 (-0.0676)	27.3269 (-0.0733)
100	18.2696	18.2306 (0.0390)	18.2538 (0.0157)	15.3675	15.3885 (-0.0210)	15.3644 (0.0031)
120	10.9074	10.8445 (0.0629)	10.8596 (0.0478)	8.0609	8.1061 (-0.0452)	8.0079 (0.0530)
140	6.3000	6.3279 (-0.0279)	6.3589 (-0.0589)	3.9528	4.0534 (-0.1006)	4.0036 (-0.0508)
160	3.7257	3.7320 (-0.0063)	3.7298 (-0.0041)	1.9780	1.9734 (0.0046)	1.9770 (0.0010)

(b) Call Europee Lévy JD						
Primo stato				Secondo stato		
K	<i>FFT</i>	<i>MC</i>	<i>DF</i>	<i>FFT</i>	<i>MC</i>	<i>DF</i>
60	45.4112	45.1914(0.2197)	45.3937(0.0174)	44.0907	44.0719(0.0189)	44.0760(0.0147)
80	31.0731	31.0004(0.0727)	31.0945(-0.0214)	28.9194	28.9595(-0.0401)	28.9548(-0.0354)
100	20.7340	20.7178(0.0162)	20.6822(0.0517)	18.2181	18.1974(0.0207)	18.1650(0.0531)
120	13.7479	13.6173(0.1306)	13.6442(0.1037)	11.3218	11.2965(0.0253)	11.2104(0.1114)
140	9.0385	9.0242(0.0143)	9.0406(-0.0021)	6.9411	6.9713(-0.0302)	6.9351(0.0059)
160	6.1102	6.1026(0.0076)	6.0579(0.0523)	4.4011	4.3642(0.0369)	4.3437(0.0574)

Tabella 7.3: Prezzi di opzioni Call Europee nel modello di Black-Scholes e nel modello di Merton jump-diffusion in ipotesi di regime-switching.

salti $\mu = 0.2$ e varianza $\delta = 0.1$. Per quanto riguarda l'ordine di convergenza, in questo caso non è garantito che sia $O(\Delta t^2)$ in tempo perchè si è deciso di trattare il termine di accoppiamento e il termine dovuto alla presenza dei salti in maniera esplicita per velocizzare la computazione. Il metodo guadagna in velocità ma perde in ordine di convergenza, come osservato nel Capitolo 5. Osserviamo, in Tabella 7.6 che, all'aumentare del numero di nodi sulle griglie spaziale e temporale per il metodo alle differenze finite e all'aumentare di N_{sim} ed N_{dates} per il metodo Monte Carlo, si arrivi a convergenza. Ciò è vero, ovviamente, sia per il caso di Modello standard di Black-Scholes sia per il modello di Merton jump-diffusion. Possiamo osservare anche che la convergenza del metodo Monte Carlo non è monotona, come previsto dalla teoria.

Analizziamo, come già fatto nel caso del modello di Black-Scholes, i tempi computazionali richiesti ai tre metodi. Dai risultati in Tabella 7.4 osserviamo che il metodo della *FFT* è il più veloce. È da notare come il tempo necessario per il metodo delle differenze finite e del Monte Carlo sia aumentato rispetto al modello standard di Black-Scholes. L'aumento di tempo nel caso si usino differenze finite per la risoluzione della *PDE* è dovuto alla presenza del termine integrale che va calcolato numericamente. Nel metodo Monte Carlo la maggior quantità di tempo si spiega notando che, oltre a dover generare gli istanti di salto per la dinamica del sottostante, è necessario controllare ad ogni istante se il tempo è di salto e quindi generare l'intensità del salto.

<i>Metodo</i>	<i>T (s)</i>
<i>FFT</i>	0.2707
<i>DF</i>	337.0962
<i>MC</i>	7.4840e+05

Tabella 7.4: Tempo necessario per il calcolo di 4096 opzioni con i tre metodi di pricing nel modello di Merton jump diffusion.

Riproponiamo l'analisi di efficienza dei due metodi, assumendo che il prezzo esatto sia quello calcolato dal metodo della *FFT* mentre il vettore degli strike sia pari a quello del caso precedente. In Figura 7.2 è riportato l'andamento dell' *RMSRD* al crescere del tempo di computazione. Osserviamo come la convergenza a zero dell' *RMSRD* sia molto più rapida con il metodo delle differenze finite rispetto che al metodo Monte Carlo, a parità di tempo impiegato.

In conclusione possiamo affermare che il metodo migliore ai fini del pricing di opzioni Europee risulta essere la *FFT* perchè combina accuratezza e velocità. Non abbiamo utilizzato la formula diretta per due motivi: innanzitutto essa vale unicamente per il modello di Black-Scholes in cui l'unico parametro soggetto a switching è la volatilità e già questo rappresenta una

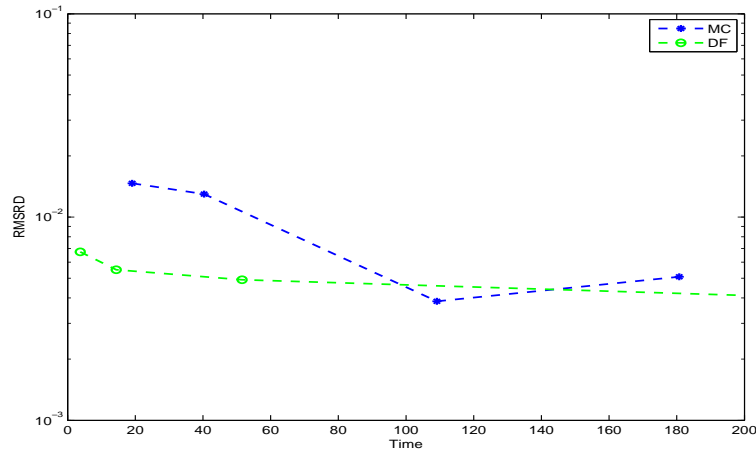


Figura 7.2: Efficienza dei metodi delle differenze finite e Monte Carlo per Call Europea nel modello di Lévy jump-diffusion .

<i>Metodo</i>	<i>Tempo</i>
<i>FFT</i>	0.2566
<i>DI</i>	1.2064e+03

Tabella 7.5: Tempo necessario per il pricing di 4096 opzioni utilizzando la *FFT* e la formula d'integrazione diretta.

limitazione al modello. Gli altri metodi consentono infatti di effettuare il pricing di opzioni in ipotesi sulla dinamica del sottostante ben più complicate. Inoltre, anche se ammettessimo di poterci porre nel modello di Black-Scholes con r costante e σ_t soggetto a switching, dai risultati riportati in Tabella 7.5 osserviamo che il metodo della *FFT* è comunque più veloce dell'integrazione diretta.

Tabella 7.6: Analisi di convergenza dei metodi.

(a) Modello Black-Scholes con regime-switching

M	N	N_{sim}	N_{dates}	FFT	DF	MC	$IC_{\alpha=0.5}$	Err_{FFT-DF}	Err_{FFT-MC}	Err_{DF-MC}
100	100	50000	100	18.2696	18.2349	18.1247	[17.8546 18.3948]	0.0347	0.1449	0.1102
250	100	100000	100	18.2696	18.2640	18.3087	[18.1146 18.5028]	0.0055	-0.0391	-0.0447
500	100	200000	200	18.2696	18.2691	18.1422	[18.0065 18.2779]	0.0004	0.1274	0.1269
1000	100	300000	200	18.2696	18.2687	18.3514	[18.2399 18.4629]	0.0009	-0.0818	-0.0827
2000	200	500000	200	18.2696	18.2694	18.2514	[18.1652 18.3377]	0.0002	0.0181	0.0179

(b) Modello Merton jump diffusion con regime-switching

M	N	N_{sim}	N_{dates}	FFT	DF	MC	$IC_{\alpha=0.5}$	Err_{FFT-DF}	Err_{FFT-MC}	Err_{DF-MC}
100	100	50000	100	20.7340	20.6731	20.5933	[20.2456 20.9410]	0.0608	0.1407	0.0798
250	100	100000	100	20.7340	20.6880	20.6788	[20.4333 20.9243]	0.0460	0.0552	0.0092
500	100	200000	200	20.7340	20.6782	20.6520	[20.4803 20.8238]	0.0558	0.0819	0.0261
1000	100	300000	200	20.7340	20.6834	20.7274	[20.5859 20.8689]	0.0505	0.0066	-0.0440
2000	200	500000	200	20.7340	20.7090	20.7121	[20.6030 20.8213]	0.0250	0.0219	-0.0032

7.2 Opzioni Americane

In questa sezione testeremo i metodi presentati nei capitoli precedenti per il pricing di opzioni Americane. Consideriamo per i nostri esperimenti il set di parametri in Tabella 7.7. Come già fatto per le opzioni Europee ana-

Parametro	Valore
S_0	100
r_1	0.1
r_2	0.04
σ_1	0.4
σ_2	0.2
q_{11}	-1.4
q_{22}	-1.1
T	1
λ	1
μ	0.1
δ	0.2

Tabella 7.7: Parametri modello

lizziamo, innanzitutto, il caso in cui il modello sia quello di Black-Scholes i cui parametri siano soggetti a switching. Consideriamo una Put Americana. Per il primo esperimento consideriamo i seguenti parametri di discretizzazione: per il metodo Montecarlo poniamo $N_{sim} = 100000$ ed $N_{dates} = 100$ mentre per il *SOR* proiettato e per il metodo delle linee scegliamo, così da ottenere un risultato accurato, $N = 1000$ ed $M = 100$, mentre poniamo al tolleranza per il *PSOR* pari a 10^{-5} e $\omega = 1.2$. Come valore dello strike poniamo $K = 60, 80, \dots, 160$. Quello che è interessante notare è il tempo

K	$PSOR$	MC	$IC_{\alpha=0.5}$	MOL
60	0.4923	0.4930(-0.0007)	[0.4773,0.5088]	0.4927(-0.0004)
80	3.1632	3.1582(0.0050)	[3.0314,3.2850]	3.1543(0.0090)
100	10.3414	10.3204(0.0210)	[9.9057,10.7351]	10.3234(0.0180)
120	22.8897	22.7756(0.1141)	[22.1908,23.3604]	22.8755(0.0142)
140	40.1006	40.0821(0.0185)	[39.8038,40.3604]	40.0898(0.0108)
160	60.0000	60.0000(0.0000)	[59.8069,59.9071]	60.0000(0.0000)

Tabella 7.8: Prezzi della Put Americana in ipotesi di regime-switching. Tra parentesi l'errore commesso dal metodo numerico rispetto al risultato del *SOR* proiettato.

necessario ai diversi metodi per calcolare il prezzo di un'opzione.

(a) Tempi computazionali		(b) Iterazioni del <i>PSOR</i> al variare di N . $M = 100$, $TOLL = 10^{-5}$.	
Metodo	$T(s)$	N	Iterazioni
<i>MC</i>	37.22	1000	290
<i>PSOR</i>	651.14	3000	1756
<i>MOL</i>	6.79	5000	4765

Tabella 7.9: Metodi per le opzioni Americane a confronto

Dai risultati riportati in Tabella 7.8 possiamo verificare che tutti e tre i metodi di pricing producono risultati simili tra loro, differendo, generalmente, all'ordine dei centesimi. Osserviamo in Tabella 7.8(a) come il metodo delle linee sia di gran lunga più celere del metodo *PSOR* e anche del Monte Carlo. Ciò che penalizza il *SOR* proiettato è il fatto che si debba risolvere ad ogni passo un sistema lineare di N incognite e il tutto va ripetuto un numero di volte tale da scendere sotto la tolleranza fissata. Quello che abbiamo riscontrato nei nostri esperimenti è che, aumentando il numero di nodi di discretizzazione sulla griglia del sottostante, non solo aumenta la dimensione della matrice ma anche il numero di iterazioni necessarie ad ogni passo per scendere sotto il livello di tolleranza scelto, accettando così la nuova soluzione calcolata, potendo così procedere al passo temporale successivo (Tabella 7.8(b)). Quello che avviene, quindi, è che il tempo necessario per la computazione diventa rapidamente inaccettabile se N cresce eccessivamente.

Facciamo ora un confronto ulteriore tra il metodo *PSOR* ed il metodo delle linee. Consideriamo vari strike price e variamo il numero di punti di discretizzazione dei due metodi osservando i risultati ed il tempo impiegato in Tabella 7.10

Metodo (N, M)	K					Tempo (s)
	80	90	100	110	120	
<i>MOL</i> (100, 50)	3.1368	6.0736	10.3145	15.9462	22.8666	1.90
<i>MOL</i> (500, 100)	3.1542	6.0716	10.3237	15.9480	22.8763	5.15
<i>MOL</i> (1000, 150)	3.1570	6.0765	10.3290	15.9539	22.8803	10.28
<i>PSOR</i> (100, 50)	3.1728	6.0980	10.3534	15.9783	22.9022	2.53
<i>PSOR</i> (500, 100)	3.1635	6.0869	10.3416	15.9659	22.8900	174.5
<i>PSOR</i> (1000, 150)	3.1630	6.0867	10.3415	15.9658	22.8897	1188.1

Tabella 7.10: Confronto tempi del *MOL* e del *PSOR*.

Osserviamo come il metodo delle differenze finite ed il metodo delle linee impieghino sostanzialmente lo stesso tempo solo nel caso in cui il numero

di nodi di discretizzazione sia basso. All'aumentare del numero di nodi il metodo delle linee è molto più conveniente dal punto di vista computazionale rispetto al *PSOR*, risultando ben 100 volte più veloce.

Analogamente a quanto fatto per le opzioni Europee compariamo ora i tre metodi e valutiamone l'efficienza. Assumiamo che la soluzione esatta sia quella calcolata con il *MOL* con $N = 5000$ e $M = 700$, seguendo l'idea di Mayer [11], e sia $K = 60, 80, \dots, 160$.

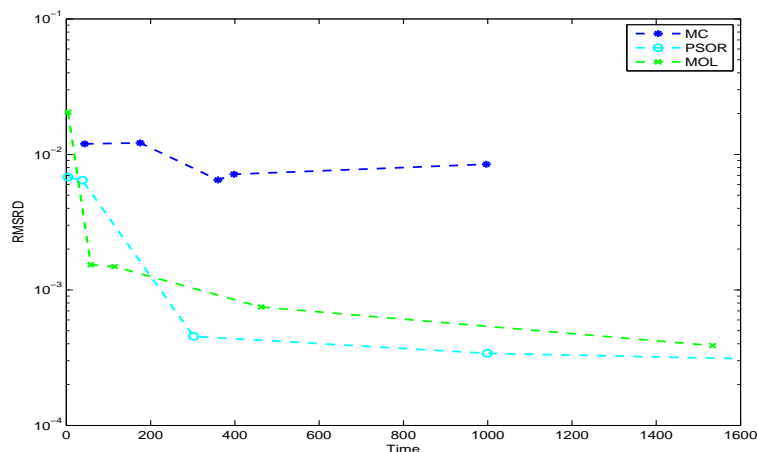


Figura 7.3: Efficienza dei metodi delle differenze finite e MonteCarlo per Put Americana.

Ciò che osserviamo è che, per ottenere un *RMSRD* dell'ordine di 10^{-3} , il metodo delle linee impiega circa cinquanta secondi mentre il metodo alle differenze finite necessita di circa trecento secondi. Siccome ai fini del pricing un errore ai centesimi è accettabile, concludiamo che il metodo delle linee è più efficiente per prezzare derivati. Per quanto riguarda il metodo Monte Carlo osserviamo che esso non riesce a raggiungere un livello di precisione così alto: inoltre l'andamento dell'errore non è monotono, come già accennato.

Da ultimo, analogamente a quanto fatto nel caso delle opzione Europee, proponiamo un'analisi numerica di convergenza dei metodi. In questo caso scegliamo di porre $K = 110$. Osserviamo che all'aumentare delle simulazioni Monte Carlo ed all'aumentare dei nodi di discretizzazione per quanto riguarda il metodo delle differenze finite e il *MOL*, i tre metodi producano risultati sempre più vicini tra loro. Questo risulta evidente dai risultati riportati in Tabella 7.11.

Consideriamo ora il caso in cui il sottostante evolva seguendo un modello di Lévy di tipo jump-diffusion. Dai risultati riportati nella Tabella 7.12 si evince la coerenza dei vari metodi di studiati. Durante le prove numeriche abbiamo riscontrato che, così come accade nel caso delle opzioni Europee, il tempo di computazione è più alto nel caso in cui la dinamica del sottostante sia modellizzata con un precesso di Lévy, a causa del calcolo numerico del

M	N	N_{sim}	$PSOR$	MOL	MC	$IC_{\alpha=0.5}$	$Err_{PSOR-MOL}$	$Err_{PSOR-MC}$	ERR_{MOL-MC}
50	10	10000	15.9365	15.9090	15.9429	[14.1904,17.6954]	0.0274	-0.0064	-0.0339
100	100	40000	15.9943	16.0015	15.8864	[15.0260,16.7467]	-0.0072	0.1079	0.1150
300	100	80000	15.9691	15.9497	15.8287	[15.2204,16.4370]	0.0193	0.1403	0.1210
500	100	100000	15.9642	15.9493	15.8004	[15.2580,16.3429]	0.0149	0.1638	0.1489
1000	200	250000	15.9658	15.9568	15.8644	[15.5148,16.2140]	0.0090	0.1013	0.0924

Tabella 7.11: Analisi di convergenza dei metodi nel caso di Modello Black-Scholes. $N_{dates} = 100$.

K	$PSOR$	MC	$IC_{\alpha=0.5}$	MOL
60	0.9310	0.9224(0.0086)	[0.8898,0.9549]	0.9383(-0.0073)
80	4.6210	4.6182(0.0028)	[4.4046,4.8318]	4.6292(-0.0083)
100	12.7124	12.8266(-0.1142)	[12.2376,13.4156]	12.7417(-0.0293)
120	25.1672	25.1338(0.0335)	[24.2124,26.0551]	25.2450(-0.0777)
140	41.1731	41.1105(0.0626)	[40.2856,41.9354]	41.2639(-0.0908)
160	60.0000	60.0000(-0.0000)	[59.7743,60.0887]	59.9912(0.0088)

Tabella 7.12: Prezzi della Put Americana in ipotesi di regime-switching modello di Lévy. Tra parentesi la differenza del prezzo calcolato dal metodo numerico rispetto al prezzo ottenuto con il SOR proiettato.

termine integrale. Concludiamo dunque che il metodo più efficiente per il pricing sia il metodo delle linee poichè il SOR proiettato, già oneroso di per sè, soprattutto se il numero di nodi sulla griglia dei sottostanti è elevato, risulta ulteriormente rallentato dal calcolo del termine integrale. Anche in questo caso abbiamo verificato la convergenza dei metodi proposti e abbiamo riportato i risultati in Tabella 7.13, dove si è scelto sempre $K = 110$.

Come ulteriore verifica dei metodi di pricing, riportiamo in Tabella 7.14 i prezzi delle Put Americane in caso di regime-switching nel modello di Lévy Merton jump-diffusion e nel modello standard di Black-Scholes.

Riportiamo di seguito, in Tabella 7.15, il tempo impiegato dai metodi alle differenze finite e dal metodo delle linee a parità di numero di nodi di discretizzazione spaziale e temporale.

Osserviamo che in tutti i casi, eccetto che per il SOR proiettato, il tempo necessario per il calcolo del prezzo delle varie opzioni è sempre superiore nel caso in cui il modello per il sottostante sia un jump-diffusion, come già osservato in precedenza. Il fatto che il SOR proiettato impieghi meno tempo nel caso Lévy rispetto al caso Black-Scholes è da imputarsi al fatto che, ad ogni step temporale, sono necessarie meno iterazioni prima di scendere sotto la tolleranza fissata e poter accettare così la nuova soluzione. Osserviamo, tuttavia, che la differenza di tempo nei due modelli si riduce sempre di più all'aumentare del numero di nodi perché il calcolo del termine integrale diventa via via sempre più oneroso.

In conclusione possiamo affermare che il metodo migliore ai fini del pricing di opzioni Americane è il metodo delle linee che combina accuratezza e velocità, come si evince dagli esperimenti numerici effettuati.

N	M	N_{sim}	$PSOR$	MOL	MC	$IC_{\alpha=0.5}$	$Err_{PDE-MOL}$	Err_{PDE-MC}	ERR_{MOL-MC}
50	10	10000	15.9365	15.9090	15.9429	[14.1904,17.6954]	0.0274	-0.0064	-0.0339
100	100	40000	15.9943	16.0015	15.8864	[15.0260,16.7467]	-0.0072	0.1079	0.1151
300	100	80000	15.9691	15.9497	15.8287	[15.2204,16.4370]	0.0193	0.1403	0.1210
500	100	100000	15.9642	15.9493	15.8004	[15.2580,16.3429]	0.0149	0.1638	0.1489
1000	200	250000	15.9658	15.9568	15.8644	[15.5148,16.2140]	0.0090	0.1013	0.0924

Tabella 7.13: Analisi di convergenza dei metodi nel caso di modello di Lévy. $N_{dates} = 100$.

(a) Put Americane Lévy jump-diffusion						
Primo stato				Secondo stato		
K	<i>MOL</i>	<i>MC</i>	<i>PSOR</i>	<i>MOL</i>	<i>MC</i>	<i>PSOR</i>
60	0.9383	0.9224(-0.0159)	0.9310(-0.0073)	0.5034	0.4970(-0.0064)	0.4975(-0.0059)
80	4.6292	4.6182(-0.0111)	4.6210(-0.0083)	3.4111	3.4230(0.0119)	3.4068(-0.0043)
100	12.7417	12.8266(0.0849)	12.7124(-0.0293)	11.2825	11.2785(-0.0040)	11.2544(-0.0281)
120	25.2450	25.1338(-0.1112)	25.1673(-0.0777)	24.2452	24.1735(-0.0717)	24.1457(-0.0995)
140	41.2639	41.1105(-0.1534)	41.173086(-0.0908)	40.8836	40.7824(-0.1012)	40.7560(-0.1277)
160	59.9912	60.0000(0.0088)	60.0000(0.0088)	60.0000	60.0008(0.0008)	60.0000(0.0000)

(b) Put Americane Black-Scholes						
Primo stato				Secondo stato		
K	<i>MOL</i>	<i>MC</i>	<i>PSOR</i>	<i>MOL</i>	<i>MC</i>	<i>PSOR</i>
60	0.4928	0.4814(-0.0114)	0.4924(-0.0004)	0.1911	0.1923(0.0012)	0.1897(-0.0014)
80	3.1542	3.1388(-0.0154)	3.1632(0.0090)	1.8424	1.8434(0.0010)	1.8488(0.0063)
100	10.3231	10.2236(-0.0995)	10.3392(0.0162)	8.1708	8.1777(0.0069)	8.1920(0.0212)
120	22.8750	22.7275(-0.1475)	22.8897(0.0147)	21.2305	21.1054(-0.1251)	21.2435(0.0131)
140	40.0991	40.0000(-0.0991)	40.1007(0.0016)	40.0000	40.0000(0.0000)	40.0000(0.0000)
160	60.0000	60.0000(0.0000)	60.0000(0.0000)	60.0000	60.0000(0.0000)	60.0000(0.0000)

Tabella 7.14: Prezzi Put Americane nei due modelli ottenuti con *MOL*, *PSOR* e Monte Carlo. Tra parentesi l'errore del metodo calcolato rispetto al prezzo calcolato con il *MOL*.

Tempo (s)							
		Black-Scholes			Lévy jump-diffusion		
N	M	DF_{Eur}	DF_{Am}	MOL_{Am}	DF_{Eur}	DF_{Am}	MOL_{Am}
10	10	0.0035	0.0381	0.3237	0.0052	0.0446	0.0774
100	100	0.0094	4.2216	3.5317	0.5157	4.0450	4.0647
200	100	0.0109	12.4422	7.0516	1.5804	14.8357	8.1465
300	100	0.0124	49.9933	10.5859	3.2571	37.0103	12.6776
500	100	0.0157	216.7250	17.7665	8.4084	185.3603	20.8228

Tabella 7.15: Tempi impiegati dai vari metodi di pricing per la risoluzione delle PDE .

7.3 Un caso più generale

In tutti gli esperimenti numerici fin'ora condotti, abbiamo sempre considerato il caso in cui a cambiare aleatoriamente fossero r e σ , mentre i parametri del processo jump-diffusion venivano assunti, per semplicità espositiva e di notazione, costanti. Nulla ci vieta di concedere a questi parametri, λ , μ e δ di cambiare durante il passaggio da uno stato dell'economia all'altro.

Consideriamo allora il caso in cui tutti i parametri possano variare durante il passaggio di stato, e verifichiamo la correttezza degli algoritmi. Concentriamoci sul caso Europeo. Il procedimento per il caso Americano è analogo e nulla aggiunge, di fatto, alla trattazione, quindi lo omettiamo. Consideriamo un'opzione Call con $S_t = 100$, $K = 90$, $T = 1$, $q_{11} = -2$, $q_{22} = -2$ ed il seguente set di parametri per il sottostante mostrato in Tabella 7.16.

<i>Parametro</i>	<i>Primo stato</i>	<i>Secondo stato</i>
r	0.14	0.4
σ	0.3	0.2
λ	1	2
μ	0.1	-0.1
δ	0.1	0.2

Tabella 7.16: Parametri.

Nei seguenti esperimenti decidiamo di porre come parametri per il Monte Carlo $N_{sim} = 250000$ ed $N_{dates} = 100$ mentre per lo schema alle differenze finite si è scelto $N = 1000$ ed $M = 100$. Assumiamo ancora una volta che il prezzo esatto sia quello calcolato con la trasformata di Fourier e confrontiamolo con gli errori commessi dagli altri due metodi. Il codice necessario per risolvere l'equazione alle derivate parziali integrale è facilmente derivabile da quello presentato per risolvere il problema in cui i parametri λ , μ e δ sono costanti. E' sufficiente sostituire considerare due differenti misure di Lévy, $\nu_1(x)$ e $\nu_2(x)$, una per ognuno dei due differenti stati dell'economia. Per quanto riguarda il Montecarlo, invece, l'idea per risolvere il problema è stata presentata alla fine del Capitolo 4. Si era proposto, come primo approccio, di ricondursi al caso con λ costante stimando tramite simulazioni un certo $\bar{\lambda}$ calcolato come media pesata sul tempo di soggiorno di λ_1 e λ_2 , mentre un procedimento alternativo consisteva nello sfruttare l'indipendenza del processo di salto dal processo che regola il passaggio di stato. Per il primo metodo nel nostro esperimento abbiamo ricavato $\bar{\lambda}$ facendo 20000 simulazioni.

I risultati dell'esperimento sono riportati in Tabella 7.17. Osserviamo come il prezzo ottenuto utilizzando le differenze finite per la *PIDE* sia in accordo con quello calcolato tramite *FFT* mentre per quanto riguarda il metodo Monte Carlo, sebbene entrambi gli approcci portino a risultati simili e siano

quindi corretti, l'idea di stimare $\bar{\lambda}$ come media pesata di λ_1 e λ_2 porti, in generale, ad errori più elevati rispetto a sfruttare l'indipendenza del processo del passaggio da uno stato dell'economia all'altro e del processo di salto nella dinamica dei sottostanti. Questo è un risultato atteso poichè, riconducendoci al caso con λ costante, introduciamo un errore di approssimazione dovuto al fatto che consideriamo λ , che è una quantità stocastica, come una quantità deterministica ed un errore numerico dovuto al calcolo di $\bar{\lambda}$. Quindi, in accordo a quanto già osservato in precedenza, per il metodo Monte Carlo è un'idea migliore seguire quest'ultimo approccio.

(a) Prezzo delle Call Europee con procedimento di stima per $\bar{\lambda}$						
Primo stato				Secondo stato		
K	FFT	MC	DF	FFT	MC	DF
60	45.9688	45.8637(0.1051)	45.9377(0.0311)	44.5289	44.4014(0.1275)	44.4874(0.0416)
80	30.1144	30.0249(0.0895)	30.1450(-0.0306)	28.6731	28.4149(0.2582)	28.6921(-0.0190)
100	18.0106	17.9962(0.0144)	17.9465(0.0641)	16.7221	16.2618(0.4603)	16.6458(0.0763)
120	10.0237	10.0273(-0.0037)	9.8984(0.1253)	8.9598	8.4950(0.4648)	8.8286(0.1312)
140	5.2054	5.4729(-0.2675)	5.2189(-0.0136)	4.4300	4.2460(0.1840)	4.4361(-0.0061)
160	2.7450	2.9186(-0.1736)	2.7058(0.0392)	2.2320	2.0990(0.1329)	2.1920(0.0400)
(b) Prezzo delle Call Europee con generazione separata dei tempi di salto						
Primo stato				Secondo stato		
K	FFT	MC	DF	FFT	MC	DF
60	45.9688	45.9166(0.0522)	45.9377(0.0311)	44.5289	44.6381(-0.1092)	44.4874(0.0416)
80	30.1144	30.0765(0.0380)	30.1450(-0.0306)	28.6731	28.8578(-0.1847)	28.6921(-0.0190)
100	18.0106	17.9337(0.0769)	17.9465(0.0641)	16.7221	16.7588(-0.0367)	16.6458(0.0763)
120	10.0237	9.9080(0.1157)	9.8984(0.1253)	8.9598	8.9176(0.0422)	8.8286(0.1312)
140	5.2054	5.2830(-0.0776)	5.2189(-0.0136)	4.42300	4.4754(-0.0455)	4.4361(-0.0061)
160	2.7450	2.7206(0.0243)	2.7058(0.0392)	2.2320	2.2392(-0.0072)	2.1920(0.0400)

Tabella 7.17: Confronto dei due approcci per il metodo Monte Carlo nel caso di λ_t soggetto a switching.

7.4 L'influenza del regime-switching

In questa sezione vedremo come poter passare da uno stato dell'economia all'altro nel periodo di tempo $[0, T]$ influisca notevolmente sul prezzo dell'opzione. Terremo ora fissi tutti i parametri variando, di volta in volta, solo il generatore della catena di Markov Q . Consideriamo prima il caso Europeo per il quale utilizzeremo come metodo di pricing l'algoritmo della *FFT* poichè si è rivelato il più efficiente. Come parametri scegliamo quelli riportati nella Tabella 7.18. Osserviamo allora i risultati riportati in Tabella 7.19.

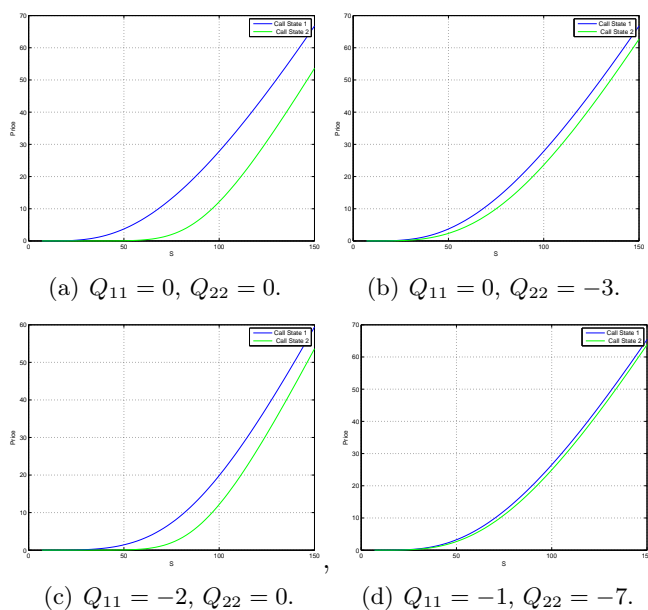
<i>Parametro</i>	<i>Primo stato</i>	<i>Secondo stato</i>
r	0.08	0.02
σ	0.6	0.2
λ	2	1
μ	0.1	-0.1
δ	0.1	0.2

Tabella 7.18: Parametri.

Q_{11}	Q_{22}	Primo stato	Secondo stato
0	0	27.9857	12.1793
0	-3	27.9857	23.7290
-2	0	19.8913	12.1793
-1	-7	26.6411	25.0444

Tabella 7.19: Prezzo della Call Europea al variare di Q .

Ricordiamo preliminarmente che il generatore della catena di Markov Q è strettamente collegato alle probabilità di transizione da uno stato all'altro come descritto nel Capitolo 2: più è alto in valore assoluto l'elemento sulla diagonale e maggiore è la probabilità di abbandonare quello stato e finire nell'altro. Se $Q_{ii} = 0$ allora lo stato è detto assorbente, cioè non è più possibile abbandonarlo. Nel caso in cui il generatore sia nullo, i prezzi delle opzioni nei due stati dell'economia non risentono l'uno dell'altro e, di fatto, coincidono con i prezzi delle Call Europee nel modello di Merton ciascuna col il proprio set di parametri: non è contemplata, difatti, la possibilità che l'economia possa cambiare e quindi i prezzi dell'opzione non sono influenzati dall'esistenza dell'altro stato del mondo, che risulta, sostanzialmente, “invisibile”. Osserviamo, dai risultati riportati in Tabella 7.19, che un investitore si aspetta che il sottostante nel primo stato dell'economia incrementerà il suo valore maggiormente che nel secondo: è disposto, difatti, a pagare un prezzo più alto per acquistare la Call.

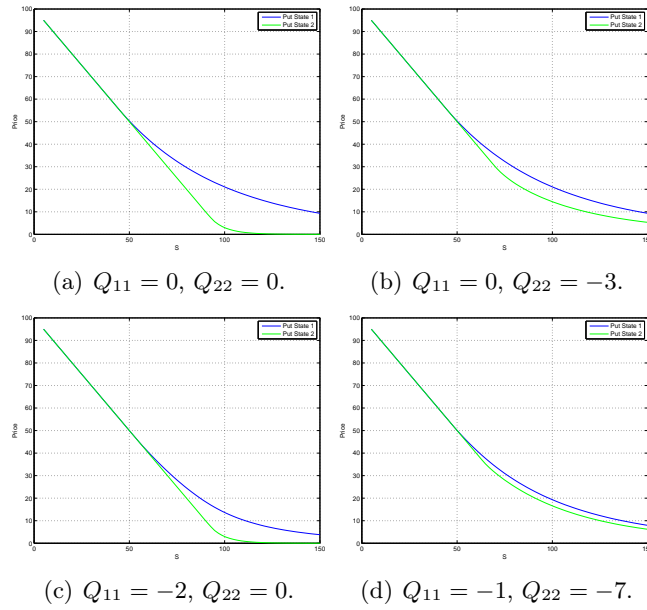
Figura 7.4: Prezzo opzione Call Europea al variare di Q .

Consideriamo ora il caso in cui il primo stato, caratterizzato da prezzi più alti, sia uno stato assorbente ed il secondo abbia un valore del generatore Q_{22} non nullo. Passare dal primo al secondo stato è impossibile, di conseguenza, i prezzi delle opzioni in tale stato rimarranno invariati rispetto al caso precedentemente considerato. Viceversa, essendoci una probabilità non nulla di passare dal secondo al primo stato, migliore, per assunzione, di quello di partenza, i prezzi delle opzioni Call nel secondo stato saranno più elevati di quelli calcolati con $Q_{22} = 0$, come mostra la Figura 7.4(b). Al contrario, come mostra la Figura 7.4(c), se lo stato assorbente fosse il secondo il prezzo delle Call nel primo stato scenderebbe rispetto al caso $Q_{11} = 0$, perché rifletterebbe la probabilità, diversa da zero, di passare ad un peggiore stato dell'economia.

Se invece ci fosse la possibilità di passare da uno stato all'altro e viceversa i prezzi diverrebbero via via più simili al decrescere della differenza tra Q_{11} e Q_{22} , avvicinandosi al prezzo dello stato in cui il tempo di soggiorno è più basso, cioè in cui $|Q_{ii}|$ è più alto.

A conclusioni analoghe si arriva analizzando il caso delle opzioni Americane: in Tabella 7.20 sono riportati i prezzi di un'opzione Put Americana che è stata prezzata considerando gli stessi parametri del caso Europeo utilizzando il metodo di pricing più efficiente, cioè il metodo delle linee. Quanto detto, graficamente risulta ancora più chiaro: in Figura 7.4 è riportato il prezzo della Call Europea al variare di S_0 per i quattro casi analizzati sopra mentre in Figura 7.5 è stata fatta la stessa cosa per la Put Americana.

Q_{11}	Q_{22}	Primo stato	Secondo stato
0	0	21.0652	2.9874
0	-3	21.0652	14.5004
-2	0	13.6621	2.9874
-1	-7	19.3181	16.5926

Tabella 7.20: Prezzo della Put Americana al variare di Q .Figura 7.5: Prezzo opzione Put Americana al variare di Q .

Vediamo come è possibile utilizzare tali considerazioni per un economia reale. Supponiamo di essere in una fase di mercato con un certo tasso di interesse privo di rischio $r_1 = 0.05$ e supponiamo di prevedere un crollo nel tasso di interesse ad un valore di $r_2 = 0.01$ entro un anno. Come possiamo prezzare una Call Europea in tale situazione? Potremmo essere ottimisti e decidere di prezzare la Call utilizzando il tasso risk-free attuale oppure essere pessimisti e considerare il crollo dei tassi inevitabile decidendo di usare r_2 : nel primo caso otterremo un sovrapprezzo della Call rispetto al valore che dovrebbe avere nella realtà mentre nel secondo caso la sottoprezzeremo. Utilizzando, invece, il modello di regime-switching possiamo ottenere un prezzo intermedio che si pone tra le due situazioni estreme e che considera l'eventualità che in un certo istante di tempo, da ora ad un anno, il tasso risk-free diminuisca. Come mostrato in Figura 7.6, otteniamo un prezzo che si pone a metà tra quelli ricavati con i due diversi approcci: più basso di quello ottenuto utilizzando r_1 , perchè c'è la possibilità di una fase di mercato

più sfavorevole dell'attuale, e più alto rispetto al caso in cui si considerasse r_2 , poichè consideriamo il crollo dei tassi di interesse possibile ma non attuale. Come risulta ovvio, il modo in cui il grafico del prezzo dell'opzione in

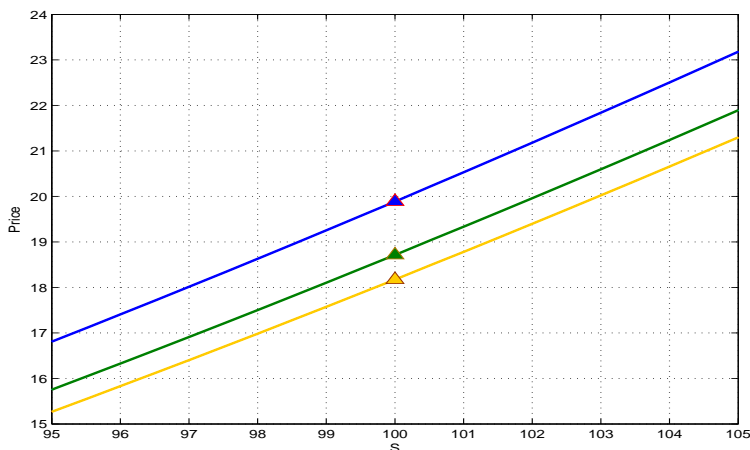


Figura 7.6: Prezzi delle Call per vari tassi di interesse risk-free. In blu e in giallo quelli ottenuti considerando rispettivamente r_1 ed r_2 senza possibilità di switch, mentre in verde quello ottenuto con il modello di regime-switching.

ipotesi di regime-switching si posizionerà tra i due grafici che rappresentano il prezzo dell'opzione nei due stati del mondo in assenza di switching, dipenderà dal generatore della catena di Markov Q , come si osserva in Figura 7.7. Come abbiamo già avuto modo di osservare nel Capitolo 2, il valore atteso del primo tempo di uscita dallo stato 1, indicato con τ_1 , è relazionato a Q_{11} tramite

$$\mathbb{E}[\tau_1] = \frac{1}{|Q_{11}|}.$$

Quindi, maggiore è $|Q_{11}|$, prima il processo di Markov abbandonerà lo stato 1, finendo nell'altro. Economicamente possiamo dire che: prima ci si aspetta possa verificarsi un crollo dei tassi di interessi, più il prezzo dell'opzione avente regime di partenza caratterizzato da $r_1 = 0.05$ sarà vicino al prezzo della Call Europea valutata con $r = 0.01$.

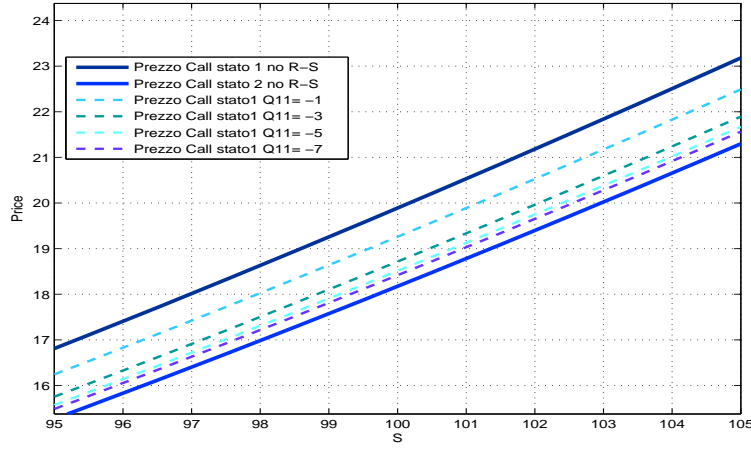


Figura 7.7: Prezzi delle Call per vari tassi di interesse risk-free. In linea continua i prezzi ottenuti considerando rispettivamente r_1 ed r_2 senza possibilità di switch, mentre in linea tratteggiata quelli ottenuti con il modello di regime-switching per diversi valori di Q_{11} , mentre $Q_{22} = 0$.

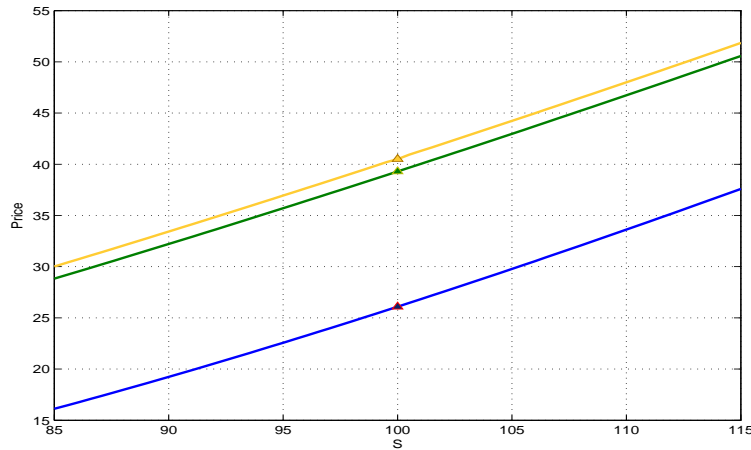


Figura 7.8: Prezzi delle Call per vari valori della volatilità σ . In blu e in giallo quelli ottenuti considerando rispettivamente σ_1 ed σ_2 senza possibilità di switch, mentre in verde quello ottenuto con il modello di regime-switching.

Analogamente, un simile approccio può essere utilizzato se si prevede un qualsiasi cambio di situazione economica. Supponiamo che da ora a tre anni sia previsto un aumento della volatilità dei prezzi da $\sigma_1 = 0.2$ a $\sigma_2 = 0.5$ dovuta, ad esempio, ad incertezza nei mercati finanziari. Già sappiamo che il prezzo di una Call Europea è funzione crescente della volatilità. Quale volatilità è opportuno considerare ai fini del pricing? Ancora una volta il modello di regime-switching può catturare la possibilità di cambio di volatilità in un determinato istante aleatorio nell'arco di tre anni. Il prezzo della

Call ottenuto in tal caso sarà un prezzo intermedio tra quelli ottenuti tra σ_1 e σ_2 come mostrato in Figura 7.8 e dipenderà dalla probabilità che si verifichi un cambio di tendenza dell'economia. In Figura 7.9 riportiamo diversi grafici del prezzo dell'opzione al variare di Q .

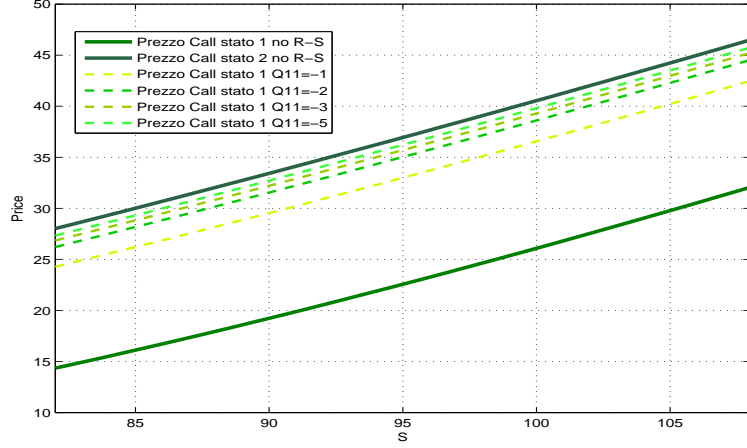


Figura 7.9: Prezzi delle Call per vari valori della volatilità σ . In linea continua i prezzi ottenuti considerando rispettivamente σ_1 ed σ_2 senza possibilità di switch, mentre in linea tratteggiata quelli ottenuti con il modello di regime-switching per diversi valori di Q_{11} , mentre $Q_{22} = 0$.

Ovviamente risulta necessaria la calibrazione del generatore Q della catena di Markov. Si potrebbe pensare, in un primo momento, di utilizzare uno dei tanti algoritmi noti per stimare le probabilità di transizione da uno stato all'altro, tra i quali ricordiamo l'algoritmo di Baum-Welch [3]. Questo metodo si basa su tre assunzioni:

- I dati a disposizione provengono da un processo stocastico che può essere approssimato con una catena di Markov nascosta.
- Il numero di stati è finito e le probabilità di transizione sono sconosciute.
- Possiamo assegnare dei valori iniziali, eventualmente casuali, alle probabilità.

L'algoritmo procede quindi con una serie di passi del tipo *expectation-maximization* che lo portano a stimare le probabilità di transizione, arrestandosi quando l'aggiornamento delle probabilità non è più significativo. Ottenuta la matrice delle probabilità di transizione P , possiamo calcolare il generatore della catena di Markov Q ricordando che

$$P = \exp(Qt).$$

Tuttavia esistono metodi che permettono di stimare direttamente Q . Come osservato da Hahn e Sass [20], passando dal tempo continuo al tempo discreto discreto, potremmo utilizzare l'algoritmo EM , come proposto da Elliot in [15]. In presenza di basse volatilità e piccoli tassi di transizione l'algoritmo EM produce risultati molto accurati. Se i tassi di transizione sono alti, tuttavia, la legge del modello discretizzato differisce molto dalla legge a tempo continuo e l'algoritmo EM produce pessimi risultati. In tempo continuo, difatti, non è noto nessun tipo di filtro che permetta di stimare lo stato della catena di Markov e non è quindi possibile calcolare lo stimatore di massima verosimiglianza tramite l'algoritmo EM : si veda per dettagli [15]. Un'alternativa è quella di usare un metodo basato sui momenti, come proposto da Elliot in [16], ma in tal caso sarebbe necessaria una grande quantità di dati. Adottando invece un approccio bayesiano, è possibile utilizzare due metodi basati sulle catene di Markov Monte Carlo ($MCMC$): il primo [2] di essi consta nel discretizzare il tempo e quindi approssimare il generatore Q con una matrice di transizione mentre il secondo [26] lavora in tempo continuo. Entambi i metodi utilizzano la tecnica del campionamento moltiplicativo a blocchi di Metropolis-Hastings. Tuttavia, come è stato mostrato da Hahn in [19], tale algoritmo, nel caso in cui ci siano alti tassi di transizione e poche osservazioni, risulta lento ed instabile.

L'idea proposta da Hahn e Sass nel 2009 [20] è di combinare i due approcci relativi alle $MCMC$ di modo da trarre i vantaggi relativi sia al lavorare nel discreto, dove è possibile l'utilizzo di filtri, che nel continuo, dove non vengono introdotti errori di discretizzazione. Tale idea si mostra essere vincente, come dimostrano i risultati ottenuti in [20].

Conclusioni

In questo lavoro abbiamo ipotizzato che i parametri che regolano l'andamento dei prezzi di mercato dipendano dallo stato dell'economia in cui ci si trova e che si possa passare, in istanti di tempo aleatori, da uno stato dell'economia all'altro: fenomeno detto di regime-switching. Dopo aver introdotto il modello di regime-switching in ambito log-normale, fornendo, quindi, un'estensione quindi del modello di Black-Scholes, abbiamo ricavato l'equazione alle derivate parziali che ci permette di valutare il prezzo di un'opzione estendendola poi ai più generali modelli di Lévy.

Sotto alcune ipotesi semplificative siamo riusciti, inoltre, a derivare una formula di pricing chiusa per le opzioni Europee che ci ha consentito di verificare la correttezza di tutti i metodi numerici studiati. Si è visto, infatti, come tutti i principali strumenti di finanza quantitativa utilizzati per il pricing di derivati quali fast Fourier transform, differenze finite per la risoluzione dell'equazione alle derivate parziali, metodo Monte Carlo, metodo delle linee, siano applicabili, opportunamente modificati, anche in ipotesi di regime-switching. Per l'algoritmo *FFT* siamo stata calcolata la funzione caratteristica che ci consente di calcolare il prezzo di un'opzione Europea: si è notato che l'unica aggiunta rispetto ai processi di Lévy con parametri costanti è il calcolo della funzione caratteristica del vettore che tiene conto del tempo di soggiorno del processo in ognuno degli stati dell'economia nel periodo da 0 a T . Anche il metodo Monte Carlo e l'algoritmo di Longstaff e Schwartz sono stati adattati a questo modello sia nel caso in cui lo switch interessi solo il tasso risk-free r e la volatilità σ , sia nel caso in cui coinvolga anche i parametri della parte di salto, ovvero λ , μ e δ . In quest'ultimo caso ci siamo resi conto che approssimare λ con una certa $\bar{\lambda}$ medio costante non sia un approccio ottimale per il pricing: sfruttando l'indipendenza del processo di salto dal processo che regola il passaggio tra stati dell'economia, in aggiunta alla proprietà di assenza di memoria del processo di salto, abbiamo ottenuto risultati molto più accurati. Relativamente alla risoluzione delle equazioni alle derivate parziali del nostro modello abbiamo utilizzato i metodi numerici standard alle differenze finite osservando che, trattando esplicitamente il termine di

accoppiamento, si ottiene un prezzo di poco differente rispetto al caso in cui lo si tratti implicitamente o con θ -metodo, guadagnando notevolmente in velocità di calcolo. Nel caso delle opzioni Americane si è rivelato necessario trovare un metodo alternativo al *SOR* proiettato a causa dell'eccessivo tempo computazionale richiesto, ragione per cui abbiamo introdotto il metodo delle linee che ci ha permesso di effettuare il pricing in tempo notevolmente più breve. Dopo aver verificato la coerenza e l'efficienza di tutti i metodi proposti siamo arrivati alla conclusione che il metodo migliore, considerando un trade-off tra precisione e tempo computazionale richiesto, per il pricing di opzioni Europee è la *FFT*, mentre per il caso Americano è consigliato l'utilizzo del metodo delle linee.

Infine, abbiamo cercato di spiegare e mostrare come una realtà empiricamente osservabile quale il problema della determinazione del prezzo dei derivati nell'ipotesi in cui ci sia un cambio improvviso del tasso risk-free o della volatilità dei prezzi di mercato possa essere risolta attraverso l'utilizzo dei modelli regime-switching.

Appendice A

Funzione Caratteristica

In questa Appendice riportiamo la derivazione della funzione caratteristica nel modello di Black-Scholes nell'ipotesi di regime switching. Per i dettagli si rimanda a [10]. Consideriamo un modello a $\mathcal{M} = 1, \dots, m$ stati. Sia \mathcal{F}_T la σ -algebra generata dalla catena di Markov. Sia inoltre $f_{\mathcal{F}_t}$ la densità condizionata della funzione $X(T)$ data \mathcal{F}_T . Definiamo

$$L_T = \int_0^T \alpha_t dt, \quad V_T = \int_0^T \sigma_t^2 dt, \quad R_T = \int_0^T r_t dt.$$

Sia $C(k)$ il prezzo della Call: definito $c(k)$ come

$$c(k) = e^{\rho k} \frac{C(k)}{S_0} \quad -\infty < k < \infty,$$

la trasformata di Fourier di $c(k)$ è calcolata come segue:

$$\begin{aligned} \psi(u) &= \int_{-\infty}^{\infty} e^{iuk} c(k) dk \\ &= \int_{-\infty}^{\infty} e^{iuk} e^{\rho k} \mathbb{E} \left[\left(e^{-R_T} (e^{X(T)-e^k})^+ \mid \mathcal{F}_T \right) dk \right] \\ &= \mathbb{E} \left\{ \int_{-\infty}^{\infty} e^{iuk} e^{\rho k} e^{-R_T} \mathbb{E} \left[\left(e^{X(T)} - e^k \right)^+ \mid \mathcal{F}_T \right] dk \right\} \\ &= \mathbb{E} \left\{ \int_{-\infty}^{\infty} e^{iuk} e^{\rho k} e^{-R_T} \int_k^{\infty} (e^x - e^k) f_{\mathcal{F}_T}(x) dx dk \right\} \end{aligned}$$

$$\begin{aligned}
&= \mathbb{E} \left\{ \int_{-\infty}^{\infty} e^{-R_T} f_{\mathcal{F}_T}(x) \int_{-\infty}^x \left(e^x e^{(\rho+iu)k} - e^{(1+\rho+iu)k} \right) dk dx \right\} \\
&= \mathbb{E} \left\{ \int_{-\infty}^{\infty} e^{-R_T} f_{\mathcal{F}_T}(x) \left(\frac{e^{(1+\rho+iu)x}}{\rho+iu} - \frac{e^{(1+\rho+iu)x}}{1+\rho+iu} \right) dx \right\} \\
&= \mathbb{E} \left\{ e^{-R_T} \left(\frac{\phi_{\mathcal{F}_T}(u-i(1+\rho))}{\rho+iu} - \frac{\phi_{\mathcal{F}_T}(u-i(1+\rho))}{1+\rho+iu} \right) \right\} \\
&= \frac{\mathbb{E} [e^{-R_T} \phi_{\mathcal{F}_T}(u-i(1+\rho))]}{\rho^2 + \rho - u^2 + i(1+2\rho)u},
\end{aligned}$$

dove

$$\phi_{\mathcal{F}_t}(u) = \mathbb{E} [e^{iuX(T)} | \mathcal{F}_T] = \int_{-\infty}^{\infty} e^{iux} f_{\mathcal{F}_T}(x) dx$$

è la funzione caratteristica condizionata di $X(T)$ dato \mathcal{F}_T .

Osserviamo che dato \mathcal{F}_T , $X(T)$ ha distribuzione Gaussiana con media $(L_T - (1/2)V_T)$ e varianza V_T . Segue che

$$\phi_{\mathcal{F}_T}(u) = \exp \left(iu \left(L_T - \frac{1}{2}V_T \right) - \frac{1}{2}u^2 V_T \right).$$

Allora otteniamo che

$$\psi(u) = \frac{\mathbb{E} [\exp ((1+\rho) (L_T + \frac{1}{2}\rho V_T) - R_T - \frac{1}{2}u^2 V_T + iu (L_T + (\frac{1}{2} + \rho) V_T))]}{\rho^2 + \rho - u^2 + i(1+2\rho)u}. \quad (\text{A.1})$$

Per derivare una formula esplicita per $\psi(u)$, è necessario calcolare il valore atteso rispetto alle variabili aleatorie L_T , V_T e R_T . Sia

$$T_j = \int_0^T I_{[\gamma(t)=j]} dt, \quad j \in \mathcal{M},$$

il tempo di soggiorno della catena di Markov γ nello stato j durante l'intervallo $[0, T]$. Allora $\sum_{j=1}^m T_j = T$. Segue che

$$\begin{aligned}
L_T &= \sum_{j=1}^{m-1} (\alpha(j) - \alpha(m) T_j) + \alpha(m) T, \\
V_T &= \sum_{j=1}^{m-1} (\sigma^2(j) - \sigma^2(m)) T_j + \sigma^2(m) T, \\
R_T &= \sum_{j=1}^{m-1} (r(j) - r(m)) T_j + r(m) T.
\end{aligned} \quad (\text{A.2})$$

Utilizzando (A.2) in (A.1), otteniamo

$$\psi(u) = \frac{1}{\rho^2 + \rho - u^2 + i(1 + 2\rho)u} \exp(B(u)T) \mathbb{E} \left[\exp \left(i \sum_{j=1}^{m-1} A(u, j) T_j \right) \right],$$

dove per $j = 1, \dots, m-1$,

$$\begin{aligned} A(u, j) = & \left[(\alpha(j) - \alpha(m)) + \left(\frac{1}{2} + \rho \right) (\sigma^2(j) - \sigma^2(m)) \right] u + \frac{1}{2} u^2 (\sigma^2(j) - \sigma^2(m)) i \\ & + \left[(r(j) - r(m)) - (1 + \rho) (\alpha(j) - \alpha(m)) - \frac{1}{2} \rho (1 + \rho) (\sigma^2(j) - \sigma^2(m)) \right] i, \end{aligned}$$

$$B(u) = iu \left[\alpha(m) + \left(\frac{1}{2} + \rho \right) \sigma^2(m) \right] - \frac{1}{2} u^2 \sigma^2(m) + (1 + \rho) \alpha(m) - r(m) + \frac{1}{2} \rho (1 + \rho) \sigma^2(m).$$

Perciò, il calcolo di $\psi(u)$ si riduce a calcolare la funzione caratteristica del vettore aleatorio $(T_1, \dots, T_{m-1})'$. I dettagli possono essere trovati in [27].

Appendice B

Codici

In questa Appendice riportiamo tutti i codici che abbiamo utilizzato per produrre i risultati numerici del Capitolo 7. Per chiarezza abbiamo deciso di separare i codici veri e propri per il pricing dalle funzioni ausiliarie utilizzate da più codici contemporaneamente le quali sono riportate per ultime. Prima sono riportati i metodi per le opzioni europee ed in seguito i metodi per le opzioni americane

B.1 Opzioni Europee

```
1 function [Call1, Call2]=Call_Price_RS_Explicit_Formula(S0,K,r,sigma,T,Q)
2 %
3 % Prezza una Call Europea nel modello di Black-Scholes in ipotesi di regime
4 % switching sulla varianza.
5 %
6 % S0 valore iniziale del sottostante
7 % K strike price
8 % r tasso risk-free
9 % sigma volatilita'
10 % T maturity
11 % Q generatore della catena di Markov
12 %
13 v=@(t) (sigma(1)^2-sigma(2)^2).*t+sigma(2)^2*T;
14 m=@(t) log(S0) +(r*T-0.5*v(t));
15
16 vT=(sigma(1)^2-sigma(2)^2)*T+sigma(2)^2*T;
17 mT= log(S0) +(r*T-0.5*vT);
18
19 v0=sigma(2)^2*T;
20 m0= log(S0) +r*T-0.5*v0;
21
22 rho=@(x,t) 1./sqrt(2*pi.*v(t)) .*exp( -((x-m(t)).^2) ./ (2.*v(t)));
23 rhoT=@(x) 1./sqrt(2*pi*vT) .*exp( -((x-mT).^2) ./ (2.*vT));
```

```

24 rho0=@(x) 1./sqrt(2*pi*v0) .*exp( -((x-m0).^2) ./ (2.*v0));
25
26 f0 =@(t) exp(-Q(2,1)*(T-t) -Q(1,2).*t) .*...
27 (Q(1,2) .* besseli(0,2*sqrt((Q(1,2)*Q(2,1).*t.*(T-t)))) +...
28 sqrt((Q(2,1)*Q(1,2).*t)./(T-t)) .* ...
29 besseli(1,2*sqrt((Q(1,2)*Q(2,1).*t.*(T-t)))));
30
31 f0s= exp(-Q(1,2)*T);
32
33 f1 =@(t) exp(-Q(2,1)*(T-t) -Q(1,2).*t) .*...
34 (Q(2,1) .* besseli(0,2*sqrt((Q(1,2)*Q(2,1).*t.*(T-t)))) +...
35 sqrt((Q(2,1)*Q(1,2).*t)./t) .* ...
36 besseli(1,2*sqrt((Q(1,2)*Q(2,1).*t.*(T-t)))));
37
38 f1s= exp(-Q(2,1)*T);
39
40 fun0=@(y,t) y./(y+K) .* rho(log(y+K),t).*f0(t);
41 fun0s=@(y) y./(y+K) .* rhoT(log(y+K)).*f0s;
42
43 fun1=@(y,t) y./(y+K) .* rho(log(y+K),t).*f1(t);
44 fun1s=@(y) y./(y+K) .* rho0(log(y+K)).*f1s;
45
46 Call1=exp(-r*T).*(integral2(fun0,0,Inf,0,T)+integral(fun0s,0,Inf));
47 Call2=exp(-r*T).*(integral2(fun1,0,Inf,0,T)+integral(fun1s,0,Inf));
48
49
50 end

```

```

1 function [ CallPrice1 , CallPrice2 ] = Call_Price_RS_FFT_JD...
2 (S0,K,r , sigma ,Q,T,lambda,mu,delta)
3 %-----
4 % Prezza una Call Europea in ipotesi di Regime Switching tra due stati
5 % tramite FFT nel modello di Merton jump diffusion
6 %-----
7 % S0 valore iniziale del sottostante
8 % K vettore degli strike
9 % r tasso risk free vettore 1x2
10 % sigma volatilita' vettore 1x2
11 % Q generatore della catena di Markov matrice 2x2
12 % T maturity
13 % lambda intensita' del processo di poisson vettore 1x2
14 % mu media dei salti del processo di salto vettore 1x2
15 % delta deviazione standard dei salti del processo di salto vettore 1x2
16 %-----
17 alpha=1.00;
18 dk=0.01;
19 N=4096;
20 du=2*pi/(dk*N);
21 k=(0:N-1)-N/2)*dk;
22 u=[0:N-1]*du;
23 v=u-1i*(1+alpha);
24

```



```

25 sort(K);
26 CallPrice1=zeros(1,length(K));
27 CallPrice2=zeros(1,length(K));
28 idx_in=1;
29 while(idx_in<length(K))
30     idx_in=idx_in+1;
31 end
32
33 A1=( (r(1)-r(2))+(0.5+alpha)*(sigma(1)^2-sigma(2)^2)).*u +...
34     0.5*u.^2*(sigma(1)^2-sigma(2)^2).*1i +...
35     ((r(1)-r(2))-(1+alpha)*(r(1)-r(2))-...
36     0.5*alpha*(1+alpha)*(sigma(1)^2-sigma(2)^2)).*1i;
37
38 B1=1i.*u.*(r(2)+(0.5+alpha)*sigma(2)^2)-0.5*u.^2.*sigma(2)^2+...
39     (1+alpha)*r(2)-r(2)+0.5*alpha*(1+alpha)*sigma(2)^2;
40
41 J1=exp(mu(1)+0.5*delta(1)^2)-1;
42 J2=exp(mu(2)+0.5*delta(2)^2)-1;
43 csi1=-lambda(1).*(1+1i.*v*J1) +...
44     lambda(1).*exp(0.5.*v.*(2*1i*mu(1)-delta(1)^2.*v));
45 csi2=-lambda(2).*(1+1i.*v*J2) +...
46     lambda(2).*exp(0.5.*v.*(2*1i*mu(2)-delta(2)^2.*v));
47 A2=-1i.*(csi1-csi2);
48 B2=-lambda(2).*(1+1i.*v*J2) +...
49     lambda(2).*exp(0.5*v.*(2*1i*mu(2)-delta(2)^2.*v));
50
51 A=A1+A2;
52 B=B1+B2;
53
54 s1= 0.5.*(-(Q(1,2)+Q(2,1)-1i.*A) +...
55     sqrt((Q(1,2)+Q(2,1)-1i.*A).^2 + 4*1i*Q(2,1).*A));
56 s2= 0.5.*(-(Q(1,2)+Q(2,1)-1i.*A) -...
57     sqrt((Q(1,2)+Q(2,1)-1i.*A).^2 + 4*1i*Q(2,1).*A));
58
59 phi1=((s1+Q(1,2)+Q(2,1)).*exp(s1.*T) -...
60     (s2+Q(1,2)+Q(2,1)).*exp(s2.*T))./(s1-s2);
61 phi2=((s1+Q(1,2)+Q(2,1)-1i.*A).*exp(s1.*T) -...
62     (s2+Q(1,2)+Q(2,1)-1i.*A).*exp(s2.*T))./(s1-s2);
63
64 %Primo Stato
65 ModCF1=(exp(B.*T).*phi1)./(alpha^2 + alpha -u.^2 + 1i.*(1+2*alpha).*u);
66
67 SimpsonW=zeros(1,N);
68 SimpsonW(1)=1/3;
69 for i=2:N
70
71     if(mod(i,2)==0)
72         SimpsonW(i)=4/3;
73     else
74         SimpsonW(i)=2/3;
75     end
76 end
77
78 FftFunc1=exp(1i.*pi/du.*u).*ModCF1.*SimpsonW*du;

```

```

79  payoff1=real ( fft ( FftFunc1 ) );
80
81  for i=1:idx_in
82      myk=log (K(i)/S0);
83      tmp=abs (k-myk);
84      [minimum_dummy,index]=min (tmp);
85
86      if ( i==1)
87          p_in=index;
88      end
89
90      if ( i==idx_in)
91          p_fi=index;
92      end
93
94
95      CallPrice1 ( i)=exp(-myk.*alpha) .*payoff1 (index) ./pi*S0;
96  end
97
98  %Secondo Stato
99  ModCF2= (exp (B.*T) .*phi2) ./ (alpha^2 + alpha -u.^2 + 1 i.*(1+2*alpha) .*u);
100
101  SimpsonW=zeros (1,N);
102  SimpsonW(1)=1/3;
103  for i=2:N
104
105      if (mod (i,2)==0)
106          SimpsonW ( i)=4/3;
107      else
108          SimpsonW ( i)=2/3;
109      end
110  end
111
112  FftFunc2=exp (1 i .*pi/du.*u) .*ModCF2.*SimpsonW*du;
113  payoff2=real ( fft ( FftFunc2 ) );
114
115  for i=1:idx_in
116      myk=log (K(i)/S0);
117      tmp=abs (k-myk);
118      [minimum_dummy,index]=min (tmp);
119      CallPrice2 ( i)=exp(-myk.*alpha) .*payoff2 (index) ./pi*S0;
120  end
121
122  end

```

```

1  function [Price1 ,Price2]=Call_Price_RS_FiniteDifference ...
2      (S0,K,r ,sigma ,delta ,T,theta ,Smin ,Smax ,Ndates ,N,Q,C,mu,d)
3  %-----
4  % Prezza un'opzione Call Europea nel modello di Merton jump diffusion nel
5  % caso in cui sia presente regime switching con il metodo delle differenze
6  % finite
7  %-----

```

```

8 % S0 valore iniziale del sottostante
9 % K vettore degli strike
10 % r tasso risk free vettore 1x2
11 % sigma volatilita' vettore 1x2
12 % delta dividendi
13 % T maturity
14 % theta valore per scegliere il theta metodo
15 % Smin lowerbound per la griglia spaziale
16 % Smax upperbound per la griglia spaziale
17 % Ndates numero di nodi griglia temporale
18 % N numero di nodi griglia spaziale
19 % Q generatore della catena di Markov matrice 2x2
20 % C intensita' del processo di poisson vettore 1x2
21 % mu media dei salti del processo di salto vettore 1x2
22 % d deviazione standard dei salti del processo di salto vettore 1x2
23 %
24 dt=T/Ndates;
25 t=linspace(0,T,Ndates+1);
26
27 dS=(Smax-Smin)/N;
28 S=linspace(Smin,Smax,N+1);
29 nodi=S(2:end-1)';
30
31 k1 = @(x) C(1)*exp(-(x-mu(1)).^2/(2*d(1)^2))/sqrt(2*pi*d(1)^2);
32 k2 = @(x) C(2)*exp(-(x-mu(2)).^2/(2*d(2)^2))/sqrt(2*pi*d(2)^2);
33
34 [alpha1,lambda1,Tmin1,Tmax1]=Integrali_Levy(k1,0,0,N);
35 [alpha2,lambda2,Tmin2,Tmax2]=Integrali_Levy(k2,0,0,N);
36
37 %PRIMA MATRICE
38
39 Dd1=-(r(1)-delta-alpha1)*nodi/(2*dS)+sigma(1)^2*nodi.^2/(2*dS^2);
40 Dd1=[Dd1(2:end);0];
41 Du1=(r(1)-delta-alpha1)*nodi/(2*dS)+sigma(1)^2*nodi.^2/(2*dS^2);
42 BcCall1=Du1(end);
43 Du1=[0;Du1(1:end-1)];
44 Dm1=sigma(1)^2*nodi.^2/(dS^2)+r(1)+lambda1;
45 tridiagA=[-theta*dt*Dd1,1+theta*dt*Dm1,-theta*dt*Du1];
46 A1=spdiags(tridiagA,-1:1,N-1,N-1);
47 tridiagB=[-(theta-1)*dt*Dd1,1+(theta-1)*dt*Dm1,-(theta-1)*dt*Du1];
48 B1=spdiags(tridiagB,-1:1,N-1,N-1);
49
50 %SECONDA MATRICE
51
52 Dd2=-(r(2)-delta-alpha2)*nodi/(2*dS)+sigma(2)^2*nodi.^2/(2*dS^2);
53 Dd2=[Dd2(2:end);0];
54 Du2=(r(2)-delta-alpha2)*nodi/(2*dS)+sigma(2)^2*nodi.^2/(2*dS^2);
55 BcCall2=Du2(end);
56 Du2=[0;Du2(1:end-1)];
57 Dm2=sigma(2)^2*nodi.^2/(dS^2)+r(2)+lambda2;
58 tridiagA=[-theta*dt*Dd2,1+theta*dt*Dm2,-theta*dt*Du2];
59 A2=spdiags(tridiagA,-1:1,N-1,N-1);
60 tridiagB=[-(theta-1)*dt*Dd2,1+(theta-1)*dt*Dm2,-(theta-1)*dt*Du2];
61 B2=spdiags(tridiagB,-1:1,N-1,N-1);

```

```

62
63 %———— Soluzione ————%
64 y1=max(nodi-K,0);
65 y2=max(nodi-K,0);
66
67 Bc1=zeros(N-1,1);
68 Bc2=zeros(N-1,1);
69
70 for j=Ndates-1:-1:0
71
72     if (C==0)
73         levy_term1=0;
74         levy_term2=0;
75     else
76         levy_term1=LevyMar(k1,nodi,y1,dS);
77         levy_term2=LevyMar(k2,nodi,y2,dS);
78     end
79
80     y_jf=Smax-K*exp(-r(1)*(T-j*dt));
81     y_jlf=Smax-K*exp(-r(1)*(T-(j+1)*dt));
82
83     y_js=Smax-K*exp(-r(2)*(T-j*dt));
84     y_jls=Smax-K*exp(-r(2)*(T-(j+1)*dt));
85
86
87     Bc1(end)=-(theta-1)*dt*BcCall1*y_jlf+dt*theta*BcCall1*y_jf;
88     Bc2(end)=-(theta-1)*dt*BcCall2*y_jls+dt*theta*BcCall2*y_js;
89
90
91     R1=(B1*y1+Bc1+dt*Q(1,1)*y1+dt*Q(1,2)*y2+ levy_term1*dt);
92     R2=(B2*y2+Bc2+dt*Q(2,2)*y2+dt*Q(2,1)*y1+ levy_term2*dt);
93
94     y1=A1\R1;
95     y2=A2\R2;
96
97 end
98
99 Price1=interp1(nodi,y1,S0,'spline');
100 Price2=interp1(nodi,y2,S0,'spline');
101
102 end

```

```

1 function [Price1,Price2]=Call_Price_RS_FiniteDifference_VC...
2     (S0,K,r,sigma,C,mu,delta,Q,T,Smin,Smax,N,M,flag)
3 %
4 % Prezza un'opzione Call Europea nel modello di Merton jump diffusion nel
5 % caso in cui sia presente regime switching con il metodo delle differenze
6 % finite. Utilizza un cambio di variabili. I parametri interessati a
7 % switching sono il drift e la volatilita'.
8 %
9 % S0 valore iniziale del sottostante
10 % K vettore degli strike

```

```

11 % r tasso risk free vettore 1x2
12 % sigma volatilita' vettore 1x2
13 % delta dividendi
14 % T maturity
15 % theta valore per scegliere il theta metodo
16 % Smin lowerbound per la griglia spaziale
17 % Smax upperbound per la griglia spaziale
18 % Ndates numero di nodi griglia temporale
19 % N numero di nodi griglia spaziale
20 % Q generatore della catena di Markov matrice 2x2
21 % C intensita' del processo di poisson scalare
22 % mu media dei salti del processo di salto scalare
23 % d deviazione standard dei salti del processo di salto scalare
24 %
25
26 dt=T/M;
27 xmin=log(Smin/K); xmax=log(Smax/K);
28 dx=(xmax-xmin)/N; x=linspace(xmin,xmax,N+1); nodi=x(2:end-1)';
29
30 k = @(y) C*exp(-(y-mu).^2/(2*delta^2))/sqrt(2*pi*delta^2);
31 [alpha,lambda,Tmin,Tmax]=integrali_Levy(k,xmin,xmax,N);
32
33 %%PRIMA MATRICE
34 trasp=0.5*sigma(1)^2+alpha;
35 a=(trasp>=0);
36 Dd1=0.5*sigma(1)^2/dx^2+a*trasp/dx;
37 Dm1=-sigma(1)^2/dx^2-(2*a-1)*trasp/dx-lambda;
38 Du1=0.5*sigma(1)^2/dx^2-(1-a)*trasp/dx;
39 e=ones(N-1,1);
40 tridiagD=[Dd1*e Dm1*e Du1*e];
41 D=spdiags(tridiagD,-1:1,N-1,N-1);
42 A1=speye(N-1)-dt*D;
43 clear D;
44
45 %%SECONDA MATRICE
46 trasp=0.5*sigma(2)^2+alpha;
47 a=(trasp>=0);
48 Dd2=0.5*sigma(2)^2/dx^2+a*trasp/dx;
49 Dm2=-sigma(2)^2/dx^2-(2*a-1)*trasp/dx-lambda;
50 Du2=0.5*sigma(2)^2/dx^2-(1-a)*trasp/dx;
51 e=ones(N-1,1);
52 tridiagD=[Dd2*e Dm2*e Du2*e];
53 D=spdiags(tridiagD,-1:1,N-1,N-1);
54 A2=speye(N-1)-dt*D;
55 clear D;
56
57
58 u1=max(exp(nodi)-1,0);
59 u2=max(exp(nodi)-1,0);
60
61
62 b1=zeros(N-1,1);
63 b2=zeros(N-1,1);
64

```

```

65
66 b1(end)=Du1*(exp(xmax)-1);
67 b2(end)=Du2*(exp(xmax)-1);
68
69
70 for j=1:M
71
72     if C==0
73         J1=0;
74         J2=0;
75     else
76
77         if flag==1
78
79             J1=integrale_Levy(k,Tmin,Tmax,N,x,[0;u1;exp(xmax)-1]);
80             J2=integrale_Levy(k,Tmin,Tmax,N,x,[0;u2;exp(xmax)-1]);
81
82
83         else
84             J1=integrale_Levy(k,Tmin,Tmax,N,x,[0;u1;1-exp(xmin)]);
85             J2=integrale_Levy(k,Tmin,Tmax,N,x,[0;u2;1-exp(xmin)]);
86         end
87
88     end
89     u1=A1\((u1+dt*Q(1,1)*u1+Q(1,2)*dt*u2+dt*(J1+b1));
90     u2=A2\((u2+dt*Q(2,1)*u1+Q(2,2)*dt*u2+dt*(J2+b2));
91 end
92
93 S1=K*exp(nodi-r(1)*T);
94 S2=K*exp(nodi-r(2)*T);
95 C1=K*u1*exp(-r(1)*T);
96 C2=K*u2*exp(-r(2)*T);
97
98 Price1=interp1(S1,C1,S0,'spline');
99 Price2=interp1(S2,C2,S0,'spline');
100 end
101
102
103 function J=integrale_Levy(k,Tmin,Tmax,N,x,u)
104 %
105 % Funzione per il calcolo del termine integrale dovuto alla presenza di
106 % salti nella dinamica del sottostante
107 %
108 y=linspace(Tmin,Tmax,N);
109 w=ones(1,N)*(y(2)-y(1)); w(1)=0.5*w(1); w(end)=0.5*w(end);
110 J=zeros(N-1,1);
111 w=w.*k(y);
112 for i=1:N-1
113     J(i)=sum(w.*f_u(x(i+1)+y,x,u));
114 end
115 end
116
117 function f=f_u(z,x,u)
118 %

```

```

119 % Funzione per il calcolo della soluzione numerica al di fuori del dominio
120 %-----
121
122 f=zeros(size(z));
123 index=find( (z>=x(1)).*(z<=x(end)) );
124 f(index)=interp1(x,u,z(index));
125 index=find( (z>x(end)) );
126 f(index)=exp(z(index))-1;
127
128
129 end

1 function [ Call , IC_Call , Sminass , Smaxass ] = Call_Price_RS_MonteCarlo ...
2 ( S0 , K , r , T , sigma , mu , d , lambda , Q , Nsim , Ndates , k )
3 %-----
4 % Prezza un'opzione Call Europea con metodo Monte Carlo e con sottostante
5 % che evolve secondo un modello di Merton jump diffusion
6 % S0 sottostante
7 % K strike
8 % r tasso risk free vettore 1x2
9 % T maturity
10 % sigma volatilita' vettore 1x2
11 % mu media dei salti
12 % d varianza dei salti
13 % lambda frequenza dei salti vettore 1x2
14 % Q generatore catena di Markov matrice 2x2
15 % Nsim numero simulazioni Monte Carlo
16 % Ndates numero passi temporali
17 % k regime iniziale
18 %-----
19 Callprices=zeros(1,Nsim);
20 Smin=zeros(1,Nsim);
21 Smax=zeros(1,Nsim);
22 dt=T/Ndates;
23
24 for i=1:Nsim
25     [ SinglePath , dswitch , s , Smin(i) , Smax(i) ] = SinglePathJDRegimeSwitching ...
26         ( S0 , r , sigma , mu , d , T , Q , lambda , Ndates , k );
27     Callprices(i)=max( SinglePath(end)-K,0);
28     L=length(dswitch);
29     for j=0:L-2
30         Callprices(i)=exp(-r(s)*(dswitch(L-j)-dswitch(L-j-1))*dt)* ...
31             Callprices(i);
32         if(s==1)
33             s=2;
34         else
35             s=1;
36         end
37     end
38 end
39
40 [ Call , dummy , IC_Call ] = normfit( Callprices );

```

```

41 Sminass=min(Smin);
42 Smaxass=max(Smax);
43
44
45 end

```

```

1  function [SinglePath,dswitch,finalstate,SminPath,SmaxPath] = ...
2      SinglePathJDRegimeSwitching (S0,r,sigma,mu,d,T,Q,lambda,Ndates,k)
3  % -----
4  % Genera una singola traiettoria di un Merton Jump diffusion in regime
5  % switch in cui switchano tutti i parametri
6  % -----
7  % S0 sottostante
8  % K strike
9  % r tasso risk free vettore 1x2
10 % delta dividendo
11 % sigma volatilita' vettore 1x2
12 % mu media dei salti
13 % d varianza dei salti
14 % T maturity
15 % Q generatore catena di Markov matrice 2x2
16 % lambda frequenza dei salti vettore 1x2
17 % Ndates numero passi temporali
18 % k regime iniziale
19 % -----
20 dt=T/Ndates;
21
22 SinglePath=zeros(1,Ndates);
23 SinglePath(1)=S0;
24
25 [tau1,Nt1]=JumpTimeGen(T,lambda(1));
26 t=0:dt:T;
27 for i=1:Nt1
28     temp=abs(t-tau1(i));
29     minimum=min(temp);
30     ik1(i)=find(temp==minimum);
31 end
32
33 if(Nt1~=0)
34     tauj1=t(ik1);
35 end
36
37 [tau2,Nt2]=JumpTimeGen(T,lambda(2));
38 t=0:dt:T;
39 for i=1:Nt2
40     temp=abs(t-tau2(i));
41     minimum=min(temp);
42     ik2(i)=find(temp==minimum);
43 end
44
45 if(Nt2~=0)
46     tauj2=t(ik2);

```



```

47 end
48
49 brow=randn(1,Ndates);
50
51 switchtime=exprnd(1/abs(Q(k,k)));
52
53 templ=abs(switchtime-t);
54 minimum1=min(templ);
55 dswitch(1)=find(templ==minimum1);
56 contswitch=1;
57
58 for j=1:Ndates
59
60     if (t(j)==t(dswitch(contswitch)))
61         contswitch=contswitch+1;
62         if (k==2)
63             k=1;
64         else
65             k=2;
66         end
67
68         switchtime=t(j)+exprnd(1/abs(Q(k,k)));
69         templ=abs(switchtime-t);
70         minimum1=min(templ);
71         dswitch(contswitch)=find(templ==minimum1);
72     end
73
74     SinglePath(j+1)=SinglePath(j).*exp((r(k)-(sigma(k)^2)/2-...
75         lambda(k)*(exp((d(k)^2)/2+mu(k))-1))*dt + ...
76         sigma(k)*sqrt(dt).*brow(j));
77
78
79     if (k==1)
80
81         for i=1:Nt1
82             if (tau1(i)==t(j))
83                 Y=mu(k)+d(k) * randn;
84                 SinglePath(j+1)=SinglePath(j+1).*exp(Y);
85             end
86         end
87
88     else
89         for i=1:Nt2
90             if (tau2(i)==t(j))
91                 Y=mu(k)+d(k) * randn;
92                 SinglePath(j+1)=SinglePath(j+1).*exp(Y);
93             end
94         end
95     end
96
97 end
98
99 end
100 SminPath=min(SinglePath);

```

```

101 SmaxPath=max( SinglePath );
102 dswitch=[0,dswitch ];
103 finalstate=k;
104 end

```

B.2 Opzioni Americane

```

1  function [Put,IC_Put,minimoASS,massimoASS] =...
2      American_Put_RegimeSwitching_MC...
3      (S0,K,r,delta,T,sigma,mu,d,lambda,Q,Nsim,Ndates,k)
4  %
5  % Prezza con il montecarlo un'opzione put americana secondo l'algoritmo di
6  % Longstaff-Schwartz in ipotesi di regime switching e modello di Merton
7  % jump diffusion
8  %
9  % S0 sottostante
10 % K strike
11 % r tasso privo di rischio
12 % delta dividendo
13 % T maturity
14 % sigma volatilita'
15 % lambda salti del processo di poisson
16 % Q matrice di switch
17 % Nsim numero simulazioni del montecarlo
18 % Ndates numero di date del montecarlo
19 % k stato di Partenza
20 %
21 dt=T/Ndates;
22 SPath=zeros(Nsim,Ndates+1);
23 states=zeros(Nsim,Ndates+1);
24 finalstate=zeros(1,Nsim);
25
26 for i=1:Nsim
27     [SPath(i,:),states(i,:),finalstate(i)]=...
28         SinglePathJDRegimeSwitching_ALL...
29         (S0,r,delta,sigma,mu,d,T,Q,lambda,Ndates,k);
30 end
31
32 massimoV=max(SPath,[],1);
33 minimoV=min(SPath,[],1);
34 minimoASS=min(minimoV);
35 massimoASS=max(massimoV);
36
37 %Grado dei polinomi 3
38 SPath=SPath(:,2:end);
39 ExerciseTime=Ndates*ones(Nsim,1);
40 CashFlows=max(K-SPath(:,Ndates),0);
41
42 for j=Ndates-1:-1:1
43
44     InMoney=find(SPath(:,j)<K);
45     S=SPath(InMoney,j);

```

```

46
47     RegMatrix=[ones(length(S),1),S,S.^2];
48
49     Ydata=zeros(size(InMoney));
50
51     for i=1:length(InMoney)
52         k=states(i,j);
53         Ydata(i)=exp(-r(k)*dt*(ExerciseTime(InMoney(i))-j)).*...
54             CashFlows(InMoney(i));
55     end
56
57     alpha=RegMatrix\Ydata;
58
59     IV=K-S;
60     CV=RegMatrix*alpha;
61     Index=find(IV>CV);
62     ExercisePath=InMoney(Index);
63     CashFlows(ExercisePath)=IV(Index);
64     ExerciseTime(ExercisePath)=j;
65
66 end
67
68 Prices=CashFlows;
69
70 for i=1:length(Prices)
71
72     s=states(i,ExerciseTime(i));
73     ExerciseTime(i);
74
75     for j=ExerciseTime(i):-1:1
76         s=states(i,j);
77         Prices(i)=exp(-r(s)*dt)* Prices(i);
78
79     end
80
81 end
82
83 h=mean(Prices);
84 sd=var(Prices);
85 IC_Put(1)=h-sd/sqrt(Nsim);
86 IC_Put(2)=h+sd/sqrt(Nsim);
87 Put=max(K-S0,mean(Prices));
88
89
90 end

1 function [SinglePath,state,finalstate,SminPath,SmaxPath] =...
2     SinglePathJDRegimeSwitching_ALL...
3     (S0,r,delta,sigma,mu,d,T,Q,lambda,Ndates,k)
4 %
5 % Genera una traiettoria del sottostante nel caso in cui esso evolva
6 % secondo modello di Merton jump diffusion e sia presente switch sul tasso

```

```

7  % risk-free sul dividendo e sulla varianza
8  % S0 valore iniziale sottostante
9  % r tasso risk-free
10 % delta dividendo vettore 1x2
11 % sigma volatilita' vettore 1x2
12 % mu ampiezza dei salti
13 % d deviazione standard dei salti
14 % Q generatore della catena di Markov matrice 2x2
15 % lambda intensita' dei salti
16 % T maturity
17 % Ndates time points
18 % k regime di partenza
19 %
20
21 dt=T/Ndates;
22
23 SinglePath=zeros(1,Ndates);
24 SinglePath(1)=S0;
25 [tau,Nt]=JumpTimeGen(T,lambda);
26 state=zeros(Ndates+1,1);
27 state(1)=k;
28
29 t=0:dt:T;
30
31 for i=1:Nt
32     temp=abs(t-tau(i));
33     minimum=min(temp);
34     ik(i)=find(temp==minimum);
35 end
36
37 if(Nt~=0)
38
39     tauj=t(ik);
40 end
41
42 brow=randn(1,Ndates);
43
44 if(Q(k,k)~=0)
45     switchtime=exprnd(1/abs(Q(k,k)));
46 else
47     switchtime=T*2;
48 end
49
50 templ=abs(switchtime-t);
51 minimum1=min(templ);
52 dswitch(1)=find(templ==minimum1);
53 contswitch=1;
54
55 for j=1:Ndates
56
57     state(j+1)=k;
58     if(t(j)==t(dswitch(contswitch)))
59
60         contswitch=contswitch+1;

```

```

61         if (k==2)
62             k=1;
63         else
64             k=2;
65         end
66
67         if (Q(k,k)~=0)
68             switchtime=t(j)+exprnd(1/abs(Q(k,k)));
69             temp1=abs(switchtime-t);
70             minimum1=min(temp1);
71             dswitch(contswitch)=find(temp1==minimum1);
72         else
73             switchtime=T*2;
74             temp1=abs(switchtime-t);
75             minimum1=min(temp1);
76             dswitch(contswitch)=find(temp1==minimum1);
77
78         end
79
80
81     end
82
83     SinglePath(j+1)=SinglePath(j).*exp((r(k)-delta(k)-...
84         (sigma(k)^2)/2-lambda*(exp((d^2)/2+mu)-1))*dt +...
85         sigma(k)*sqrt(dt).*brow(j));
86
87
88
89     for i=1:Nt
90         if (tau_j(i)==t(j))
91             Y=mu+d * randn;
92             SinglePath(j+1)=SinglePath(j+1).*exp(Y);
93         end
94     end
95
96 end
97 SminPath=min(SinglePath);
98 SmaxPath=max(SinglePath);
99 dswitch=[0,dswitch];
100 finalstate=k;
101 end

```

```

1 function [Price1,Price2]=American_Put_RegimeSwitching_MOL...
2     (S0,K,r,dividend,sigma,T,N,M,Smin,Smax,Q,C,mu,delta)
3 %
4 % Prezza una Put europea con il metodo delle linee in ipotesi di regime
5 % switching sul tasso di interesse risk free e sulla volatilita'
6 % S0 valore iniziale sottostante
7 % K stike price
8 % r tasso risk-free
9 % dividend dividendo
10 % sigma volatility

```

```

11 % T maturity
12 % N time points
13 % M spatial points
14 % Smin lowerbound griglia dei sottostanti
15 % Smax upperbound griglia dei sottostanti
16 % Q generatore della catena di Markov
17 % C intensita' dei salti
18 % mu ampiezza dei salti
19 % delta deviazione standard dei salti
20 %
21
22 X=Smax/K;
23 dtau=T/N;
24 dx=X/M;
25
26 if (C==0)
27     lambda=0;
28     alpha=0;
29 else
30     k = @(x) C*exp(-(x-mu).^2/(2*delta^2))/sqrt(2*pi*delta^2);
31     [alpha,lambda,Tmin,Tmax]=Integrali_Levy(k,0,0,M);
32 end
33
34 a1=@(x)0;
35 b1=@(x)1;
36 c1=@(x) 1/(max(1.e-6,0.5*sigma(1)^2*x.^2)) * ...
37     (r(1)-dividend(1)+lambda+1/dtau);
38 d1=@(x) (r(1)-dividend(1)-alpha)*x/(max(1.e-6,0.5*sigma(1)^2*x.^2));
39 a2=@(x)0;
40 b2=@(x)1;
41 c2=@(x) 1/(max(1.e-6,0.5*sigma(2)^2*x.^2)) * ...
42     (r(2)-dividend(2)+lambda+1/dtau);
43 d2=@(x) (r(2)-dividend(2)-alpha)*x/(max(1.e-6,0.5*sigma(2)^2*x.^2));
44 lambda1= -(r(1)-dividend(1)-0.5*sigma(1)^2)-sqrt((r(1)-...
45     dividend(1)-0.5*sigma(1)^2)^2+...
46     2*(r(1)-dividend(1))*sigma(1)^2))/sigma(1)^2;
47 lambda2= -(r(2)-dividend(2)-0.5*sigma(2)^2)-sqrt((r(2)-...
48     dividend(2)-0.5*sigma(2)^2)^2+...
49     2*(r(2)-dividend(2))*sigma(2)^2))/sigma(2)^2;
50 s_inf1=lambda1/(lambda1-1);
51 s_inf2=lambda2/(lambda2-1);
52
53 if (lambda==0)
54     s_inf=min(s_inf1,s_inf2);
55 else
56     s_inf=Smin/K;
57 end
58
59 xnodes1=linspace(s_inf,X,M+1);
60 xnodes2=linspace(s_inf,X,M+1);
61 R1=ODE_Solver_R (xnodes1,a1,b1,c1,d1,0,-1);
62 R2=ODE_Solver_R (xnodes2,a2,b2,c2,d2,0,-1);
63
64 u01=max(1-xnodes1,0);

```

```

65  u02=max(1-xnodes2,0);
66  s1d=s_inf;
67  s2d=s_inf;
68
69  for l=1:N
70      if (lambda==0)
71          levy_term1=0;
72          levy_term2=0;
73
74          g1d= -1./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)) .* u01./dtau ...
75              - Q(1,1)./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)).* u01 - ...
76              Q(1,2)./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)).*u02;
77          g2d= -1./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)) .* u02./dtau ...
78              - Q(2,1)./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)).* u01 - ...
79              Q(2,2)./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)).*u02;
80
81      else
82          levy_term1=LevyTerm(k,xnodes1,u01,dx);
83          levy_term2=LevyTerm(k,xnodes2,u02,dx);
84
85          g1d= -1./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)) .* u01./dtau ...
86              - Q(1,1)./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)).* u01 - ...
87              Q(1,2)./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2)).*u02 - ...
88              levy_term1./(max(1.e-6,0.5*sigma(1)^2*xnodes1.^2));
89          g2d= -1./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)) .* u02./dtau ...
90              - Q(2,1)./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)).* u01 - ...
91              Q(2,2)./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)).*u02 - ...
92              levy_term2./(max(1.e-6,0.5*sigma(2)^2*xnodes2.^2)) ;
93
94      end
95
96
97      X0=0;
98      w1=ODE_Solver_w(xnodes1,a1,c1,X0,R1,g1d);
99      w2=ODE_Solver_w(xnodes2,a2,c2,X0,R2,g2d);
100
101
102      s1d=mynewsd(xnodes1,s1d,R1,w1);
103      s2d=mynewsd(xnodes2,s2d,R2,w2);
104
105      temp1=abs(xnodes1-s1d);
106      temp2=abs(xnodes2-s2d);
107
108      [val,index1]=min(temp1);
109      [val,index2]=min(temp2);
110
111
112      v1=ODE_Solver_v(xnodes1(index1:end),c1,d1,-1,R1(index1:end),...
113          w1(index1:end),g1d(index1:end));
114      v2=ODE_Solver_v(xnodes2(index2:end),c2,d2,-1,R2(index2:end),...
115          w2(index2:end),g2d(index2:end));
116      u01=zeros(1,M+1);
117      u02=zeros(1,M+1);
118

```

```

119     for j=1:M
120
121         if (xnodes1(j)<=s1d)
122             u01(j)=max(1-xnodes1(j),0);
123         else
124             u01(j)=R1(j)*v1(j-index1+1)+w1(j);
125         end
126     end
127
128     for j=1:M
129
130         if (xnodes2(j)<=s2d)
131             u02(j)=max(1-xnodes2(j),0);
132         else
133             u02(j)=R2(j)*v2(j-index2+1)+w2(j);
134         end
135     end
136
137
138 end
139
140 S1=K*xnodes1;
141 S2=K*xnodes2;
142 sol1=K*u01;
143 sol2=K*u02;
144 Price1=interp1(S1,sol1,S0);
145 Price2=interp1(S2,sol2,S0);
146
147 end
148
149
150
151 function s=mynewsd(xnodes,sold,R,w)
152
153 s=sold;
154 Nn=length(xnodes);
155 phi=R-w+(1-xnodes');
156
157 for i=1:Nn-1
158
159     if (phi(i)*phi(i+1)<0)
160         s=xnodes(i+1);
161         nodi=[xnodes(i),xnodes(i+1)];
162         valori=[phi(i),phi(i+1)];
163         x=linspace(nodi(1),nodi(end),1000);
164         pol=Lag_interp(nodi,valori,x);
165         pol=abs(pol);
166         [val,index]=min(pol);
167         s=x(index);
168         break;
169     end
170
171 end
172

```



```

173 end
174
175 function u=ODE_Solver_R (x,a,b,c,d,X0,direction)
176 %-----
177 % Risolutore veloce della ODE di Riccati.
178 % x nodi su cui calcolare la soluzione
179 % a,b,c,d funzioni del cambio di coordinate secondo Riccati
180 % X0 valore iniziale
181 % direction direzione di integrazione 1 in avanti -1 all'indietro
182 %-----
183
184 if (direction==-1)
185     N_nodes=length(x);
186     u=zeros(N_nodes,1);
187     u(end)=X0;
188
189     for j=N_nodes-1:-1:1
190
191         alpha_j= c(x(j))/2;
192         beta_j=1/(x(j)-x(j+1)) - (a(x(j))+d(x(j)))/2;
193         gamma_j= c(x(j+1))*u(j+1)^2*0.5 - (1/(x(j)-x(j+1)) +...
194             (a(x(j+1))+d(x(j+1)))/2)*u(j+1) - (b(x(j))+b(x(j+1)))/2;
195         u(j)=-2*gamma_j/(beta_j - sqrt(beta_j^2-4*alpha_j*gamma_j));
196
197     end
198
199 end
200
201 end
202
203 function u=ODE_Solver_w(x,a,c,X0,R,g)
204 %-----
205 % Risolve l'equazione di w con il metodo veloce
206 %-----
207
208 N_nodes=length(x);
209 u=zeros(N_nodes,1);
210 u(end)=X0;
211
212 for j=N_nodes-1:-1:1
213
214     u(j)= 1/(1-0.5*(x(j)-x(j+1))*(a(x(j))-c(x(j))*R(j)))*...
215         (u(j+1)+ 0.5*(x(j)-x(j+1))*((a(x(j+1))-c(x(j+1))*R(j+1))*u(j+1)...
216             - (R(j)*g(j)+R(j+1)*g(j+1))));
217
218 end
219
220 end
221
222 function u=ODE_Solver_v(x,c,d,X0,R,w,g)
223 %-----
224 % Risolve l'equazione di v con il metodo veloce
225 %-----
226 N_nodes=length(x);

```

```

227 u=zeros(N_nodes,1);
228 u(1)=X0;
229
230 for j=2:N_nodes
231
232     u(j)= 1/(1-0.5*(x(j)-x(j-1))*(c(x(j))*R(j)-d(x(j)))) *...
233           (u(j-1) + 0.5*(x(j)-x(j-1)) * (c(x(j))*w(j)+ g(j) +c(x(j-1))*...
234           (R(j-1)*u(j-1)+w(j-1))-d(x(j-1))*u(j-1)+g(j-1)));
235
236 end
237
238 end
239
240
241 function sum=Lag_interp (xi,yi,x)
242 %-----
243 % Valuta il polinomio interpolante di Lagrange
244 % xi valori delle ascisse noti
245 % yi valori delle ordinate noti
246 % x valori in cui calcolare il valore del polinomio
247 %-----
248
249 n = length(xi);
250 m = length(x);
251 sum = zeros(m,1);
252 for j = 1:m
253     sum(j) = 0;
254     for k = 1:n
255         % Valutazione del j-esimo polinomio di Lagrange
256         space = [1:k-1, k+1:n];
257         pr = prod(x(j) - xi(space))/prod(xi(k) - xi(space));

```



```

1 function [Put1,Put2]=American_Put_RegimeSwitching_SOR...
2     (S0,K,r,sigma,dividend,T,theta,Smin,Smax,M,N,flag,Q,C,mu,delta)
3 %-----
4 % Prezza un'opzione Put o Call Americana in ipotesi di Regime switching con
5 % sottostante che evolve secondo modello di Merton jump diffusion con il
6 % metodo del SOR proiettato
7 %-----
8 % S0 Valore iniziale del sottostante
9 % K Strike Price
10 % r tasso risk-free vettore 1 x 2
11 % sigma volatilita' vettore 1 x 2
12 % dividend dividendo vettore 1 x 2
13 % T Maturity
14 % theta 1 implicito 0 esplicito
15 % Smin Estremo inferiore di computazione
16 % Smax Estremo superiore di computazione
17 % M numero di punti discretizzazione temporale
18 % N numero di punti discretizzazione spaziale
19 % flag 1 per la Call -1 per la Put
20 % Q matrice di Switch

```

```

21 % C frequenza di salto del processo di Jump diffusion
22 % mu media dei salti
23 % delta deviazione standard dei salti
24 %
25 dt=T/M;
26 dS=(Smax-Smin)/N;
27 S=linspace(Smin,Smax,N+1);
28 nodi=S(2:end-1)';
29 k = @(x) C*exp(-(x-mu).^2/(2*delta^2))/sqrt(2*pi*delta^2);
30 [alpha,lambda,Tmin,Tmax]=Integrali_Levy(k,0,0,N);
31
32 %PRIMA MATRICE
33
34 Dd1=-(r(1)-dividend(1)-alpha)*nodi/(2*dS)+sigma(1)^2*nodi.^2/(2*dS^2);
35 BcPut1=Dd1(1);
36 Dd1=[Dd1(2:end);0];
37 Du1=(r(1)-dividend(1)-alpha)*nodi/(2*dS)+sigma(1)^2*nodi.^2/(2*dS^2);
38 BcCall1=Du1(end);
39 Du1=[0;Du1(1:end-1)];
40 Dm1=sigma(1)^2*nodi.^2/(dS^2)+r(1)+lambda;
41 tridiagA=[-theta*dt*Dd1,1+theta*dt*Dm1,-theta*dt*Du1];
42 A1=spdiags(tridiagA,-1:1,N-1,N-1);
43 tridiagB=[-(theta-1)*dt*Dd1,1+(theta-1)*dt*Dm1,-(theta-1)*dt*Du1];
44 B1=spdiags(tridiagB,-1:1,N-1,N-1);
45
46 %SECONDA MATRICE
47
48 Dd2=-(r(2)-dividend(2)-alpha)*nodi/(2*dS)+sigma(2)^2*nodi.^2/(2*dS^2);
49 BcPut2=Dd2(1);
50 Dd2=[Dd2(2:end);0];
51 Du2=(r(2)-dividend(2)-alpha)*nodi/(2*dS)+sigma(2)^2*nodi.^2/(2*dS^2);
52 BcCall2=Du2(end);
53 Du2=[0;Du2(1:end-1)];
54 Dm2=sigma(2)^2*nodi.^2/(dS^2)+r(2)+lambda;
55 tridiagA=[-theta*dt*Dd2,1+theta*dt*Dm2,-theta*dt*Du2];
56 A2=spdiags(tridiagA,-1:1,N-1,N-1);
57 tridiagB=[-(theta-1)*dt*Dd2,1+(theta-1)*dt*Dm2,-(theta-1)*dt*Du2];
58 B2=spdiags(tridiagB,-1:1,N-1,N-1);
59
60 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
61
62 y1=max(flag*(nodi-K),0);
63 y2=max(flag*(nodi-K),0);
64 Bc1=zeros(N-1,1);
65 Bc2=zeros(N-1,1);
66
67 for j=M-1:-1:0
68
69     if (C==0)
70         levy_term1=0;
71         levy_term2=0;
72     else
73         levy_term1=LevyMar(k,nodi,y1,dS);
74         levy_term2=LevyMar(k,nodi,y2,dS);

```

```

75     end
76
77
78     if flag==1
79         y_jf=Smax-K*exp(-r(1)*(T-j*dt));
80         y_jf1=Smax-K*exp(-r(1)*(T-(j+1)*dt));
81         y_js=Smax-K*exp(-r(2)*(T-j*dt));
82         y_js1=Smax-K*exp(-r(2)*(T-(j+1)*dt));
83
84         Bc1(end)=-(theta-1)*dt*BcCall1*y_jf1+dt*theta*BcCall1*y_jf;
85         Bc2(end)=-(theta-1)*dt*BcCall2*y_js1+dt*theta*BcCall2*y_js;
86
87
88     else
89         y_jf=K*exp(-r(1)*(T-j*dt))-Smin;
90         y_jf1=K*exp(-r(1)*(T-(j+1)*dt))-Smin;
91         y_js=K*exp(-r(2)*(T-j*dt))-Smin;
92         y_js1=K*exp(-r(2)*(T-(j+1)*dt))-Smin;
93
94         Bc1(1)=theta*dt*BcPut1*y_jf-(theta-1)*dt*BcPut1*y_jf1;
95         Bc2(1)=theta*dt*BcPut2*y_js-(theta-1)*dt*BcPut2*y_js1;
96     end
97
98
99
100     R1=(B1*y1+Bc1+dt*Q(1,1)*y1+dt*Q(1,2)*y2+ levy_term1*dt);
101     R2=(B2*y2+Bc2+dt*Q(2,2)*y2+dt*Q(2,1)*y1+ levy_term2*dt);
102
103
104     MAXITER=10000;
105     w=1.2;
106     TOL=1.e-5;
107
108
109     for jj=1:MAXITER
110         y_old1=y1;
111         for ii=1:length(A1)
112             if ii==1
113                 z=(R1(ii)-A1(ii,ii+1)*y1(ii+1))/A1(ii,ii);
114             elseif ii==length(A1)
115                 z=(R1(ii)-A1(ii,ii-1)*y1(ii-1))/A1(ii,ii);
116             else
117                 z=(R1(ii)-A1(ii,ii-1)*y1(ii-1)-...
118                     A1(ii,ii+1)*y1(ii+1))/A1(ii,ii);
119             end
120             y1(ii)=max(flag*(nodi(ii)-K),y1(ii)+w*(z-y_old1(ii)));
121
122         end
123         if norm(y1-y_old1)<TOL
124             break;
125         end
126
127     end
128

```

```

129
130     for jj=1:MAXITER
131         y_old2=y2;
132         for ii=1:length(A2)
133             if ii==1
134                 z=(R2(ii)-A2(ii,ii+1)*y2(ii+1))/A2(ii,ii);
135             elseif ii==length(A2)
136                 z=(R2(ii)-A2(ii,ii-1)*y2(ii-1))/A2(ii,ii);
137             else
138                 z=(R2(ii)-A2(ii,ii-1)*y2(ii-1)-...
139                   A2(ii,ii+1)*y2(ii+1))/A2(ii,ii);
140             end
141             y2(ii)=max(flag*(nodi(ii)-K),y2(ii)+w*(z-y_old2(ii)));
142
143         end
144         if norm(y2-y_old2)<TOL
145             break;
146         end
147     end
148
149 end
150
151 Put1=interp1(nodi,y1,S0);
152 Put2=interp1(nodi,y2,S0);
153
154 end
155
156
157 function [alpha,lambda,Tmin,Tmax]=Integrali_Levy(k,xmin,xmax,N)

```

B.3 Funzioni Ausiliarie

```

1 function [JumpTimes,NT]=JumpTimeGen(T,lambda)
2 %-----
3 % Genera i tempi di salto tra [0,T] secondo l'algoritmo di countdown
4 % simulation
5 % T istante finale
6 % lambda frequenza dei salti
7 %-----
8
9 tau(1)=0;
10 i=0;
11 while (tau(i+1)<=T)
12
13     i=i+1;
14     tau(i+1)=tau(i)-log(rand)/lambda;
15
16 end
17
18 JumpTimes=tau(2:end-1);
19 NT=i-1;
20 end

```

```

1  function I=LevyTerm(k,S,P,dS)
2  %
3  % Calcola il termine integrale delle PIDE
4  %
5  N=length(S);
6  w=ones(N,1); w(1)=0.5; w(N)=0.5;
7  I=zeros(N,1);
8
9  for j=1:N
10     I=I+w(j)*dS*(P(j)).*k(log(S(j)./S(1:end)))./S(j);
11 end
12 end

1  function [alpha,lambda,Tmin,Tmax]=Integrali_Levy(k,xmin,xmax,N)
2  %
3  % Calcola i Termini Integrali alpha e lambda del processo di Levy
4  %
5  tol=1e-8;
6  Tmin=xmin;
7  while (k(Tmin)>tol)
8     Tmin=Tmin-1;
9  end
10 Tmax=xmax;
11 while (k(Tmax)>tol)
12     Tmax=Tmax+1;
13 end
14
15 y=linspace(Tmin,Tmax,N);
16 w=ones(1,N)*(y(2)-y(1)); w(1)=0.5*w(1); w(end)=0.5*w(end);
17 alpha=sum(w.*k(y).*(exp(y)-1));
18 lambda=sum(w.*k(y));
19 end

```

Bibliografia

- [1] S. Asmussen e J. Rosiński. «Approximations of small jumps of Lévy processes with a view towards simulation». In: *J. Appl. Probab.* 38 (2001), pp. 482–493.
- [2] F. G. Ball et al. «Bayesian inference for ion-channel gating mechanisms directly from single-channel recordings, using Markov chain Monte Carlo». In: *Proceedings of the Royal Society Series* 455 (1999).
- [3] L. E. Baum, J. Sass e T. Petrie. «Statistical inference for probabilistic function of finite state Markov chains». 1966.
- [4] A Bensoussan e J.-L. Lions. *Contrôle Impulsionnel et Inéquations Quasi-Variationnelles*. Paris: Dumond.
- [5] Thomas Björk. *Arbitrage Theory in Continuous Time*. Oxford University Press, 2009.
- [6] Black, Scholes e Merton. «The Pricing of Options and Corporate Liabilities». In: *Journal of Political Economy* 81 3 (1973), pp. 637–654.
- [7] J. Buffington e J. Elliot. «American options with regime switching». In: *International Journal of Theoretical and Applied Finance* 5 (2002), pp. 497–514.
- [8] L. Carmellino. *Metodi Montecarlo in Finanza*. URL: www.mat.uniroma2.it/~caramell.
- [9] P. Carr e D.B. Madann. «Option valuation using fast Fourier transform». In: *Journal of Computation Finance* 2 4 (1998), pp. 61–73.
- [10] F. Cheng-Der et al. «Option pricing with Markov switching». In: *Journal of Data Science* 10 (2012), pp. 483–509.
- [11] C. Chiarella et al. «The evaluation of american option prices under stochastic volatility and jump-diffusion dynamics using the method of lines». In: *International Juornale of Theretical and Applied Finance* 12.3 (2009), pp. 393–425.

- [12] R. Cont e P. Tankov. *Financial Modelling with Jump Process*. Chapman & Hall, 2004.
- [13] C.W. Cryer. «The solution of quadratic programming problem using systematic overrelaxion». In: *SIAM J. Control* 9 (1971), pp. 385–392.
- [14] C. M. Elliot e J.R Ockedon. *Weak and variational methods for moving boundary problems*. Boston: Pitman Pub., 1982.
- [15] R. J. Elliott, W. C. Hunter e B. M. Jamieson. «Drift and volatility estimation in discrete time». In: *Journal of Economic Dynamics and Control* 22 (1997), pp. 209–218.
- [16] R. J. Elliott, V. Krishnamurthy e J. Sass. «Moment based regression algorithm for drift and volatility estimation in continuous time Markov switching model». In: *Econometrics Journal* 11 (2008), pp. 244–270.
- [17] F. Fagnola. *Modelli Dinamici Stocastici*. 2010.
- [18] X. Guo. «Information and option pricing». In: *Quantitative Finance* 1 (2001), pp. 38–44.
- [19] M. Hahn, S. Frühwirth-Schnatter e J. Sass. «Markov chain Monte Carlo methods for parameter estimation in multidimensional continuous time Markov switching models». 2007.
- [20] M. Hahn e J. Sass. «Parameter Estimation in Continuous Time Markov Switching Model: A Semi-Continuous Markov Chain Monte Carlo Approach». In: *Bayesian Analysis* 1 (2009), pp. 63–84.
- [21] J.M. Harrison e S.R. Pliska. «Martingales and Stochastic integrals in the theory of continuous trading». In: *Stochastic Processes and their Applications* 3.11 (1981), pp. 215–260.
- [22] O. Häggström. *Finite Markov Chains and Algorithmic Applications*. Cambridge University press, 2002.
- [23] Y. Huang, P.A. Forsyth e Labahn G. «Methods for pricing american options under regime switching». In: *SIAM Journal on Scientific Computing* 33:5 (2011), pp. 2144–2168.
- [24] J.S. Kennedy. «Hedging contingent claims in markets with jumps». Tesi di dott. School of Computer Science, University of Waterloo, 2007.
- [25] D. Lamberton e B. Lapeyre. *Introduction au Calcul Stochastique Appliqué à la Finance*. Paris: Ellipses.
- [26] J. C. Liechty e G. O. Roberts. «Markov chain Monte Carlo methods for switching diffusion models». In: *Biometrika* 88 (2001), pp. 299–315.
- [27] R. H. Liu, Q. Zhang e G. Yin. «Option pricing in a regime-switching model using fast Fourier transform». In: *Journal of Applied Mathematics and Stochastic Analysis* 2006 (2006), pp. 1–22.

- [28] Francis Longstaff e Eduardo Schwartz. «Valuing American Options by Simulation: A Simple Least-Squares Approach». In: *The Review of Financial Studies* 14.1 (2001).
- [29] B.B. Mandelbrot. «The variation of certain speculative prices». In: *Journal of Business* XXXVI (1963), pp. 392–417.
- [30] G.H. Mayer. *Pricing Options and Bonds with the Method of Lines*. 2013. URL: <http://people.math.gatech.edu/~meyer/MOL-notes/>.
- [31] G.H. Mayer. «The evaluation of American Options with the Method of Lines». 1994.
- [32] G.H. Mayer. «The numerical valuation of options with underlying jumps». In: *Acta Math. Univ. Comenianae* LXVII.1 (1998), pp. 69–82.
- [33] N. Metropolis e S. Ulam. «The Monte Carlo method». In: *Journal of the American Statistical Association* 44 (1949), pp. 335–341.
- [34] H. Pham. «Optimal stopping, free boundary and American option in a jump-diffusion model». In: *Appl. Math. Optim.* 35 (1997), pp. 145–164.
- [35] H. Pham. «Optimal stopping of controlled jump diffusion processes: A viscosity solution approach». In: *Journal of Mathematical Systems* 8 (1998), pp. 1–27.
- [36] A. Quarteroni, R. Sacco e R. Salieri. *Matematica Numerica*. Springer, 2008.
- [37] E. N. Sarmin e L.A. Chudov. «On the stability of the numerical integration of systems of ordinary differential equations arising in the use of the straight line method». In: *USSR Computational Mathematics and Mathematical Physics* 3(6) (1963), 1537–1543.
- [38] Rüdiger U. Seydel. *Tools for Computational Finance*. Springer-Verlag Berlin Heideberg, 2009.
- [39] C. Whang e R. Chen. «Option Pricing for Merton Jump Diffusion Model with Regime-Switching using FFT». In: (2009).
- [40] P. Wilmott, S. Howison e J. Dewynne. *The mathematics of financial derivatives: a student introductions*. Cambridge: Cambridge University press.
- [41] D.D. Yao, Q. Zhang e X.Y. Zhou. *Stachastic Processes, Optimization, and Control Theory Applications in Financial Engineering, Queuing Networks, and Manufacturing Systems*. New York: Springer, 2006, pp. 281–300.

Ringraziamenti

A conclusione di questo lavoro vorrei ringraziare innanzitutto il Professor Marazzina che mi ha dato la possibilità di realizzarlo, fornendomi idee, spunti e suggerendomi modifiche al fine di renderlo migliore.

Ringrazio nonna Teresa perché, forse, le sue innumerevoli preghiere prima di ogni esame non sono rimaste totalmente inascoltate e so che sarà felice di non dovermi più vedere studiare così tanto.

Grazie anche a tutti i miei zii, vicini e soprattutto lontani, e ai miei cugini per essersi sempre interessati ai miei studi, per avermi fatto sentire il loro appoggio ed il loro incoraggiamento in vari momenti, anche difficili, di questi anni di studio.

Un pensiero particolare va a tutti coloro che sono stati i miei compagni di corso, soprattutto Ciocca e Davide, che in questi anni trascorsi al Politecnico hanno condiviso con me le ansie pre-esame, le ore di studio ed innumerevoli gustosi caffè dei distributori automatici. Un grazie particolare merita Davide che ha saputo rendere fin troppo goliardiche anche le materie più noiose, motivare le intuizioni più assurde, creare nuovi teoremi, sciogliere le situazioni più intricate ed infondermi positività.

Non minore importanza hanno avuto tutti i miei amici, nessuno escluso, per avermi distratto nei giorni di maggior stress, per essermi stati vicino e per avermi esortato a tener duro quando la fatica iniziava a farsi sentire.

Voglio ricordare, più di tutti, chi purtroppo non può essere presente in questo momento per me importante: papà e nonna Angela che con ottimismo, tenacia e positività sono stati sempre capaci di infondermi serenità anche nei giorni più bui, diventando un modello per chi vorrei essere. Il grande entusiasmo che hanno sempre manifestato per il cammino che avevo deciso di intraprendere e la fiducia che riponevano nei miei confronti è stato uno sprone per portare a termine questo percorso.

Grazie a mamma per aver tenuto duro in questi difficili anni, che hanno messo tutti noi duramente alla prova, per non aver mollato, per aver trovato la forza di andare avanti e per avermi affiancato nel corso degli studi che, giunto a conclusione, spero la renda fiera di me.

Un pensiero anche per mia sorella Elisa perché, nonostante il carattere totalmente opposto al mio, si è sempre sforzata di comprendere i miei musi lunghi, le mie lamentele, le mie paranoie ed i periodi di nervosismo.

Da ultimo, Aurora, detta anche Miss Manicardi. Sei stata la persona che ha saputo ridarmi gioia e felicità, che mi mette di buon umore ogni volta che sorride, con cui ho trascorso questi anni di studio intenso e con la quale oggi, in allegria, li concludo. Condividere ogni singola cosa con te, al Politecnico e non, è stata, sicuramente, la cosa più bella che mi potesse capitare: ti amo ingegnere!

Matteo