UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Quebra de resultado por fator de risco em uma carteira de ativos

Carlos Roberto Ferrari Junior

Dissertação de Mestrado do Programa de Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria (MECAI)

ICMC USP
SÃO CARLOS

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: ______________________

Carlos Roberto Ferrari Junior

Quebra de resultado por fator de risco em uma carteira de ativos

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em CiÃłncias – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria. *VERSÃO REVISADA*

Área de Concentração: Matemática, Estatística e Computação

Orientador: Prof. Dr. Antonio Castelo Filho

USP – São Carlos
Maio de 2018

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

F375q

Ferrari Jr, Carlos Roberto
Quebra de resultado por fator de risco / Carlos Roberto Ferrari Jr; orientador Antonio Castelo Filho. -- São Carlos, 2018.
82 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2018.

1. Ativos do mercado financeiro brasileiro. 2. Resultado por fator de risco. 3. Gerindo um portfolio. I. Castelo Filho, Antonio, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Carlos Roberto Ferrari Junior

Profit and losses broken by risk factor in a portfolio

Master dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master – Professional Masters in Mathematics, Statistics and Computing Applied to Industry. *FINAL VERSION*

Concentration Area: Mathematics, Statistics and Computing

Advisor: Prof. Dr. Antonio Castelo Filho

USP – São Carlos
May 2018

RESUMO

FERRARI, C. R. J. **Quebra de resultado por fator de risco em uma carteira de ativos**. 2018. 82 p. Dissertação (Mestrado – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

O mercado financeiro possui diversos tipos de operadores buscando lucros com diferentes focos de atuação. Temos, por exemplo, especuladores tentando prever futuras variações nos ativos, ou bancos provendo liquidez para as demandas das empresas. De qualquer maneira, cada ativo financeiro normalmente contém diversos riscos de mercado na sua precificação. Além disso, muitos desses riscos são, na verdade, indesejáveis, fazendo com que seja necessário buscar proteção contra eles, utilizando-se de outros ativos. Assim, numa carteira com diversos produtos, é vital para sua administração a divisão por fator específico de risco de mercado.

O objetivo deste trabalho consiste em resolver uma demanda real de uma gestora de recursos onde o autor atuou: desenvolver uma ferramenta capaz de calcular os ganhos ou perdas de uma carteira de ativos, detalhando o resultado por tipo de risco de mercado. Para isso foi desenvolvido um sistema no formato de *add-in* de Excel no qual é possível definir as diversas curvas de mercado para diferentes datas, assim como calcular o valor presente dos ativos. Com isso, é também possível determinar o valor total do *portfolio* e a variação de um dia para o outro, conhecido como *PNL* diário.

Nesta dissertação serão mostrados detalhes da implementação do sistema e será proposta uma metodologia para calcular o *PNL* por cada risco de mercado. O trabalho também apresentará o racional por trás da escolha de cada método, e também discutirá como eles poderiam ser refinados.

Palavras-chave: Ativos do mercado financeiro brasileiro, Resultado por fator de risco, Gerindo um *portfolio*.

ABSTRACT

FERRARI, C. R. J. **Profit and losses broken by risk factor in a portfolio**. 2018. 82 p. Dissertação (Mestrado – Mestrado Profissional em Matemática, Estatística e Computação Aplicadas à Indústria) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

The Brazilian financial market has several types of players, with dfferent ways of acting to pursue profits. For instance, there are the speculators that try to foresee market moves, and banks that provide liquidity for the demands of corporate players. But every asset has several market risks at their pricing, and normally many of them are not wanted. Therefore, players seek to trade other assets to hedge themselves from theses undesirable risks. Therefore, it is vital to have the breakdown of every market risk when managing a portfolio.

The main purpose of this work is so solve a real demand for an asset management company for which this author used to work: the development of a tool capable of calculating the profit and losses of a portfolio, breaking down the result by each risk factor. An Excel add-in has been developed where it is possible to set several market curves for different dates, as to calculate the present value of the assets. With all that, it is possible to obtain the daily PNL.

In this thesis we will discuss system's implementation details, and a methodology will be presented to calculate the PNL by risk factor. It will also include the rational behind the decision of each chosen method, as discussions of how they could be developed.

Keywords: Assets of the Brazilian financial market, *PNL* explanation by risk factor, Managing a portfolio.

LISTA DE ILUSTRAÇÕES

Figura 1 – Resultados calculados com SIRI em Excel 48

LISTA DE TABELAS

Tabela 1 – Curva de CDI para D0 - 12 de setembro de 2017 21
Tabela 2 – Curva de CDI para D-1 - 11 de setembro de 2017 22
Tabela 3 – Curva de juros em Dólares OIS D0 - 12 de setembro de 2017 23
Tabela 4 – Curva de juros em Dólares OIS D-1 - 11 de setembro de 2017 24
Tabela 5 – Curva de cupom cambial para D0 - 12 de setembro de 2017 25
Tabela 6 – Curva de cupom cambial para D-1 - 11 de setembro de 2017 26
Tabela 7 – Curva de prêmio *onoff* para D0 - 12 de setembro de 2017 27
Tabela 8 – Curva de prêmio *onoff* para D-1 - 11 de setembro de 2017 28
Tabela 9 – Superfície de volatilidade para D0 - 12 de setembro de 2017 30
Tabela 10 – Superfície de volatilidade para D-1 - 11 de setembro de 2017 31

SUMÁRIO

1	**INTRODUÇÃO**	**15**
1.1	**Hedgers**	**15**
1.2	**Especuladores**	**15**
1.3	**“Market Makers”**	**15**
1.4	**Objetivos gerais**	**16**
1.5	**Organização**	**17**
2	**CONSTRUÇÃO DE CURVAS DE MERCADO**	**19**
2.1	**Curva de juros em moeda local (CDI)**	**19**
2.2	**Curva de juros em Dólares (OIS)**	**21**
2.3	**Curva de juros de dólares no Brasil (cupom cambial)**	**23**
2.4	**Curva de prêmio de convertibilidade (onoff)**	**25**
2.5	**Superfície de volatilidade**	**27**
2.5.1	***Moneyness***	***29***
2.5.2	***Call-Put Parity***	***29***
2.5.3	***Delta***	***29***
3	**PRECIFICAÇÃO DOS ATIVOS**	**33**
3.1	**Contrato Futuro de DI1 da BMF**	**33**
3.2	**Contrato Futuro de DOL da BMF**	**34**
3.3	**Contrato Futuro de DDI da BMF**	**35**
3.4	**NDF Offshore**	**36**
3.5	**NDO Offshore**	**37**
3.5.1	***Modelo de Black and Scholes***	***37***
3.5.2	***Paridade call-put***	***39***
3.5.3	***Black 1976***	***40***
3.6	***Swap* Pré vs CDI Offshore**	**41**
4	**DESENVOLVIMENTO**	**43**
4.1	**PNL por *Theta***	**44**
4.2	**PNL por *Delta***	**44**
4.3	**PNL por *Rho* e *Vega***	**44**
4.4	**PNL por *Cross Curves***	**45**
4.5	**Resíduo**	**46**

5 RESULTADOS 47
5.1 *Siri* 47
5.2 *Portfolio* Teórico 48
5.3 Resultados 48

6 CONCLUSÃO 51

REFERÊNCIAS 53

APÊNDICE A CLASSE CURVES 55

APÊNDICE B *SPLINE* CÚBICO 65

APÊNDICE C CLASSE `LVOLSURFACE` 69

APÊNDICE D CLASSE `MARKETVOLSURFACE` 73

APÊNDICE E CLASSE BLACK76 77

CAPÍTULO

1

INTRODUÇÃO

O mercado de capitais no Brasil é bem desenvolvido, e diferentes tipos de participantes realizam operações de acordo com seus interesses. Podemos dividí-los em três grandes grupos de acordo com o que buscam ao operar no mercado financeiro.

1.1 Hedgers

Vem do termo em inglês “hedge”, que significa zerar um risco de mercado evitando variações de resultado dada uma flutuação no preço do ativo objeto. Esse tipo de participante faz operações para se proteger das variações de preço. Um exemplo comum é o de empresas exportadoras e importadoras que normalmente têm um descasamento na moeda dos seus custos e de suas receitas. Assim, elas buscam o mercado para travar seus ganhos e não estarem sujeitas à variação do câmbio, evitando que esse risco possa atrapalhar a lucratividade da empresa.

1.2 Especuladores

Esse tipo de participante tenta usar o mercado financeiro para obter lucros com a variação dos preços dos ativos. Se baseando em análises fundamentalistas ou técnicas, eles buscam se posicionar de maneira a antecipar movimentos de mercado, correndo risco de perdas financeiras, e com isso, tentam gerar ganhos. O melhor exemplo são os fundos de investimento, que buscam maximizar lucratividade de seus cotistas, mas bancos e mesmo pessoas físicas com alto patrimônio também costumam fazer operações com esses fins.

1.3 “Market Makers”

A tradução do inglês fazedores de mercado funciona bem na descrição desse tipo de participante. Normalmente a operação realizada pelos outros dois participantes já mencionados tem descasamentos de características específicas, abrindo espaço para que alguém tenha que preencher esta lacuna. Normalmente esse trabalho é feito pelos bancos, que por exemplo, operam dólares contra especuladores com liquidação em uma data diferente da requerida por um empresa buscando hedge, fazendo com que seja possível obter ganhos incorrendo pequenos riscos de mercado.

1.4 Objetivos gerais

Porém, na prática, os ativos financeiros possuem mais de um risco embutido em sua precificação. Assim, quando esses participantes buscam o mercado para executar suas operações, eles se deparam com a necessidade de compor um *portfolio* com vários produtos a fim de que os riscos indesejáveis sejam mitigados.

Esse tipo de operação pode complicanr bastante a atuação do profissional (princiapalmente inexperientes). A maior parte da bibliografia sobre o assunto foca apenas na parte teórica, sem abordar problemas práticos. A falta de trabalhos na área, assim como a de profissionais com a experiência necessária, faz com que seja comum encontrar empresas de médio e grande porte sem um sistema que faça os cálculos específicos essenciais.

Com isso, surgiu a necessidade do desenvolvimento de um produto que faça o cálculo do resultado quebrado por fator de risco. Esse trabalho tem como objetivo propor uma metodologia para realizar estes cálculos, e explicar, didaticamente, a lógica envolvida. Em paralelo, ele visa introduzir discussões mais profundas sobre os temas, e apontar referências para que essas informações possam ser obtidas. Com isso, outras empresas poderão com maior facilidade criar seus próprios sistemas a partir do método que aqui será proposto.

Fazendo com que essa dissertação auxilie todos os tipos de participantes, posto que estes precisam de uma ferramenta que os ajude a gerir um *portofolio* com diversos ativos para atingir seus diferentes objetivos. O vasto conhecimento sobre produtos operados facilita, proporcionalmente, a capacidade do profissional de otimizar seus resultados. Aqui buscamos servir de base para iniciantes que procuram entender a dinâmica do funcionamento de uma carteira composta por diversos produtos.

O programa aqui discutido foi desenvolvido em uma gestora de recursos que funcionava como especulador, onde se buscava maximizar o retorno financeiro para os cotistas dos fundos, através de previsões de como o mercado financeiro iria reagir no futuro. Foi desenvolvido um *add-in* de Excel onde é possível criar curvas de mercado e precificar ativos com elas. Com essa ferramenta, foi possível refinar as posições do fundo, aumentando a gama de riscos de mercado operados para se obter ganhos.

1.5 Organização

Inicialmente, esta monografia irá apresentar modelos para a construção das curvas de mercado. Nessa etapa serão discutidas as particularidades de cada uma, assim como o motivo pelo qual cada modelo foi escolhido. As escolhas serão justificadas e alternativas também serão mencionadas para que futuramente seja possível um aperfeiçoamento do sistema, ou, simplesmente, uma calibração diferente que agrade mais ao usuário.

Após a construção das curvas de mercado, os modelos utilizados para precificar os ativos, que utilizarão as curvas já definidas serão discutidas. Também serão discutidos outros modelos mais complexos para que o desenvolvedor possa escolher um método outro que o aqui utilizado, mas que permita obter facilmente resultados satisfatórios. A dissertação também busca se sevir de uma didática simples para implementar mais produtos e curvas que não estejam modelados aqui, utilizando a mesma metodologia.

Uma vez que os produtos estejam devidamente precificados, o desenvolvimento deste trabalho irá expor os cálculos que devem ser utilizados para que as respostas condizentes sejam obtidas. Em seguida, será criado um *portfolio* teórico com os produtos descritos anteriormente que serão utilizados para a realização dos cálculos propostos. Para concluir, será feita uma análise da qualidade dos resultados obtidos de trabalhos futuros que possam evoluir o sistema serão propostos.

CAPÍTULO

2

CONSTRUÇÃO DE CURVAS DE MERCADO

Antes de precificarmos qualquer ativo, é importante construir as curvas de mercado que serão utilizadas. Nesse processo, a definição dos parâmetros impactará diretamente o valor presente dos produtos, assim como a variação de preço de um dia para o outro.

O primeiro ponto a ser definido será quais os vértices utilizados para cada curva. A partir da fonte desses dados e da dinâmica própria do índice, será determinado o método de interpolação a ser utilizado. Vale ressaltar que não existe resposta certa ou errada neste passo, porém é importante buscar um método que tente se aproximar ao máximo de como a curva se comporta na prática com o passar do tempo.

No mercado global encontramos várias formas distintas de expressar taxas de juros. No Brasil a capitalização é diária - levando em conta dias úteis - enquanto nos EUA trabalha-se mais comumente com juros simples pensando em dias corridos, com pagamentos trimestrais. No entanto, os fatores de capitalização oriundo das taxas é o que será de fato utilizado para precificar os ativos, por isso é importante que estes sejam calculados de acordo com a convenção da taxa utilizadas pela fonte de dados. Uma maneira de evitar problemas é encontrar uma fonte de dados onde os fatores de capitalização são parâmetros conhecidos.

2.1 Curva de juros em moeda local (CDI)

O Comitê de Política Monetária, conhecido como COPOM, define a Meta Selic, que é a taxa básica de juros da economia, e o Banco Central do Brasil dispõe de diversos instrumentos para que o mercado pratique taxas próximas às definidas pelo COPOM.

Uma maneira de balisar a taxa de juros do mercado é quando o BC faz operações de go around, onde ele toma o caixa excedente dos bancos, ou empresta dinheiro se necessário no overnight. Essas operações ajudam a formar a taxa Selic efetiva.

Como o volume negociado nas operações que formam o Selic é muito superior ao que forma o CDI, a expectativa da taxa Selic é utilizada para determinação do CDI futuro. Atualmente o BC empresta dinheiro 10bps acima da taxa Selic aos bancos, e toma 10bps abaixo.

Como o mercado é credor em caixa, há anos a Selic é 10bps menor que a Meta Selic. Já o CDI é bem correlacionado com a Selic, porém como poucos negócios são registrados diariamente, o cálculo tem caído no *fallback* da CETIP que, com os parâmetros atuais, acaba sendo 1bp abaixo da Selic. Assim, o CDI tem sido 11bps menor que a Meta Selic.

O CDI over é uma taxa utilizada na maioria dos contratos de derivativos de juros no Brasil e é divulgada pela Central de Custódia e de Liquidação Financeira de Títulos (CETIP), sendo calculada com base taxa média dos CDI's (Certificado de Depósito Interbancário) de um dia praticada pelos bancos. O CDI representa a taxa média pela qual os bancos trocaram depósitos com vencimento no dia útil seguinte.

O COPOM se reúne 8 vezes por ano, em média a cada 45 dias, e as datas exatas destas reuniões são divulgadas com mais de um ano de antecedência. No período entre as reuniões, a Meta Selic é mantida constante, e toda e qualquer decidida nesses encontros passa a valer a partir do dia posterior à tomada de decisão. Isso gera uma dinâmica onde a taxa permanece constante entre as reuniões, dando salto quando alterada. fazendo com que, o método de interpolação utilizado seja flat forward para a curva de CDI.

A fonte de dados mais comumente utilizada é a oriunda dos contratos de DI1 na BMF. Normalmente os vértices utilizados para a construção da curva são as datas de COPOM conhecidas e, posteriormente, o vencimento dos contratos com liquidez relevante na BMF. Porém, como os contratos de DI1 vencem sempre no primeiro dia do mês, é necessário utilizar métodos de aproximação para construir a curva nos pontos das reuniões do COPOM. Um método é encontrado (CARREIRA; JR, 2016).

Assim, neste trabalho, a curva de CDI será construída apenas pelos contratos da BMF e será utilizada a interpolação flat forward entre eles. Esse método funciona de maneira satisfatória quando ao longo da curva, encontramos variações suaves da Selic. Porém, em períodos de ciclos agressivos - tanto de alta como de queda de juros -, onde, por exemplo, a curva precifica movimentos de 100bps por reunião, a interpolação sem utilizar as datas exatas das reuniões acaba gerando erros significativos em prazos mais curtos.

Já na parte mais longa da curva, onde as datas das reuniões nem são definidas, pode-se discutir a utilização de outro método de interpolação como spline cúbico. Vale ressaltar que a premissa que utilizamos para escolher o flat forward foi o de que a taxa de juros dá saltos discretos: porém - quando as datas exatas destas mudanças não são sabidas, também faria sentido usar um método que gera variações mais suaves. No entanto, além dos problemas numéricos usuais da implementação do spline cúbico, ele também gera um efeito de que todos os pontos são utilizados para interpolar quaisquer prazos.

Portanto, uma variação abrupta na parte curta da curva, causa impacto na interpolação da parte longa. Esse efeito não seria necessariamente incorreto, porém é uma diferença significativa entre os dois métodos que deve ser levada em conta pelo operador.

$$\text{Fator}_{CDI} = (1 + \text{DI}_{fut})^{\frac{\text{du}}{252}} \tag{2.1}$$

onde:

DI_{fut} = valor da taxa do DI futuro para o vencimento;

du = dias úteis no calendário brasileiro entre a partida e o vencimento.

Tabela 1 – Curva de CDI para D0 - 12 de setembro de 2017

Prazo	CDI	DU	Fator
2-Jan-18	7.64%	75	1.022159
1-Feb-18	7.51%	97	1.028273
1-Mar-18	7.43%	115	1.033263
2-Apr-18	7.38%	136	1.039155
2-May-18	7.36%	157	1.045226
1-Jun-18	7.36%	178	1.051409
2-Jul-18	7.37%	199	1.057743
1-Oct-18	7.47%	263	1.078131
2-Jan-19	7.64%	325	1.099549
1-Apr-19	7.80%	386	1.121953
1-Jul-19	8.00%	448	1.146615
1-Oct-19	8.22%	514	1.174729
2-Jan-20	8.39%	578	1.202927
1-Jul-20	8.74%	701	1.262389
4-Jan-21	9.01%	829	1.328001
1-Jul-21	9.25%	952	1.397030
3-Jan-22	9.41%	1,080	1.470316
2-Jan-23	9.63%	1,331	1.625373
2-Jan-25	9.94%	1,834	1.993385
4-Jan-27	10.17%	2,337	2.454591

Fonte – BMF

2.2 Curva de juros em Dólares (OIS)

Após definir a curva de juros da moeda local, é necessário fazer o mesmo para a curva em dólares, ou USD. Até a crise financeira de 2008, os bancos eram vistos como participantes que não apresentavem risco de crédito para suas contrapartes. E com isso a **LIBOR** – taxa de juros média interbancária utilizada por um grande número de bancos no mercado monetário londrino para empréstimos mútuos sem garantia realizados em dólares americanos – era utilizada para a precificação de todos os ativos no exterior.

Tabela 2 – Curva de CDI para D-1 - 11 de setembro de 2017

Prazo	CDI	DU	Fator
2-Jan-18	7.74%	76	1.022742
1-Feb-18	7.47%	98	1.028410
1-Mar-18	7.47%	116	1.033729
2-Apr-18	7.51%	137	1.040175
2-May-18	7.50%	158	1.046398
1-Jun-18	7.49%	179	1.052658
2-Jul-18	7.51%	200	1.059166
1-Oct-18	7.52%	264	1.078870
2-Jan-19	7.70%	326	1.100740
1-Apr-19	7.83%	387	1.122803
1-Jul-19	8.07%	449	1.148254
1-Oct-19	8.20%	515	1.174852
2-Jan-20	8.39%	579	1.203418
1-Jul-20	8.74%	702	1.262776
4-Jan-21	9.03%	830	1.329241
1-Jul-21	9.23%	953	1.396310
3-Jan-22	9.39%	1,081	1.469493
2-Jan-23	9.61%	1,332	1.624206
2-Jan-25	9.92%	1,835	1.990867
4-Jan-27	10.13%	2,338	2.448390

Fonte – BMF

Ela é disponibilizada em 7 períodos de duração diferentes: de overnight a 12 meses. A taxa mais utilizada era a Libor de 3 meses, com juros simples calculados por dias corridos fazendo com que o mercado pudesse ser operado por derivativos na bolsa americana CME, os chamados *Eurodollar*, ou *swaps* entre duas contrapartes. Por conta de características próprias de cada ativo e suas dinâmicas, que são melhores explicadas em (CARREIRA; JR, 2016), usualmente a curva é construída ate três meses pela Libor 3M, depois até dois anos pelos contratos de Eurodollar e após isso, as taxas de swap.

No entanto, depois da crise de 2008, onde a solvência dos bancos não era mais dada como premissa, as contrapartes começaram a depositar margens de garantia conforme sofriam perdas em ativos que ainda não tiveram fluxo de caixa. Assim, o mercado migrou da Libor para a curva de juros da margem depositada, fazendo uso muitas vezes de contratos que preveem o depósito de uma margem em diferentes moedas, como Dólar, Euro ou Libra Esterlina. Inclusive, as contrapartes costumam trocar a moeda de garantia de acordo com a variação do custo para que lhes seja mais conveniente. Para simplificar, aqui será discutido apenas a taxa de Dólares OIS, que é a expectativa futura do Fed Funds, a taxa de remuneração de reservas no Banco Central Americano.

O OIS tem uma estrutura diferente da Libor, onde ao invés de ter uma taxa fixa por três meses, varia diariamente. Como a Libor é a taxa operada entre bancos, normalmente ela é mais alta do que o Fed Funds, por conter algum risco de crédito embutido.

Neste trabalho utilizaremos a taxa de OIS para construir a curva de juros em moeda estrangeira. Por conta das diferentes notações de cada tipo de taxa de juros, é mais fácil e seguro utilizar fontes de dados que já contenham o fator de capitalização para cada prazo, já que está é a informação que será de fato utilizada na precificação dos ativos.

O método de interpolação utilizado será o spline cubico, porém todas as considerações feitas para o CDI são válidas aqui também.

$$\mathrm{Fator}_{\mathrm{OIS}} = 1 + \mathrm{OIS}\frac{\mathrm{dc}}{360} \tag{2.2}$$

onde:

OIS = valor da taxa OIS para o vencimento;

dc = dias corridos entre a partida e o vencimento.

Tabela 3 – Curva de juros em Dólares OIS D0 - 12 de setembro de 2017

Tenor	Prazo	Fator	DC	OIS
1M	13-Oct-17	1.0009914	31	1.151%
2M	13-Nov-17	1.0019907	62	1.156%
3M	13-Dec-17	1.0029570	92	1.157%
4M	12-Jan-18	1.0039967	122	1.179%
6M	13-Mar-18	1.0060814	182	1.203%
9M	13-Jun-18	1.0094612	274	1.243%
12M	13-Sep-18	1.0128551	366	1.264%
18M	13-Mar-19	1.0200534	547	1.320%
2Y	13-Sep-19	1.0276757	731	1.363%
3Y	11-Sep-20	1.0440070	1,095	1.447%
4Y	13-Sep-21	1.0620278	1,462	1.527%
5Y	13-Sep-22	1.0815165	1,827	1.606%
7Y	13-Sep-24	1.1262628	2,558	1.777%
10Y	13-Sep-27	1.2012512	3,653	1.983%

Fonte – Bloomberg

2.3 Curva de juros de dólares no Brasil (cupom cambial)

Moedas de países desenvolvidos e com mercado de capitais mais sólidos são normalmente *deliverable*, ou seja, existe, de fato, a troca das duas moedas nas negociações. Por outro lado, no Brasil, assim como em muitos outros países em desenvolvimento, os contratos de derivativo são *non deliverable*, ou seja, ao término da operação, calcula-se o ganho ou perda financeiro que é pago apenas em uma moeda, no país onde a operação é registrada. Resumindo, um derivativo de câmbio no Brasil é liquidado em Reais, enquanto nos EUA ele é pago em Dólares.

Tabela 4 – Curva de juros em Dólares OIS D-1 - 11 de setembro de 2017

Tenor	Prazo	Fator	DC	OIS
1M	11-Oct-17	1.0009597	30	1.152%
2M	10-Nov-17	1.0019267	60	1.156%
3M	12-Dec-17	1.0029549	92	1.156%
4M	12-Jan-18	1.0040125	123	1.174%
6M	12-Mar-18	1.0060619	182	1.199%
9M	12-Jun-18	1.0093522	274	1.229%
12M	12-Sep-18	1.0127574	366	1.255%
18M	12-Mar-19	1.0198563	547	1.307%
2Y	12-Sep-19	1.0272700	731	1.343%
3Y	11-Sep-20	1.0431534	1,096	1.417%
4Y	10-Sep-21	1.0603851	1,460	1.489%
5Y	12-Sep-22	1.0791844	1,827	1.560%
7Y	12-Sep-24	1.1216700	2,558	1.712%
10Y	10-Sep-27	1.1956282	3,651	1.929%

Fonte – Bloomberg

Isso faz com que não se possa utilizar a curva de juros americana para precificar ativos em Dólares no Brasil, gerando a curva de cupom cambial. Portanto, uma maneira de se obter essa taxa, é descontar os valores futuros da paridade USDBRL pelo CDI, assim isolando a componente de juros em Dólar. Na literatura, é comumente encontrado o termo cupom limpo e cupom sujo, onde o primeiro é calculado através da taxa *spot* e o outro da *ptax*. A tese de (FRANçA, 2010) traz mais detalhes sobre isso junto com outras tecnicalidades dos contratos da BMF.

A dificuldade da construção da curva de cupom cambial é a convenção de datas. Onde o dólar spot é operado para liquidação em dois dias úteis considerando feriados no Brasil e em Nova Iorque. Os derivativos offshore também liquidam dois dias úteis após a taxa de *fixing*, no entanto, ativos onshore, como os contratos da BMF, pagam um dia útil após o fixing, e apenas levando em conta os feriados locais. Assim, ao utilizar dados da BMF, é necessário ajustar os dias de liquidação para calcular nossa curva de cupom cambial que chamaremos D+2.

(CARREIRA; JR, 2016) explica detalhadamente o porquê de utilizar a costrução dessa maneira, visando sempre evitar possíveis arbitragens nos preços. Neste trabalho, utilizaremos como fonte de dados os próprios valores futuros dos contratos do USDBRL, para posteriormente descontarmos o valor do CDI e obter a taxa do cupom cambial. A interpolação mais comumente usada é o flat forward, porém ela não tem o mesmo apelo teórico do CDI, então pode-se facilmente trocar para spline cubico como na curva americana. Nesta dissertação utilizaremos a interpolação flat forward e a convenção de capitalização anual por dias corridos, lembrando que BMF utiliza capitalização simples, no entanto, é importante frisar que contanto que o fator de capitalização seja calculado de acordo com a convenção da taxa, os cálculos são igualmente corretos.

$$\text{Fwd}_{usdbrl} = \text{Spot}_{usdbrl} \frac{\text{Fator}_{\text{CDI}}}{\text{Fator}_{\text{CC}}} \tag{2.3}$$

$$\text{Fator}_{\text{CC}} = 1 + \text{CC}\frac{\text{dc}}{360} \tag{2.4}$$

onde:

CC = valor da taxa do cupom cambial para o vencimento;

dc = dias corridos entre a partida e o vencimento.

Tabela 5 – Curva de cupom cambial para D0 - 12 de setembro de 2017

Prazo	Cupom cambial	DC	Fator
2-Oct-17	3.72%	20	1.002066
1-Nov-17	3.07%	50	1.004264
1-Dec-17	2.87%	80	1.006383
2-Jan-18	2.75%	112	1.008542
1-Feb-18	2.67%	142	1.010517
1-Mar-18	2.62%	170	1.012383
1-Apr-18	2.59%	201	1.014440
2-Jul-18	2.61%	293	1.021266
1-Oct-18	2.62%	384	1.027966
2-Jan-19	2.73%	477	1.036222
1-Apr-19	2.79%	566	1.043787
1-Jul-19	2.84%	657	1.051828
1-Oct-19	2.86%	749	1.059434
2-Jan-20	2.93%	842	1.068525
1-Apr-20	2.98%	932	1.077238
1-Jul-20	3.04%	1,023	1.086508
1-Oct-20	3.07%	1,115	1.095124
4-Jan-21	3.15%	1,210	1.105927
1-Jul-21	3.23%	1,388	1.124705
2-Jan-22	3.30%	1,573	1.144390
2-Jan-23	3.41%	1,938	1.183440
2-Jan-24	3.59%	2,303	1.229575
2-Jan-25	3.79%	2,669	1.281103
2-Jan-26	3.98%	3,034	1.335560
4-Jan-27	4.15%	3,401	1.391748

Fonte – BMF

2.4 Curva de prêmio de convertibilidade (onoff)

Um derivativo de câmbio pode ser operado tanto no Brasil como no exterior, ainda que a especificação do contrato seja a mesma, a lei que rege ativos financeiros é bem diferente em casos

Tabela 6 – Curva de cupom cambial para D-1 - 11 de setembro de 2017

Prazo	Cupom cambial	DC	Fator
2-Oct-17	3.53%	21	1.002058
1-Nov-17	2.97%	51	1.004211
1-Dec-17	2.79%	81	1.006282
2-Jan-18	2.72%	113	1.008534
1-Feb-18	2.59%	143	1.010274
1-Mar-18	2.56%	171	1.012173
1-Apr-18	2.56%	202	1.014346
2-Jul-18	2.57%	294	1.020985
1-Oct-18	2.58%	385	1.027624
2-Jan-19	2.69%	478	1.035681
1-Apr-19	2.75%	567	1.043277
1-Jul-19	2.81%	658	1.051298
1-Oct-19	2.84%	750	1.059190
2-Jan-20	2.91%	843	1.068107
1-Apr-20	2.96%	933	1.076759
1-Jul-20	3.02%	1,024	1.085820
1-Oct-20	3.06%	1,116	1.094737
4-Jan-21	3.12%	1,211	1.105094
1-Jul-21	3.21%	1,389	1.123872
2-Jan-22	3.29%	1,574	1.143874
2-Jan-23	3.48%	1,939	1.187393
2-Jan-24	3.72%	2,304	1.238392
2-Jan-25	3.90%	2,670	1.288888
2-Jan-26	4.04%	3,035	1.340888
4-Jan-27	4.21%	3,402	1.397528

Fonte – BMF

extremos de condições de mercado. O exemplo mais comum desses eventos é a manipulação da taxa de câmbio pelo governo do país.

A Argentina é um arquétipo atual, onde até pouco tempo atrás, vender dólares no mercado oficial era pior que no paralelo, posto que o governo maquiava a taxa de câmbio do país. Isso tem um efeito relevante para o mercado financeiro, já que faz com que contratos idênticos liquidem por valores distintos em função de estarem sob a lei argentina ou internacional.

Mesmo no Brasil, onde não existe diferença na taxa divulgada pelo Banco Central e a operada pelo mercado financeiro, os preços praticados em contratos locais e estrangeiros são distintos. Isso é conhecido como prêmio de convertibilidade, mais comumente chamado de prêmio onoff, já que é a diferença entre o mercado *onshore* e *offshore*.

Pela definição, não faria sentido o prêmio onoff ser negativo, já que não existe precedente de um país manipular sua taxa de câmbio deixando-a menos valorizada do que a praticada de fato pelo mercado. Porém, por conta do descasamento de fluxos e custos de operação, não é incomum ver valores negativos nessa taxa para ativos brasileiros. O prêmio *onoff* será expresso em percentual ao ano e calculado em dias corridos com capitalização anual. Ressaltemos que a notação não é o mais importante, mas sim o fator de capitalização da taxa que é de fato utilizado.

A fonte de dados, normalmente, não é tão confiável já que não existe uma divulgação oficial de preços. Normalmente bancos são os participantes que conseguem operar mais facilmente no mercado local e externo, portanto são quem melhor enxergam o prêmio onoff.

$$\mathrm{Fwd}_{\mathrm{offshore}} = \mathrm{Fwd}_{\mathrm{onshore}}\mathrm{Fator}_{\mathrm{onoff}} \tag{2.5}$$

$$\mathrm{Fator}_{\mathrm{onoff}} = (1 + \mathrm{prêmio}_{\mathrm{onoff}})^{\frac{dc}{360}} \tag{2.6}$$

onde:

prêmio = valor da taxa do prêmio onoff para o vencimento;

dc = dias corridos entre a partida e o vencimento.

Tabela 7 – Curva de prêmio *onoff* para D0 - 12 de setembro de 2017

Tenor	Prazo	onoff	DC	Fator
1M	13-Oct-17	0.07%	31	1.00006
2M	13-Nov-17	0.08%	62	1.00014
3M	13-Dec-17	0.10%	92	1.00026
4M	12-Jan-18	0.10%	122	1.00034
6M	13-Mar-18	0.10%	182	1.00051
9M	13-Jun-18	0.10%	274	1.00076
12M	13-Sep-18	0.10%	366	1.00102
18M	13-Mar-19	0.07%	547	1.00106
2Y	13-Sep-19	0.05%	731	1.00102
3Y	11-Sep-20	0.03%	1,095	1.00091
4Y	13-Sep-21	0.02%	1,462	1.00081
5Y	13-Sep-22	-0.02%	1,827	0.99899
7Y	13-Sep-24	-0.07%	2,558	0.99504
10Y	13-Sep-27	-0.19%	3,653	0.98089

Fonte – Bloomberg

2.5 Superfície de volatilidade

Opção é um produto financeiro com uma definição tão simples quanto a compra de uma ação, porém com uma modelagem bem mais complexa. Um fato interessante do ativo é que muitas vezes o comprador e vendedor têm visões completamente diferentes de como utilizá-lo em seus *portfolios*.

Tabela 8 – Curva de prêmio *onoff* para D-1 - 11 de setembro de 2017

Tenor	Prazo	onoff	DC	Fator
1M	11-Oct-17	0.07%	30	1.00006
2M	10-Nov-17	0.08%	60	1.00013
3M	12-Dec-17	0.10%	92	1.00026
4M	12-Jan-18	0.10%	123	1.00034
6M	12-Mar-18	0.10%	182	1.00051
9M	12-Jun-18	0.10%	274	1.00076
12M	12-Sep-18	0.10%	366	1.00102
18M	12-Mar-19	0.07%	547	1.00106
2Y	12-Sep-19	0.05%	731	1.00102
3Y	11-Sep-20	0.03%	1,096	1.00091
4Y	10-Sep-21	0.02%	1,460	1.00081
5Y	12-Sep-22	-0.02%	1,827	0.99899
7Y	12-Sep-24	-0.07%	2,558	0.99504
10Y	10-Sep-27	-0.19%	3,651	0.98090

Fonte – Bloomberg

A maneira mais simples de utilizar uma opção para uma posição direcional é apostar no seu *payoff*, ou seja, no seu valor de liquidação. Por exemplo, numa *call*, o comprador espera que o ativo objeto esteja sendo negociado acima do *strike* no vencimento, enquanto o vendedor acredita no contrário. A diferença entre opções e futuros é que o comprador tem a proteção de perder no máximo o prêmio pago na partida.

Porém, existe uma abordagem mais técnica, onde utilizamos a precificação desse tipo ativo para ter acesso a operar um outro risco de mercado, a volatilidade. Ao criarmos um *portfolio* replicante, é possível decompor os riscos de moeda e curvas de juros de uma opção, isolando a volatilidade. Desta maneira, enquanto o comprador de uma opção pode estar visando lucrar com seu *payoff*, o vendedor pode estar operando a volatilidade do ativo objeto, fazendo com que, comumente, ambos obtenham ganhos partindo de operações opostas.

O modelo mais conhecido para precificar opções é o de (BLACK; SCHOLES, 1973), já que possui uma fórmula fechada para a determinação do prêmio e de suas derivadas, isso tudo trabalhando apenas com a volatilidade implícita de um parâmetro não conhecido. Assim temos uma equação e uma incógnita que obriga que uma de suas premissas seja a volatilidade constante, o que normalmente acaba gerando um erro grosseiro na precificação de opções.

Por conta disso, a literatura apresenta outros modelos mais complexos para precificar e calcular as derivadas de opções. (CLARK, 2011) apresenta diversos modelos, como de Heston que utiliza volatilidade estocástica. Nesta dissertação, utilizaremos o modelo de (BLACK, 1976), que será detalhado na precificação do ativo.

Assim como encontramos vários métodos para precificar opções, também temos diversas maneiras de parametrizar curvas de volatilidade. É importante ressaltar que ambos precisam ser coesos entre si. Aqui será apresentada uma maneira simplificada de como o mercado de opções de moedas opera na prática, porém a metodologia de quebra do resultado por fator de rico pode ser estendida para outras métricas também.

2.5.1 Moneyness

Primeiramente é preciso definir o conceito de *moneyness* de uma opção. Uma opção que tem o *strike* igual ao preço futuro do ativo objeto é chamada de *at-the-money*. Uma onde o *strike* seja superior ao valor futuro é *out-off-the-money* e quando este valor é inferior chamamos de *in-the-money*.

2.5.2 Call-Put Parity

Também é necessário mencionar a *call and put parity*. Embora o prêmio pago de uma *call* e um *put* de mesmo *strike* possa variar, o risco de mercado de um *call* é equivalente ao de uma *put* somada à compra do ativo objeto. Assim, ao operar volatilidade, a variação entre *call* ou *put* é irrelevante.

2.5.3 Delta

O último conceito que precisa ser apresentado aqui é o do *delta* da opção. *delta* é a derivada do preço da opção em relação a variação do ativo objeto. No caso de opções de moedas, mapear seu preço de acordo com *delta* faz mais sentido do que pelos *strikes*. Além de possibilitar análises históricas mais fidedígnas do comportamento passado da volatilidade, ao construir uma superfície em função de *deltas*, temos um formato mais suave e coezo.

Após a introdução desses conceitos, fica mais fácil explicar o racional da parametrização escolhida neste trabalho, que é uma simplificação do mais usual no mercado de moedas no Brasil. Nos modelos derivados do (BLACK; SCHOLES, 1973), grande parte da ineficiência da hipótese de volatilidade constante vem do fato de que existe uma alta correlação entre a volatilidade e o *delta* da opção. Portanto, faz todo sentido construir uma superfície de volatilidade levando-se em conta o *delta* além do prazo de vencimento.

Como a volatilidade de uma *call* e *put* de mesmo *strike* é a mesma, o mercado dá preferência para operar opções *out-of-the-money*, já que envolve menor troca de financeiro no prêmio. Assim, opções com *strike* abaixo do valor futuro do ativo serão tratadas como *calls* e as com *strike* acima, serão *puts*. E com isso, toda a superfície terá delta inferior ao de uma *at-the-money*, que será aproximado por 50%.

$$\sigma_{50\Delta} = \sigma_{ATM} \tag{2.7}$$

Os ajustes na superfície buscam atenuar a simplificação de volatilidade constante na hora de precificar as opções. Assim, os fatores mais relevantes são levados em conta para determinar a volatilidade do *delta* de uma opção. Primeiramente o *risk reversal* faz com que as *calls* tenham mais valor que as *puts*, já que normalmente as altas do câmbio são decorrência de maiores incertezas, e vice-versa.

O segundo fator é a chamada *buterfly*, que é oriunda do fato de que movimentos abruptos de mercado são, normalmente, sucedidos por períodos de alta volatilidade. Ou seja, uma opção com *delta* mais baixo deve ser operada com uma volatilidade mais alta que uma *at-the-money*.

Dessa forma, a base de dados será composta de volatilidades *at-the-money-forward*, *risk reversals* 25% e 10% *delta*, e *buterflys* de 25% e 10% *delta*, seguindo as seguintes fórmulas:

$$\sigma_{RR25\Delta} = \sigma_{25\Delta c} - \sigma_{25\Delta p}. \tag{2.8}$$

$$\sigma_{FLY25\Delta} = \frac{\sigma_{25\Delta c} + \sigma_{25\Delta p}}{2} - \sigma_{ATM} \tag{2.9}$$

Fazendo alguns rearranjos e substituindo (2.8) em (2.9), podemos facilmente chegar às seguintes fórmulas para cálculo da volatilidade por *delta*:

$$\sigma_{25\Delta c} = \sigma_{ATM} + \sigma_{FLY25\Delta} + 0.5 \times \sigma_{RR25\Delta} \tag{2.10}$$

e

$$\sigma_{25\Delta p} = \sigma_{ATM} + \sigma_{FLY25\Delta} - 0.5 \times \sigma_{RR25\Delta} \tag{2.11}$$

Tabela 9 – Superfície de volatilidade para D0 - 12 de setembro de 2017

Tenor	Prazo	ATM	RR 25D	RR 10D	FLY 25D	FLY 10D
1M	12-Oct-17	10.66%	2.26%	4.25%	0.38%	1.22%
2M	11-Nov-17	10.85%	2.45%	4.66%	0.40%	1.30%
3M	12-Dec-17	11.10%	2.75%	5.23%	0.43%	1.40%
6M	12-Mar-18	11.70%	3.35%	6.45%	0.53%	1.76%
12M	12-Sep-18	13.15%	3.95%	7.60%	0.63%	2.12%
2Y	12-Sep-19	14,40%	4.85%	9.34%	0.68%	2.33%
3Y	10-Sep-20	15.30%	5.20%	10.01%	0.70%	2.49%
5Y	12-Sep-22	17.10%	5.70%	11.12%	0.83%	3.09%
7Y	12-Sep-24	18.90%	5.95%	11.90%	0.85%	3.27%

Fonte – Bloomberg

Tabela 10 – Superfície de volatilidade para D-1 - 11 de setembro de 2017

Tenor	Prazo	ATM	RR 25D	RR 10D	FLY 25D	FLY 10D
1M	10-Oct-17	10.40%	2.10%	3,94%	0.35%	1.14%
2M	9-Nov-17	10.70%	2.35%	4.41%	0.38%	1.22%
3M	11-Dec-17	10.85%	2.60%	4.88%	0.43%	1.40%
6M	9-Mar-18	11.70%	3.15%	6.06%	0.50%	1.68%
12M	11-Sep-18	13.15%	3.80%	7.32%	0.60%	2.04%
2Y	11-Sep-19	14,35%	4.80%	9.24%	0.65%	2.24%
3Y	10-Sep-20	15.25%	5.20%	10.01%	0.70%	2.49%
5Y	9-Sep-22	17.05%	5.70%	11.12%	0.83%	3.09%
7Y	11-Sep-24	18.85%	5.95%	11.90%	0.85%	3.27%

Fonte – Bloomberg

Com a obtenção desses cinco pontos por prazo de vencimento, será utilizado uma interpolação *spline cúbica* para os outros *deltas*. Já entre prazos, a interpolação será de *flat forward*, que é equivalente à linear na variância total entre os prazos. Segundo (CLARK, 2011):

$$\sigma_{imp}^{flatfwd}(t) = \begin{cases} \sigma_1 & \text{se } t < t_1 \\ \sqrt{[\sigma_i^2 t_i + \sigma_{i,i+1}^2 (t - t_i)]/t} & \text{se } t_i \leq t < t_{i+1} \text{ para } i < N \\ \sigma_N & \text{se } t \geq t_N \end{cases} \tag{2.12}$$

onde:

$$\sigma_{i,i+1}^2 = \frac{\sigma_{i+1}^2 t_{i+1} - \sigma_i^2 t_i}{t_{i+1} - t_i}$$

Essa metodologia ainda é uma simplificação da utilizada no mercado, enquanto em (CARREIRA; JR, 2016), encontra-se uma discussão mais ampla de como garantir que a superfície de volatilidade seja livre de arbitragens.

Por fim, chegamos ao problema onde a superfície é plotada em relação ao seu *delta*, sem esquecer que estas opções têm na sua descrição, um *strike* fixo, e seu *delta* depende, entre outros fatores, da volatilidade. Assim, é necessário utilizar um método iterativo, representado pelo Algoritmo 1 , onde a partir dos dados de uma opção e um chute inicial para seu *delta*, é possível encontrar sua volatilidade.

Algoritmo 1 – Algoritmo de busca da volatilidade implícita a partir do *forward*

```
função VolImplicita(F, K, r, T, Superficie)
    Δ ← 0.5
    σ_imp ← Superficie.Volatilidade(T, Δ)
    erro ← 1
    precisao ← 1E − 5
    i ← 0
    enquanto erro > precisao e i < 100 faça
        Δ ← Delta(F, K, r, T, σ_imp)
        σ_anterior ← σ_imp
        σ_imp ← Superficie.Volatilidade(T, Δ)
        erro ← Modulo(σ_anterior − σ_imp)
        i ← i + 1
    fim enquanto
    retorna σ_imp
fim função
```

CAPÍTULO

3

PRECIFICAÇÃO DOS ATIVOS

O mercado possui uma gama extensa de produtos operados pelos diferentes participantes. Aqui abordaremos alguns deles, mas sempre tentando criar uma didática que pode ser facilmente extrapolada para outros ativos mais complexos ou específicos.

Os produtos podem ser operados em clearings ou no balcão, com características pré definidas ou customizadas. A parte mais importante para a precificação de um ativo é determinar qual seu *numeraire*.

Numeraire que, em francês, significa "dinheiro", "moeda"ou "valor nominal", é um termo econômico que representa uma unidade de contagem. Usualmente, o termo é aplicado para um bem, tornando-se sua base de valor, fazendo com que todos os ativos similares sejam contabilizados nessa base, o que permite uma comparação adequada.

Dessa forma, é importante saber em qual moeda o ativo é liquidado, para determinar o *numeraire* visto que um ativo *onshore* tem como *numeraire* o Real descontado no tempo pelo CDI, enquanto os ativos *offshore* são em Dólares descontados pela curva OIS. Após isso, o cálculo do valor do *portfolio* deve ser padronizado em uma única moeda. Neste trabalho os cálculos serão todos feitos em Dólares, portanto, ativos onshore que são pagos em Real terão seus resultados convertidos para Dólar pela cotação do câmbio *Spot*, ao passo que os ativos offshore já têm seus valores calculados em moeda estrangeira.

No caso de produtos de balcão, o *PNL* é calculado como $PV_D0 - PV_{D-1}$, enquanto que os operados em clearing como a **BMF**, o PNL é igual ao ajuste diário dividido pelo $Spot_{USDBRL}$.

3.1 Contrato Futuro de DI1 da BMF

O contrato de juros na **BMF** é o que tem o maior volume negociado no país. Dessa forma, esse é o ativo mais utilizado por participantes de mercado para tomar posições tanto especulativas como para se protegerem da flutuação da taxa de juros. Outra vantagem desse contrato é que o único risco inerente é o da taxa de juros. Segundo a **BMF**, sua descrição é a seguinte:

"O Contrato Futuro de DI1 tem como ativo subjacente a taxa média diária dos Depósitos Interfinanceiros (DI), calculada e divulgada pela CETIP, compreendida entre a data de negociação, inclusive, e a data de vencimento, exclusive, e é utilizado para proteção e gerenciamento de risco de taxa de juro de ativos/passivos referenciados em DI."

"O contrato tem valor nocional de R$100.000 na data de vencimento, e o valor na data de negociação (PU) é igual ao valor de R$100.000 descontado pela taxa negociada. Como a posição é atualizada diariamente pela Taxa DI através da dinâmica de atualização do PU pelo fator de correção, o investidor que carrega a posição até o vencimento recebe ajustes diários que somados equivalerão à diferença entre a taxa de juro contratada e a realizada, sobre o montante financeiro da operação."

$$PNL_{DI1} = n\left(\frac{100.000}{\text{Fator}_{\text{CDI D0}}} - \text{Fator}_{\text{over}}\frac{100.000}{\text{Fator}_{\text{CDI D-1}}}\right) \tag{3.1}$$

onde:

n = número de contratos.

$$\text{Fator}_{\text{over}} = (1 + \text{CDI}_{overD-1})^{\frac{\text{du}}{252}} \tag{3.2}$$

onde:

$\text{CDI}_{overD-1}$ = valor do CDI *over* no dia D-1 - 11 de setembro de 2017;

du = dias úteis entre **D-1** e **D-0**.

3.2 Contrato Futuro de DOL da BMF

O contrato de DOL é o ativo mais utilizado para operar câmbio no país. O mais líquido é o contrato curto, com vencimento no final do próprio mês corrente, diminuindo os riscos de curvas de juros, aproximando a operação ao máximo de um *spot*. Segundo a **BMF**, sua descrição é a seguinte:

"O Contrato Futuro de Dólar dos Estados Unidos da América pode servir para proteção ou especulação sobre o preço da moeda em data futura, assim como para investidores que, por exemplo tenham recebíveis em dólares dos Estados Unidos da América, ou exposição para pagamentos de passivos na moeda em datas futuras ou até mesmo negociar sobre a tendência da moeda no futuro e assim auferir lucro."

$$PNL_{DOL} = n50,000(Fwd_{D0} - Fwd_{D-1}) \tag{3.3}$$

onde:

n = número de contratos;

Fwd_{D0} = *Forward$_U$SDBRL onshore* calculado pelas curvas de mercado de *D0* para a data de vencimento do contrato;

Fwd_{D-1} = *Forward$_U$SDBRL onshore* calculado pelas curvas de mercado de *D-1* para a data de vencimento do contrato.

Um fato interessante desse contrato é que ele difere de uma **NDF** *onshore* por não trazer a valor presente o ajuste diário. Assim, suas derivadas como *delta* são bem diferentes, e esse efeito se acentua conforme mais longo seja o vencimento. Como praticamente toda a liquidez é concentrada nos dois contratos mais curtos, esse fator fica praticamente desprezível. Mas se o vencimento for mais longo, o modelo apresentado se torna ineficiente, e a correlação entre o risco cambial e o da curva de CDI deve ser levada em conta. Em (CARREIRA; JR, 2016) encontra-se uma discussão de como precificar esse efeito.

3.3 Contrato Futuro de DDI da BMF

O contrato de **DDI** é eficiente para operar a curva de cupom cambial, já que no contrato do **DOL** temos o risco do CDI embutido, e quanto mais longo maior o impacto deste na precificação. Isso faz com que o DDI acabe funcionando melhor pra curva de cupom cambial, dado que este acaba por ser o único risco de mercado embutido na sua precificação além do cambial. Segundo a **BMF**, sua descrição é a seguinte:

"O contrato funciona como uma ferramenta de proteção contra flutuações da taxa de juro referenciada ao dólar. Para entendermos o Contrato Futuro de Cupom Cambial devemos ter em mente aspectos macroeconômicos, em especial, a paridade de juros. Assim, o cupom cambial pode ser interpretado como o rendimento em dólares para estrangeiros que assumem o risco de investir no Brasil."

"O ativo subjacente do contrato, portanto, é a taxa de juro calculada pela diferença entre a taxa média de depósitos interfinanceiros de um dia (DI) e a variação cambial medida pela PTAX800, calculada e divulgada pelo Banco Central do Brasil (BCB)."

"Considerando uma instituição que queira investir na taxa de juro brasileira e precise se proteger contra o risco da variação cambial por conta da internalização de dólares, o Contrato Futuro de Cupom Cambial mostra-se um instrumento de proteção eficiente, pois remunera o investidor pela diferença entre a variação da taxa de juro em dólar contratada e a realizada do período. Importante considerar que o contrato é negociado em taxa de juro “suja”, ou seja, desconsidera-se a variação cambial entre o momento do negócio e a PTAX do dia útil anterior à data de negociação."

$$PNL_{DDI} = n50.000(\frac{\mathrm{Spot}_{\mathrm{D0}}}{\mathrm{Fator}_{\mathrm{CC\ D0}}} - \mathrm{Fator}_{\mathrm{over}}\frac{\mathrm{Spot}_{\mathrm{D\text{-}1}}}{\mathrm{Fator}_{\mathrm{CC\ D\text{-}1}}}) \tag{3.4}$$

onde:

n = número de contratos.

3.4 NDF Offshore

Uma **NDF**, ou *non deliverable forward*, é um contrato futuro normalmente entre uma moeda não conversível, onde não há troca física de moedas. A liquidação se dá calculando a diferença entre a taxa de liquidação no vencimento (no caso do **USDBRL**, normalmente é usada a taxa *PTAX* divulgada pelo Banco Central) e a acordada na partida, multiplicada pelo principal da operação. No caso da **NDF** *offshore*, o valor é pago em Dólares, e na *onshore* em Reais.

NDFs são mais comumente operadas para prazos menores de um ano, mas podendo ter vencimentos mais longos. Esse instrumento se tornou popular com empresas que buscavam se proteger de variações cambiais em países emergentes, porém, hoje este é o tipo de contrato mais líquido para moedas *non deliverable*.

A seguir temos a fórmula que utilizaremos para determinar o valor presente de uma **NDF** *offshore*.

$$\mathrm{FV}_{\mathrm{NDF}} = \frac{\mathrm{N}(S_T - K)}{S_T} \tag{3.5}$$

onde:

N = notional ou valor do finaceiro do contrato;

S_T = preço do Spot no vencimento do contrato que será calculado como $Forward_USDBRL$ *offshore* pelas curvas de mercado de *D-1* para a data de vencimento do contrato;

K = taxa de câmbio acordada na partida do contrato;

Assim, para obtermos o **PV** do produto, é necessário trazer o **FV** a valor presente:

$$\mathrm{PV}_{\mathrm{NDF}} = \frac{\mathrm{FV}_{\mathrm{NDF}}}{\mathrm{Fator}_{\mathrm{OIS}}} \tag{3.6}$$

$$\mathrm{PV}_{\mathrm{NDF}} = \frac{\mathrm{N}(\mathrm{Fwd}_{USDBRL} - K)}{\mathrm{Fwd}_{USDBRL}\mathrm{Fator}_{\mathrm{OIS}}} \tag{3.7}$$

$$\mathrm{PNL}_{\mathrm{NDF}} = \mathrm{PV}_{\mathrm{D0}} - \mathrm{PV}_{\mathrm{D\text{-}1}} = \frac{\mathrm{N}(\mathrm{Fwd}_{D0} - K)}{\mathrm{Fwd}_{D0}\mathrm{Fator}_{\mathrm{OIS\ D0}}} - \frac{\mathrm{N}(\mathrm{Fwd}_{D-1} - K)}{\mathrm{Fwd}_{D-1}\mathrm{Fator}_{\mathrm{OIS\ D\text{-}1}}} \tag{3.8}$$

3.5 NDO Offshore

Uma **NDO**, ou *non deliverable option*, é uma opção onde o comprador tem o direito, mas não a obrigação de comprar ou vender um determinado ativo por um preço pré determinado. Assim, a **NDO** tem características semelhantes as da **NDF**, produto similar a uma **NDF**, onde normalmente se tem uma moeda não conversível e não há troca física de moedas, apenas o acerto financeiro do lucro e da perda da operação. Nesse tipo de opção não é necessárioque o comprador informe o vendedor da intenção de exercer a opção. Isso é feito automaticamente se for vantajoso para o detentor da opcionalidade.

A liquidação também é parecida, porém, notamos dois tipo de opções a de compra, chamada *call*, e a de venda, que chamamos de *put*. Na partida, o comprador paga um prêmio para o vendedor, e no caso da *call*, o valor de liquidação no vencimento é a diferença entre a taxa de câmbio no vencimento (no caso do **USDBRL**, normalmente é usada a taxa *PTAX* divulgada pelo Banco Central) e a acordada na partida, multiplicada pelo principal da operação, se a primeira taxa for maior que a segunda. Se este cálculo não for obedecido, o caixa da operação fica sendo zero, já que o comprador opta por não exercer a opção. A *put* é a diferença entre a taxa de câmbio acordada na partida e a do vencimento.

NDOs também são populares com empresas buscando proteções, ou especuladores buscando lucrar com variações no mercado financeiro, porém contando com proteções ou de forma oposta, aumentando a alavancagem de seus *portfolios*.

Existem vários modelos para precificar opções no mercado, assim como foi descrito na definição da superfície de volatilidade. Nesta tese utilizaremos o modelo (BLACK, 1976) que é derivado do (BLACK; SCHOLES, 1973), o primeiro método amplamente utilizado no mercado financeiro para cálculo do prêmio de opções.

3.5.1 *Modelo de Black and Scholes*

Black e Scholes (1973) propuseram um modelo para precificar opções europeias em que o ativo subjacente fosse uma ação e, para tal, assumiram algumas premissas:

1. a taxa de juros livre de riscos é constante e conhecida;
2. o ativo objeto, no caso uma ação, não paga dividendos;
3. não há custos de transação;
4. é permitida a venda descoberta;
5. não há oportunidades de arbitragem;
6. as operações podem ser realizadas continuamente sem impactar o mercado.

O modelo proposto também assume que as variações do preço do ativo objeto em um curto espaço de tempo segue uma distribuição normal. Conforme a descrição feita por Hull (2006), definimos μ como o retorno esperado e σ como a volatilidade do ativo objeto. Em um espaço de tempo Δt o retorno esperado é dado por $\mu \Delta t$ e o desvio padrão desse retorno por $\sigma\sqrt{\Delta t}$.

Conforme levantado por Hoek (2012), o preço de uma ação é uma variável cuja variação no tempo é imprevisível e, portanto, pode ser representada por um processo estocástico. Além disso, é esperado que as variações de preço no passado e no futuro sejam independentes tornando esse um processo sem memória e satisfazendo a propriedade Markov. Dadas essas propriedades, o preço do ativo objeto no tempo pode ser modelado por um movimento Browniano.

Definição 1. Um processo $\{B_t : 0 \leqslant t < \infty\}$ contínuo no tempo é dito movimento Browniano se satisfaz as seguintes condições:

1. $B_0 = 0$;
2. Se $t_1 < t_2 \leqslant t_3 < t_4$ então os incrementos $B_{t_2} - B_{t_1}$ e $B_{t_4} - B_{t_3}$ são independentes ;
3. $B_t - B_s \sim N(0, t-s)$ para $0 \leqslant t \leqslant s$;
4. $B = \{B_t : 0 \leqslant t < \infty\}$ tem trajetórias contínuas.

O movimento Browniano descrito acima não possui *drift*, e, portanto, o seu valor esperado a qualquer momento no futuro é igual ao seu valor atual. Entretanto, como já foi dito anteriormente, o retorno esperado do ativo objeto é dado por $\mu \Delta t$. Assim, o modelo Black-Scholes assume uma equação diferencial estocástica (EDE) para modelar o preço da ação em questão, representada por um movimento Browniano geométrico, dada na seguinte foma:

$$dS_t = \mu dS_t dt + \sigma S_t dBt.$$

Aplicando o lema de Itô, pode-se facilmente verificar que:

$$dln(S_t) = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma dBt.$$

Vale notar que $ln(S_t)$ possui *drift* e volatilidade constantes e, portanto, segue uma distribuição normal. Como consequência, tem-se que $ln(S_T)$ segue uma distribuição lognormal:

$$ln(S_T) \sim N\left(ln(S_0) + \left(\mu - \frac{1}{2}\sigma^2\right) T, \sigma^2 T\right)$$

$$S_T \sim N\left(ln(S_0) + \left(\mu - \frac{1}{2}\sigma^2\right) T, \sigma^2 T\right)$$

$$\mathbf{E}\left[S_T\right] = S_0 e^{\mu T}$$

Com base nesse modelo capaz de descrever o comportamento do preço de uma ação, Black e Scholes (1973) demonstram através do lema de Itô, que o preço de uma *call* européia de valor $V(S,t)$ pode ser descrito pela seguinte equação diferencial parcial:

$$dV = \left(\frac{\partial V}{\partial S}\mu S + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right) dt + \sigma S \frac{\partial V}{\partial S} dB.$$

Baseado no princípio de não arbitragem é possível construir uma carteira livre de risco, composta pela *call* descrita acima e por uma certa quantidade do ativo objeto, e finalmente chegar à equação diferencial parcial de Black e Scholes (1973):

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0. \tag{3.9}$$

Resolvendo a Equação 3.9 considerando o resultado esperado da opção do tipo *call* europeia que vem sendo utilizada nesta seção, pode-se chegar a uma solução analítica para o preço:

$$C(S,t) = S\phi(d_1) - Ke^{-r\tau}\phi(d_2) \tag{3.10}$$

onde $\phi(.)$ é a função de distribuição acumulada de uma distribuição normal padrão e τ é o prazo até a data de maturidade da opção $(T-t)$. Os termos d_1 e d_2 são dados por:

$$d_1 = \frac{ln(S/K) + (r + \sigma^2/2)\tau}{\sigma\sqrt{\tau}},$$

e

$$d_2 = d_1 - \sigma\sqrt{\tau}.$$

3.5.2 Paridade call-put

Para que não haja condições de arbitragem, Stoll (1969) afirma que o preço de *call* implica um certo preço justo para uma *put* de mesmo *strike* e data de maturidade. Hull (2006) demonstra tal afirmação com a utilização de duas carteiras:

- Carteira A: uma *call* européia mais uma quantidade em dinheiro igual a $Ke^{-r\tau}$
- Carteira B: uma *put* com mesmo vencimento mais uma quantidade do ativo objeto

Como as duas carteiras possuem o mesmo valor esperado $max(S_T, K)$, e as opções são do tipo europeias, os seus valores presentes devem ser iguais. Assim, a paridade *call-put* pode ser definida por:

$$c + Ke^{-r\tau} = p + S_0. \tag{3.11}$$

Dessa forma, conclui-se que uma *put* também europeia tem seu preço dado por:

$$P(S,t) = Ke^{-r\tau}\phi(-d_2) - S\phi(-d_1). \tag{3.12}$$

3.5.3 Black 1976

Com algumas modificações, o modelo de Black-Scholes pode ser estendido para as demais classes de ativos financeiros. Na implementação do sistema proposto neste trabalho, as opções de moedas serão precificadas utilizando as fórmulas derivadas do modelo de Black (1976), inicialmente proposto para a precificação de opções de futuros. O porquê de tal utilização será explicado a seguir.

Começando pelo caso de ações que pagam dividendos, Hull (2006, p. 313) mostra que o valor inicial da ação S_0 pode ser reduzido pela taxa de dividendo q resultando em $S_0e^{-q\tau}$. Fazendo essa substituição nas equações (3.10) e (3.12), obtém-se:

$$c = S_0e^{-q\tau}\phi(d_1) - Ke^{-r\tau}\phi(d_2) \tag{3.13}$$

$$p = Ke^{-r\tau}\phi(-d_2) - S_0e^{-q\tau}\phi(-d_1) \tag{3.14}$$

Como:

$$ln\left(\frac{S_0e^{-q\tau}}{K}\right) = ln\frac{S_0}{K} - q\tau$$

os parâmetros d_1 e d_2 são reescritos na forma:

$$d_1 = \frac{ln(S_0/K) + (r - q + \sigma^2/2)\tau}{\sigma\sqrt{\tau}}$$

$$d_2 = d_1 - \sigma\sqrt{\tau}$$

Ao mudar para o mercado de opções de moedas, novamente podemos utilizar uma analogia de Hull (2006, p. 115), que compara um par de moedas a um ativo pagando uma taxa de juros ou dividendos conhecida, nesse caso $q = r_f$. Com isso, partindo da **??** e substituindo em (3.13) e (3.14), tem-se que o preço de uma opção de moedas é dado por

$$c = e^{-rT}[F_0\phi(d_1) - K\phi(d_2)] \tag{3.15}$$

$$p = e^{-rT}[K\phi(-d_2) - F_0\phi(-d_1)] \tag{3.16}$$

onde:

$$d_1 = \frac{ln(F_0/K) + \sigma^2 T/2}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

As fórmulas (3.15) e (3.16) são idênticas às fórmulas do modelo proposto por Black (1976) para precificação de opções de futuros. De fato, quando a taxa livre de risco é constante ou varia em função do tempo, o preço de um futuro é igual ao preço de *forward* com vencimento na mesma data. Com isso, explica-se a utilização deste modelo durante o desenvolvimento do sistema.

3.6 *Swap* Pré vs CDI Offshore

Um *swap* é um derivativo financeiro onde se troca riscos de uma posição ativa e outra passiva entre duas partes. Nesse caso, um participante fica ativo em um valor com juros pré fixado, ou seja, já é conhecido o financeiro exato que receberá no vencimento e passivo em juros pós fixados. A liquidação se dá apenas pelo diferença entre os valores no vencimento.

Esse produto, assim como o futuro de DI1 tem como propósito tomar uma posição na curva de juros do Brasil, porém como não existe ajuste diário e sua liquidação é em Dólares, fazendo surgir outros efeitos surgem na precificação do ativo.

A seguir temos a equação utilizada para calcular o valor presente do *swap* pré CDI *offshore*:

$$\mathrm{FV_{pré}} = \mathrm{N}(1 + \mathrm{Tx_{pré}})^{\frac{du}{252}} \tag{3.17}$$

onde:

N = notional ou valor do finaceiro do contrato;

$\mathrm{Tx_{pré}}$ = taxa pré fixada acordada na partida do *swap*;

du = dias úteis no calendário Brasil entre a partida do *swap* e o vencimento.

$$\mathrm{FV_{cdi}} = \mathrm{NFator}_{CDIacum}\mathrm{Fator}_{CDI} \tag{3.18}$$

onde:

N = notional ou valor do finaceiro do contrato;

$\mathrm{Fator}_{CDIacum}$ = é o fator acumulado das taxas de CDI over desde a partida do swap até D0 como na equação a seguir;

$$\mathrm{Fator}_{CDIacum} = \prod_{i=partida}^{D0} (1+\mathrm{CDI_{over\ i}})^{\frac{du}{252}} \tag{3.19}$$

Como esses valores futuros estão em Reais, mas a liquidação está em Dólares, para encontrar o valor presente é necessário dividir pelo *Forward* e descontar pela curva **OIS**. No caso de um participante que está ativo na ponta pré fixada, a fórmula do valor presente pode ser escrita da seguinte maneira:

$$\mathrm{PV_{swapoff}} = \frac{\mathrm{FV_{pré}} - \mathrm{FV_{cdi}}}{\mathrm{Fwd_{USDBRL}Fator_{OIS}}} \tag{3.20}$$

$$\mathrm{PV_{swapoff}} = N\frac{\mathrm{Fator_{CC}}}{\mathrm{Spot}_{usdbrl}\mathrm{Fator_{onoff}Fator_{OIS}}}\left(\frac{(1+\mathrm{Tx_{pré}})^{\frac{du}{252}}}{\mathrm{Fator_{CDI}}} - \mathrm{Fator}_{CDIacum}\right) \tag{3.21}$$

Como podemos ver pela equação do valor presente do *swap* pré cdi *offshore*, o fato da liquidação ser em moeda estrangeira cria esse fator de conversão $\frac{\mathrm{Fator_{CC}}}{\mathrm{Spot}_{usdbrl}\mathrm{Fator_{onoff}Fator_{OIS}}}$ em relação ao derivativo onshore, gerando o que chamamos de convexidade do ativo. Em (CARREIRA; JR, 2016) encontra-se um extensa discussão dos efeitos causados por essa convexidade tanto na precificação do produto, como na gestão de seus riscos.

CAPÍTULO

4

DESENVOLVIMENTO

Após as curvas estarem construídas e os produtos modelados, é possível determinar facilmente o valor presente do *portfolio*. A literatura é bem extensa em como se calcular as derivadas parciais para os diversos produtos, onde, muitas vezes, são chamadas de *gregas*, e são obtidas como resposta para a sensibilidade de cada fator de risco.

Uma abordagem usual para o cálculo de resultado por fator de risco, é utilizar os valores encontrados nas *gregas* e multiplicá-los pela variação de cada ativo. Esse método, mesmo que feito de maneira simplificada, ignora derivadas de segunda ordem, sendo que dependendo do ativo, altera significativamente o resultado final.

Assim, muitas vezes encontramos cálculos de derivadas analíticas de segunda ordem ou derivadas cruzadas com mais de um fator de risco, para refirnar os cálculos. Porém, neste trabalho, o foco não é determinar a sensibilidade do portoflio em relação a cada risco de mercado, mas apenas explicar o resultado. Portanto, em nenhum momento será calculado derivadas parciais para os produtos, mesmo que na prática, isso seja vital para a atuação do gestor.

Como a demanda mais usual no mercado financeiro é explicação do **PNL** diário, o exemplo proposto será do dia 12 de setembro de 2017em relação ao dia anterior, 11 de setembro de 2017. Para facilitar, criaremos a notação de **D0** para simbolizar a data para a qual serao feitos os cálculos para explicar o resultado, e **D-1** o dia anterior.

Sendo assim, o método proposto consiste em primeiramente calcular o valor do *portfolio* para **D-1** com os parâmetros de mercado para esse mesmo dia. Esse valor será chamado de **PV D-1**.

$$\mathrm{PV}_{D-1} = \mathrm{PV}_{\text{D-1}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) \tag{4.1}$$

4.1 PNL por *Theta*

Em seguida, calculamos o valor do *portfolio* para **D0**, porém com as curvas de **D-1**. A diferença entre esse resultado e o **PV D-1** é o theta, ou seja, a variação dos ativos em relação ao tempo. Nesse passo é necessário discutir como as curvas serão roladas. Esse processo pode ser feito de diversas formas, e, para cada curva vale uma discussão diferente de qual seria a mais adequada. No entanto, haja visto que este trabalho não aborda o resultado de abertura do dia, quando ainda não se têm curvas de mercado para **D0**, mas sim de fechamento, não será preciso entrar nessa discussão.

$$\mathrm{PNL}_{theta} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} \tag{4.2}$$

4.2 PNL por *Delta*

Uma vez calculado o ganho pela variação do tempo, o primeiro parâmetro a ser atualizado será o câmbio. Assim, calcula-se o *portfolio* para **D0**, ainda com todas as curvas de D-1, porém já com o USDBRL de **D0**. Ao subtrair esse valor do **PV D-1** e o **PNL** por *theta*, encontra-se o ganho devido a variação cambial.

$$\mathrm{PNL}_{delta} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\mathrm{D0}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.3}$$

4.3 PNL por *Rho* e *Vega*

Esse passo é repetido sucessivamente curva a curva. Calculando o *portfolio* para **D0** com todos os parâmetros de **D-1**, exceto aquele que busca calcular seu resultado. Então, após subtrair o novo valor do *portfolio* pelo **PV D-1** e o **PNL** por *theta*, encontra-se a explicação do resultado por esse fator de risco.

Esse método leva vantagem quando calcula as derivadas parciais e a multiplica pelas variações dos ativos, já que não somente contabiliza as derivadas de segunda ordem, como também evita a necessidade de se determinar a variação das curvas para o vencimento específico de cada ativo existente no *portfolio*. Ou seja, a variação das curvas dependem do ponto observado, o que geraria a necessidade de realizarmos um cálculo diferente para cada ativo. Já no modelo proposto, ao calcularmos o valor dos produtos com a nova curva, todos esses passos acabam sendo feitos automaticamente.

$$\mathrm{PNL}_{CDI} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\mathrm{D0}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.4}$$

$$\mathrm{PNL}_{OIS} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\mathrm{D0}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.5}$$

$$\mathrm{PNL}_{CC} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\mathrm{D0}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.6}$$

$$\mathrm{PNL}_{ONOFF} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\mathrm{D0}}, \mathrm{VOL}_{\text{D-1}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.7}$$

$$\mathrm{PNL}_{VOL} = \mathrm{PV}_{\mathrm{D0}}(\mathrm{Spot}_{\text{D-1}}, \mathrm{CDI}_{\text{D-1}}, \mathrm{OIS}_{\text{D-1}}, \mathrm{CC}_{\text{D-1}}, \mathrm{ONOFF}_{\text{D-1}}, \mathrm{VOL}_{\mathrm{D0}}) - \mathrm{PV}_{\text{D-1}} - \mathrm{PNL}_{theta} \tag{4.8}$$

4.4 PNL por *Cross Curves*

Após o cálculo individual de cada fator de risco, é necessário apurar o efeito das derivadas cruzadas entre eles e a relevância desse fenômeno depende de cada produto. Por exemplo, opções ou *swaps* mais longos, normalmente estão mais sujeitos a esses efeitos. É importante frisar que isso não ocorre apenas entre dois fatores de risco, mas entre todos os que fazem parte da precificação de cada produto.

Assim, seria necessário uma combinação muito extensa de todas as derivadas cruzadas possíveis para explicar o resultado de um *portfolio* mais amplo. Então, na prática, é importante que o gestor conheça quais os produtos ele possui em carteira e quais são suas características, sabendo escolher quais derivadas cruzadas são as mais relevantes para serem analisadas em separado.

Uma vez explicada a lógica por trás da metodologia do cálculo das derivadas cruzadas, dada a dinâmica dos produtos escolhidos nesta tese, serão calculados apenas os resultados oriundos da derivada cruzada entre o câmbio e a superfície de volatilidade, e entre as curvas de juros no Brasil, em Dólares, cupom cambial e prêmio onoff.

$$\begin{aligned}\text{PNL}_{SpotxVOL} = \text{PV}_{\text{D0}}(\text{Spot}_{\text{D0}}, \text{CDI}_{\text{D-1}}, \text{OIS}_{\text{D-1}}, \text{CC}_{\text{D-1}}, \text{ONOFF}_{\text{D-1}}, \text{VOL}_{\text{D0}}) - \text{PV}_{\text{D-1}} \\ - \text{PNL}_{\text{Spot}} - \text{PNL}_{\text{VOL}} - \text{PNL}_{theta}\end{aligned} \tag{4.9}$$

$$\begin{aligned}\text{PNL}_{XCurves} = \text{PV}_{\text{D0}}(\text{Spot}_{\text{D-1}}, \text{CDI}_{\text{D0}}, \text{OIS}_{\text{D0}}, \text{CC}_{\text{D0}}, \text{ONOFF}_{\text{D0}}, \text{VOL}_{\text{D-1}}) - \text{PV}_{\text{D-1}} \\ - \text{PNL}_{\text{CDI}} - \text{PNL}_{\text{OIS}} - \text{PNL}_{\text{CC}} - \text{PNL}_{\text{ONOFF}} - \text{PNL}_{theta}\end{aligned} \tag{4.10}$$

4.5 Resíduo

No final, é necessário calcular o valor do *portfolio* com todas as curvas atualizadas, que será chamado de **PV D0**. A diferença entre o **PV D0** e **PV D-1** é o resultado de fato no dia 12 de setembro de 2017. Ao subtrair desse **PNL** todos os resultados obtidos pelo fatores de risco individualmente, chega-se a um resíduo. Para que o método seja satisfatório, é necessário que esse resíduo seja suficientemente pequeno para que este possa continuar não sendo mapeado. Se algum fator de risco que impacte produtos no *portfolio* não for discretizado, seu efeito também aparecerá no resíduo.

$$\begin{aligned}\text{PNL}_{residuo} = \text{PV}_{\text{D0}}(\text{Spot}_{\text{D0}}, \text{CDI}_{\text{D0}}, \text{OIS}_{\text{D0}}, \text{CC}_{\text{D0}}, \text{ONOFF}_{\text{D0}}, \text{VOL}_{\text{D0}}) - \text{PV}_{\text{D-1}} - \text{PNL}_{\text{Spot}} \\ - \text{PNL}_{\text{CDI}} - \text{PNL}_{\text{OIS}} - \text{PNL}_{\text{CC}} - \text{PNL}_{\text{ONOFF}} - \text{PNL}_{\text{VOL}} - \text{PNL}_{SpotxVOL} - \text{PNL}_{XCurves} - \text{PNL}_{theta}\end{aligned} \tag{4.11}$$

CAPÍTULO 5

RESULTADOS

Neste capítulo, os modelos descritos anteriormente serão colocados em prática, calculando resultados para um *portfolio* teórico com operações genéricas com os produtos já descritos anteriormente, que reproduzem os riscos comumente encontrados no mercado financeiro. Para isso, será utilizada a ferramenta criada na gestora de recursos que o autor da tese trabalhava, que foi batizada como SIRI ou Sistema de informações de risco integrada.

5.1 *Siri*

A ferramenta foi desenvolvida em *C-sharpe* com o intuito de servir tanto aos gestores como a área de risco. Como o excel já é o sistema mais utilizado pelo mercado financeiro, o SIRI foi feito como um *add-in*, utilizando toda a *interface* já familiar para os usuários. Os anexos contém alguns dos códigos fonte implementados no sistema.

O sistema pode ser dividido em duas frentes principais. Na primeira, ele constrói as cuvas de mercado com os paramêtros estabelecidos. Nele foram programadas diversas interpolações possíveis, como *spline cúbico*, *flat forward*, entre outros, permitindo que o usuário escolha qual delas será utilizada para cada curva, conforme as discussões do Capítulo 2. Depois disso, a curva é definida na memória do computador, contendo tanto as informações pré estabelecidas como os preços de mercado.

A outra frente que a ferramenta atua é em precificar ativos. Conforme descrito no Capítulo 3, para cada produto foi descrito um modelo de precificação para o cálculo do seu valor presente. Assim, cada curva é alocada em uma célula de memória, onde posteriormente é utilizada para a precificação dos ativos.

Assim, o SIRI é capaz de calcular os resultados quebrados por fator de risco, exatamente como descrito no Capítulo 4, com uma função que calcula o **PV** utilizando como parâmetros de entrada as curvas de mercado definidas para cada data.

Um ponto a ser destacado da ferramenta, é que por conta da segregação dos cálculos das curvas com a precificação dos ativos, é possível facilmente criar um novo modelo para o cálculo do valor presente para cada produto, sem alterar a construção das curvas de mercado. Com isso, o aprimoramento dos modelos, que possibilitam aumentando a robustez dos resultados, fica bem facilitada.

5.2 *Portfolio* Teórico

Para exemplificar o método proposto, um *portfolio* teórico foi criado para a realização dos cálculos, utilizando todos os produtos que tiveram sua precificação descrita anteriormente. Embora fictíceos, as operações são das mais comuns no mercado financeiro, escolhidas de maneira que os resultados sejam didáticos, deixando claro os efeitos de cada fator de risco.

1. **DI1** - Posição aplicada em juros (ativo na ponta pré fixada) de 1.500 contratos do **DI1** no vencimento 02-Jan-23;
2. **DOL** - Posição comprada de 2.000 contratos do **DOL** no vencimento 02-Jan-20;
3. **DDI** - Posição tomada em juros (passivo na ponta pré fixada) de 2.500 contratos do **DDI** no vencimento 02-Jan-20;
4. **NDF** - Posição vendida em 250.000.000 de dólares em uma **NDF** *offshore* com vencimento em 15-Jul-19;
5. **NDO** - Posição comprada em 100.000.000 de dólares de uma *CALL* em uma **NDO** *offshore* com *strike 3,30* e vencimento em 15-Jul-19;
6. **Swap off** - Posição tomada em juros (ativo na ponta CDI) de 300.000.000 de reais de valor futuro de um *swap pré cdi offshore* com vencimento 02-Jan-23;

5.3 Resultados

Após a definição dos ativos que integrarão o *portfolio*teórico utilizado, os dados são inseridos no sistema SIRI, e os resultados podem ser visualizados através da Figura 1.

Figura 1 – Resultados calculados com SIRI em Excel

	Theta	Spot	CDI	OIS	CC	onoff	VOL	VOL x Spot	X curves	PNL	Residuo
DI1	0	0	55,448	0	0	0	0	0	0	**55,448**	0
DOL	(23,729)	803,376	57,655	0	(51,394)	0	0	0	18	**785,926**	0
DDI	24,671	(835,244)	0	0	53,433	0	0	0	384	**(756,757)**	0
NDF	48,817	(1,638,623)	(18,731)	4,782	126,259	85,096	0	0	(862)	**(1,393,261)**	0
NDO	(54,490)	1,766,745	19,518	(4,458)	(131,152)	(88,473)	20,878	(883)	(12,761)	**1,515,032**	108
Swap off	18,811	(184,719)	(123,459)	(70,518)	(95,465)	7,507		0	2,972	**(444,871)**	0
Total	**14,079**	**(88,466)**	**(9,568)**	**(70,193)**	**(98,319)**	**4,130**	**20,878**	**(883)**	**(10,248)**	**(238,482)**	**108**

A tabela tem nas suas linhas os produtos e nas colunas os fatores de risco (descrito no Capítulo 2). Assim, é esperado que cada ativo mostre resultados apenas nos fatores de risco que influenciam sua precificação, como por exemplo, no contrato de **DI1**, temos **PNL** apenas da curva CDI.

Sendo assim, com posse dessa tabela, pode-se facilmente visualizar quais fatores de risco foram responsáveis pelo resultado do *portfolio*. Também é possível verificar que alguns ativos anulam um pedaço do risco de outro, o que comumente é realizado por operadores para escapar de riscos indesejáveis inerente dos produtos escolhidos.

Também pode-se perceber que o resultado obtido é satisfatório, já que os resíduos da diferença entre os cálculos fechados do **PNL** e a soma dos resultados por cada fator de risco se mostrou desprezível.

CAPÍTULO

6

CONCLUSÃO

Determinar precisamente a origem dos resultados de uma carteira de ativos parece algo essencial para a indústria financeira, porém muitos participantes ainda não possuem ferramentas que realizem esse tipo de cálculo. Muitas vezes os operadores têm apenas acesso ao resultado de cada produto e da quebra do risco do *portfolio* por cada fator, porém sem conseguir determinar a origem precisa de seu **PNL**.

Neste trabalho foi possível propor um método para a apuração desse resultado dividido pelos fatores de risco de mercado, de uma maneira mais simples e com maior precisãodo que as formas que são usualmente debatidas na literatura especializada.

O desenvolvimento prévio do sistema SIRI foi vital para a realização dos cálculos. Criado para ser utilizado dentro da plataforma do Excel, com ele é possível customizar a definição das diversas curvas de mercado, assim como, desenvolver de forma separada modelos para a precificação de cada ativo.

A intenção sempre foi propor uma metodologia e expor a sua lógica, fazendo uma análise qualitativa das vantagens e desvantagens de cada modelo utilizado. Apesar do fato de que, muitas vezes, os modelos escolhidos são versões simplificadas dos que são utilizadas pelas instituições no mercado financeiro, demonstramos como é possível a substituição de um modelo pelo outro. Para isso, a customização que o sistema SIRI possibilita, permite que cada parte do cálculo possa ser editada sem alterar as outras etapas, gerando uma fácil evolução dos modelos criados a partir de um mais simples, porém já com resultados satisfatórios.

Sendo assim, esse trabalho deixa portas abertas em diferentes frentes, sendo a mais óbvia, a simples implementação de mais produtos e curvas de mercado, assim como utilização de modelos mais robustos para a precificação dos ativos. Produtos como opções de juros e ativos de renda variável não foram abordados, mas poderiam ser facilmente modelados a partir da metodologia proposta. Outra frente de evolução seria o refinamento da construção da superfície de volatilidade e da precificação de opções.

Outra área que o trabalho não abordou, mas que abre espaço para outro aprofundamento, são os cálculos dos riscos de mercado da carteira, ou seja, as derivadas parciais em relação às curvas construídas. Para isso não foi propusemos nenhuma metodologia nessa monografia e, para isso, seria necessário um trabalho mais abrangente. Uma vez determinados os riscos do *portfolio*, é interessante analisar a coerência entre o **PNL** explicado por fator de risco e o esperado segundo a variação dos ativos e os riscos da carteira, sem negligenciar a importância disso para operadores ao passo que muitas empresas não possuem sistemas aptos a esse tipo de análise.

Também mencionamos, ainda que de forma sucinta que a rolagem das curvas de um dia para o outro não é algo trivial, e é repleto de definições arbitrárias. Como o foco do trabalho era determinar o resultado ao final do dia, o operador já dispunha das curvas de mercado para D0, porém, na prática, um dos grandes desafios de *market makers* é determinar qual a curva de abertura, baseada na do dia anterior, mas rolada para D0. Isso pode até impactar o resultado obtido pelo PNL_{theta}, que seria explicado por outros fatores de risco. Nessa frente existe amplo espaço para discussão teórica e prática, que gera uma didática útil para operadores entenderem a dinâmica implícita em cada fator de risco e produto.

Contudo, apesar de todos os caminhos possíveis para uma ampliação do escopo da tese e o aperfeiçoamento dos modelos utilizados - como mostram os resultados, o objetivo de calcular a quebra do resultado de uma carteira de ativos foi atingido com sucesso quando obtivemos resíduos desprezíveis. Concluindo, esse sistema foi implementado com sucesso na gestora de recursos, disponibilizando uma ferramenta poderosa para os operadores.

REFERÊNCIAS

BLACK, F. The pricing of commodity contracts. **Journal of financial economics**, Elsevier, v. 3, n. 1-2, p. 167–179, 1976. Citado nas páginas 28, 37, 40 e 41.

BLACK, F.; SCHOLES, M. The pricing of options and corporate liabilities. **Journal of political economy**, The University of Chicago Press, v. 81, n. 3, p. 637–654, 1973. Citado nas páginas 28, 29, 37 e 39.

BURDEN, R. L.; FAIRES, J. D. **Numerical analysis**. 9th. ed. Cengage Learning, 2010. ISBN 9781133169338. Disponível em: <https://books.google.com.br/books?id=Dbw8AAAAQBAJ>. Citado na página 65.

CARREIRA, M. C. S.; JR, R. J. B. **Brazilian derivatives and securities: pricing and risk management of FX and interest-rate porfolios for local and global markets**. [S.l.]: Palgrave Macmillan UK, 2016. Citado nas páginas 20, 22, 24, 31, 35 e 42.

CLARK, I. J. **Foreign exchange option pricing: a practitioner's guide**. [S.l.]: John Wiley & Sons, 2011. Citado nas páginas 28 e 31.

FRANçA, D. M. **DERIVATIVOS CAMBIAIS DO MERCADO BRASILEIRO: PRECIFICAÇÃO E ADMINISTRAÇÃO DE RISCO**. Dissertação (Mestrado) — Fundação Getulio Vargas, 2010. Citado na página 24.

HOEK, B. v. d. **Alternative swaption valuation methods**. Dissertação (Mestrado) — Tilburg University, 2012. Citado na página 38.

HULL, J. **Options, futures, and other derivatives**. 6th. ed. Pearson Prentice Hall, 2006. ISBN 978-0-13-197705-1. Disponível em: <http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+563580607&sourceid=fbw_bibsonomy>. Citado nas páginas 38, 39 e 40.

STOLL, H. The relationship between put and call option prices. **Journal of finance**, v. 24, n. 5, p. 801–24, 1969. Disponível em: <http://EconPapers.repec.org/RePEc:bla:jfinan:v:24:y:1969:i:5:p:801-24>. Citado na página 39.

APÊNDICE

A

CLASSE CURVES

Código-fonte 1 – Implementação das curvas de mercado

```
using QLNet;
using SIRI.Mathematics.Interpolation;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SIRI.Pricer
{
    public class Curve : AbstractCurve
    {
        #region Members
        private Date[] _startDates;
        private Date[] _endDates;

        private double[] _timeToMaturity;
        private double[] _rates;

        private double[] _mask;
        private int _numOfUnmaskedItems;

        private double[] _timeMasked;
        private double[] _rateMasked;
        #endregion

```

```
        #region Properties
        public double[] Rates
        {
            get { return _rates; }
            set
            {
                //value = value.TrimDouble();
                if (value.Length < _startDates.Length)
                {
                    throw new ArgumentException("Rates array length must be greater or equal to dates array length.");
                }
                else if (value.Length == _startDates.Length)
                {
                    _rates = value;
                }
                else
                {
                    _rates = new double[_startDates.Length];
                    for (int i = 0; i < _startDates.Length; i++)
                    {
                        _rates[i] = value[i];
                    }
                }

                if (_mask != null)
                {
                    int j = 0;
                    for (int i = 0; i < _startDates.Length; i++)
                    {
                        if (_mask[i] == 0)
                        {
                            _timeMasked[j] = _timeToMaturity[i];
                            _rateMasked[j] = _rates[i];

                            j++;
                        }
                    }
```

```
                    _interpolator.SetAxis(_timeMasked, _rateMasked);
                }
                else
                {
                    _interpolator.SetAxis(_timeToMaturity, _rates);
                }
            }
        }
        #endregion

        private Curve(string Name, Date RefDate, string DayCounter, string Interpolation, string Compounding, int Frequency = 1, string Calendar = "BRL", double[] Mask = null)
        {
            _name = Name;
            _refDate = RefDate;

            _id = new Tuple<string, Date>(_name, _refDate);

            _calendar = CalendarFactory.Instance.GetCalendar(Calendar);
            _dayCounter = QLNet.DayCounter.GetDayCounter(DayCounter, _calendar);
            _interpolator = Interpolator.GetInterpolator(Interpolation);
            _compounding = SIRI.Pricer.Compounding.GetCompounder(Compounding, _dayCounter);

            if (Compounding.ToLower().Equals("compounded"))
            {
                ((CompoundedCompounding)_compounding).Frequency = (Frequency)Frequency;
            }
        }

        public Curve(string Name, int RefDate, double[] StartDates, double[] EndDates, string DayCounter, string Interpolation, string Compounding, int Frequency = 1, string Calendar = "BRL", double[] Mask = null)
```

```
            : this(Name, new Date(RefDate), DayCounter, Interpolation, Compounding, Frequency, Calendar, Mask)
        {
            StartDates = StartDates.TrimDouble();
            EndDates = EndDates.TrimDouble();

            if (StartDates.Length != EndDates.Length)
            {
                throw new ArgumentException("Start dates and end dates arrays must have same size.");
            }

            _numOfUnmaskedItems = StartDates.Length;
            if (Mask != null)
            {
                _mask = Mask;
                _numOfUnmaskedItems = 0;
            }
            else
            {
                _lastDate = new Date((int)EndDates[_numOfUnmaskedItems - 1]);
            }

            _startDates = new Date[StartDates.Length];
            _endDates = new Date[StartDates.Length];
            _timeToMaturity = new double[StartDates.Length];
            for (int i = 0; i < StartDates.Length; i++)
            {
                _startDates[i] = new Date((int)StartDates[i]);
                _endDates[i] = new Date((int)EndDates[i]);
                _timeToMaturity[i] = _dayCounter.dayCount(_refDate, _endDates[i]);

                if (_mask != null && _mask[i] == 0)
                {
                    _numOfUnmaskedItems++;
                    _lastDate = _endDates[i];
                }
            }

            _timeMasked = new double[_numOfUnmaskedItems];
```

```
            _rateMasked = new double[_numOfUnmaskedItems];
        }

        public Curve(string Name, Date RefDate, Date[] StartDates, Date[] EndDates, string DayCounter, string Interpolation, string Compounding, int Frequency = 1, string Calendar = "BRL", double[] Mask = null)
            : this(Name, RefDate, DayCounter, Interpolation, Compounding, Frequency, Calendar, Mask)
        {
            if (StartDates.Length != EndDates.Length)
            {
                throw new ArgumentException("Start dates and end dates arrays must have same size.");
            }

            _numOfUnmaskedItems = StartDates.Length;
            if (Mask != null)
            {
                _mask = Mask;
                _numOfUnmaskedItems = 0;
            }
            else
            {
                _lastDate = EndDates[_numOfUnmaskedItems - 1];
            }

            _startDates = new Date[StartDates.Length];
            _endDates = new Date[StartDates.Length];
            _timeToMaturity = new double[StartDates.Length];
            for (int i = 0; i < StartDates.Length; i++)
            {
                _startDates[i] = new Date(StartDates[i]);
                _endDates[i] = new Date(EndDates[i]);
                _timeToMaturity[i] = _dayCounter.dayCount(_refDate, _endDates[i]);

                if (_mask != null && _mask[i] == 0)
                {
                    _numOfUnmaskedItems++;
                    _lastDate = _endDates[i];
                }
```

```
            }

            _timeMasked = new double[_numOfUnmaskedItems];
            _rateMasked = new double[_numOfUnmaskedItems];
        }

        public Curve(string Name, Curve c)
        {
            _name = Name;
            _refDate = c._refDate;

            _id = new Tuple<string, Date>(_name, _refDate);

            UpdateFields(c);
        }

        public override double GetCurveRate(Date StartDate, Date EndDate, string ReturnType = "rate")
        {
            if (StartDate > EndDate)
            {
                throw new ArgumentException("StartDate must be before EndDate.");
            }

            double x2 = _dayCounter.dayCount(_refDate, EndDate);
            double y2 = GetCurveRate(EndDate);

            double y = -1;
            if (_refDate == StartDate)
            {
                y = y2;
            }
            else
            {
                double x1 = _dayCounter.dayCount(_refDate, StartDate);
                double y1 = GetCurveRate(StartDate);

                double df1 = _compounding.DiscountFactor(y1, _refDate, StartDate);
```

```
                double df2 = _compounding.DiscountFactor(y2, _refDate, EndDate);

                double df = df2 / df1;
                y = _compounding.ImpliedRate(1 / df, StartDate, EndDate);
            }

            double ret = 0;
            switch (ReturnType.ToLower())
            {
                case "capfct":
                case "cf":
                    ret = _compounding.CompoundFactor(y, StartDate, EndDate);
                    break;
                case "dfct":
                case "df":
                    ret = _compounding.DiscountFactor(y, StartDate, EndDate);
                    break;
                case "rate":
                    ret = y;
                    break;
                case "dv01fct":
                    ret = _compounding.DV01(y, StartDate, EndDate);;
                    break;
                default:
                    throw new ArgumentException("return type not supported: " + ReturnType);
            }

            return ret;
        }

        public override double GetCurveRate(Date EndDate, string ReturnType = "rate")
        {
            double x = _dayCounter.dayCount(_refDate, EndDate);
            double y = _interpolator.Interpolate(x);

```

```
            double ret = 0;
            switch (ReturnType.ToLower())
            {
                case "capfct":
                case "cf":
                    ret = _compounding.CompoundFactor(y, _refDate, EndDate);
                    break;
                case "df":
                case "dfct":
                    ret = _compounding.DiscountFactor(y, _refDate, EndDate);
                    break;
                case "rate":
                    ret = y;
                    break;
                case "dv01fct":
                    ret = _compounding.DV01(y, _refDate, EndDate); ;
                    break;
                default:
                    throw new ArgumentException("return type not supported: " + ReturnType);
            }
            return ret;
        }

        public override void UpdateFields(AbstractCurve Base)
        {
            Curve c = Base as Curve;
            if (c == null)
            {
                throw new ArgumentException();
            }

            _lastDate = c._lastDate;
            _calendar = c._calendar;
            _dayCounter = c.DayCounter;
            _interpolator = c._interpolator;
            _compounding = c._compounding;

            _startDates = c._startDates;
```

```
                _endDates = c._endDates;
                _timeToMaturity = c._timeToMaturity;
                _rates = c._rates;
                _mask = c._mask;
                _numOfUnmaskedItems = c._numOfUnmaskedItems;
                _timeMasked = c._timeMasked;
                _rateMasked = c._rateMasked;
            }
        }
}
```

APÊNDICE

B

SPLINE CÚBICO

Código-fonte 2 – Implementação de um *spline* cúbico natural seguindo o algortimo proposto por Burden e Faires (2010).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace SIRI.Mathematics.Interpolation
{
    public class CubicSplineInterpolator : Interpolator
    {
        private double[] _b;
        private double[] _c;
        private double[] _d;

        public CubicSplineInterpolator(bool Extrapolate = false)
            : base(Extrapolate)
        {
            _name = "NaturalCubicSpline";
        }

        /// <summary>
        /// Algorithm 3.4 of:
        ///     Numerical Analysis, Ninth Edition (Richard L. Burden)
        /// </summary>
```

```
        public override void update()
        {
            int n = _size - 1;

            _b = new double[n];
            _c = new double[n+1];
            _d = new double[n];

            double[] h = new double[n];
            for (int i = 0; i <= n-1; i++)
            {
                h[i] = _xAxis[i + 1] - _xAxis[i];
            }

            double[] alpha = new double[n];
            for (int i = 1; i <= n-1; i++)
            {
                alpha[i] = (3 / h[i]) * (_yAxis[i + 1] - _yAxis
    [i]) - (3 / h[i-1]) * (_yAxis[i] - _yAxis[i-1]);
            }

            double[] l = new double[n+1];
            double[] mi = new double[n];
            double[] z = new double[n+1];

            l[0] = 1;
            mi[0] = 0;
            z[0] = 0;

            for (int i = 1; i <= n-1; i++)
            {
                l[i] = 2 * (_xAxis[i + 1] - _xAxis[i - 1]) - h[
    i - 1] * mi[i - 1];
                mi[i] = h[i] / l[i];
                z[i] = (alpha[i] - h[i - 1] * z[i - 1]) / l[i];
            }

            l[n] = 1;
            z[n] = 0;
            _c[n] = 0;

            for (int j = n - 1; j >= 0; j--)
```

```
            {
                _c[j] = z[j] - (mi[j] * _c[j + 1]);
                _b[j] = (_yAxis[j + 1] - _yAxis[j]) / h[j] - h[j] * (_c[j + 1] + 2 * _c[j]) / 3;
                _d[j] = (_c[j + 1] - _c[j]) / (3 * h[j]);
            }
        }

        protected override double Eval(double X, int i)
        {
            double dx = X - _xAxis[i];
            return _yAxis[i] + _b[i] * dx + _c[i] * (dx * dx) + _d[i] * (dx * dx * dx);
        }
    }
}
```

APÊNDICE

C

CLASSE LVOLSURFACE

Código-fonte 3 – Implementação da superfície de volatilidade

```
using System;
using System.Collections.Generic;
using System.Linq;

using QLNet;
using SIRI.Mathematics.Interpolation;

namespace SIRI.Pricer
{
    public class LVolSurface
    {
        #region Members
        protected string _name;
        protected Date _refDate;

        protected Date[] _dates;

        protected double[] _deltasOrStrikes;
        protected double[] _timeToMaturity;
        protected double[,] _vols;

        protected Calendar2 _calendar;
        protected DayCounter _dayCounter;

        protected Interpolator[] _interpolators;
        #endregion
```

```

        #region Properties
        public string Name { get { return _name; } }
        public Date RefDate { get { return _refDate; } }

        public double[,] Vols
        {
            get { return _vols; }
            set
            {
                if (value.GetLength(0) != _dates.Length || value.GetLength(1) != _deltasOrStrikes.Length)
                {
                    throw new ArgumentException("Vols matrix rows must be equals to dates array length and columns equals to deltas or strikes array length.");
                }

                _vols = value;
                for (int i = 0; i < _dates.Length; i++)
                {
                    _interpolators[i].SetAxis(_deltasOrStrikes, _vols.SliceRow<double>(i).ToArray());
                }
            }
        }
        #endregion

        public LVolSurface(string Name, int RefDate, double[] Dates, double[] DeltasOrStrikes, string DayCounter, string Calendar = "USD", string Interpolation = "CubicSpline")
        {
            _name = Name;
            _refDate = RefDate;

            _calendar = CalendarFactory.Instance.GetCalendar(Calendar);
            _dayCounter = QLNet.DayCounter.GetDayCounter(DayCounter, _calendar);

            _deltasOrStrikes = DeltasOrStrikes.TrimDouble();

```

```
            Dates = Dates.TrimDouble();
            _dates = new Date[Dates.Length];
            _interpolators = new Interpolator[Dates.Length];
            _timeToMaturity = new double[Dates.Length];
            for (int i = 0; i < Dates.Length; i++)
            {
                _dates[i] = new Date((int)Dates[i]);
                _interpolators[i] = Interpolator.GetInterpolator(Interpolation);
                _interpolators[i].Extrapolate = true;
                _timeToMaturity[i] = _dayCounter.dayCount(_refDate, _dates[i]);
            }
        }

        public double GetVol(int EndDate, double DeltaOrStrike)
        {
            double timeToMaturity = _dayCounter.dayCount(_refDate, EndDate);
            int x = _timeToMaturity.Locate(timeToMaturity);

            double volT = 0;
            if (EndDate == _dates[x])
            {
                volT = _interpolators[x].Interpolate(DeltaOrStrike);
            }
            else
            {
                double t = _dayCounter.yearFraction(_refDate, EndDate);
                double t1 = _dayCounter.yearFraction(_refDate, _dates[x]);
                double t2 = _dayCounter.yearFraction(_refDate, _dates[x + 1]);

                double volT1 = _interpolators[x].Interpolate(DeltaOrStrike);
                double volT2 = _interpolators[x + 1].Interpolate(DeltaOrStrike);

                double varianceT1 = volT1 * volT1;
```

```
                    double varianceT2 = volT2 * volT2;

                    double varianceT = ((1 / (t2 - t1)) * (
    varianceT2 * t2 * (t - t1) + varianceT1 * t1 * (t2 - t))) /
    t;
                    volT = Math.Sqrt(varianceT);
                }

                return volT;
            }
    }
}
```

APÊNDICE D

CLASSE MARKETVOLSURFACE

Código-fonte 4 – Implementação da superfície de volatilidade pelos parâmetros ATM, RR e STR

```
using System;
using System.Collections.Generic;
using System.Linq;

using QLNet;
using SIRI.Mathematics.Interpolation;

namespace SIRI.Pricer
{
    public class MarketVolSurface : LVolSurface
    {
        private static readonly double[] DELTAS = { 0.1, 0.25, 0.50, 0.75, 0.9 };

        private readonly double STR_RR_FACTOR = 2.25;

        #region Members
        private double[] _atm;
        private double[] _rr10;
        private double[] _rr25;
        private double[] _str10;
        private double[] _str25;

        private double _atm3M;

```

```
        private double[] _shifts;

        private double _atmBump;
        private double _rrBump;
        private double _strBump;
        #endregion

        public MarketVolSurface(string Name, int RefDate, double[] Dates, string DayCounter, string Calendar = "USD", string Interpolation = "CubicSpline")
            : base(Name, RefDate, Dates, DELTAS, DayCounter, Calendar, Interpolation)
        {
            CalculateShifts();
        }

        public MarketVolSurface(string Name, Date RefDate, Date[] Dates, string DayCounter, string Calendar = "USD", string Interpolation = "CubicSpline")
            : base(Name, RefDate, Dates, DELTAS, DayCounter, Calendar, Interpolation)
        {
            CalculateShifts();
        }

        private void CalculateShifts()
        {
            _shifts = new double[_dates.Length];
            for (int i = 0; i < _dates.Length; i++)
            {
                double days = (double)(_dates[i] - _refDate);
                _shifts[i] = Math.Sqrt(90D / days);
            }
        }

        public void SetVols(double[] ATM, double[] RR10, double[] RR25, double[] STR10, double[] STR25, double ATM3M)
        {
            _atm = ATM;
            _rr10 = RR10;
            _rr25 = RR25;
            _str10 = STR10;
```

```
            _str25 = STR25;

            _atm3M = ATM3M;

            UpdateSurfaceVols();
        }

        public void SetBumps(double ATMBump, double RRBump, double STRBump)
        {
            _atmBump = ATMBump;
            _rrBump = RRBump;
            _strBump = STRBump;

            if (_rrBump != 0)
            {
                _strBump += _rrBump / ((_atm3M + ATMBump) * 100) * STR_RR_FACTOR;
            }

            UpdateSurfaceVols();
        }

        public void UpdateSurfaceVols()
        {
            double[,] vols = new double[_atm.Length, 5];
            for (int i = 0; i < _atm.Length; i++)
            {
                double atm = _atm[i] + _atmBump * _shifts[i];
                double rr25 = _rr25[i] + _rrBump * _shifts[i];
                double rr10 = rr25 * (_rr10[i] / _rr25[i]);
                double str25 = _str25[i] + _strBump * _shifts[i];
                double str10 = str25 * (_str10[i] / _str25[i]);

                vols[i, 0] = atm + str10 + 0.5 * rr10;  // 10% delta
                vols[i, 1] = atm + str25 + 0.5 * rr25;  // 25% delta
                vols[i, 2] = atm;                       // 50% delta
```

```
                vols[i, 3] = atm + str25 - 0.5 * rr25;  // 75% delta
                vols[i, 4] = atm + str10 - 0.5 * rr10;  // 90% delta
            }

            base.Vols = vols;
        }
    }
}
```

APÊNDICE

E

CLASSE BLACK76

Código-fonte 5 – Implementação do Black and Scholes

```
using SIRI.Mathematics.Statistics;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace SIRI.Pricer.Engine
{
    public class BlackScholes : IOptionPricingEngine
    {
        protected double _spot;
        private double _strike;
        private double _vol;
        private double _tVol;
        private double _rate;
        private double _tDiscount;
        private string _callPut;

        private double _discountFactor;

        private double _d1;
        private double _d2;

        private double _sqrtT;

        public double Spot
```

```
        {
            get { return _spot; }
            set
            {
                _spot = value;
                Update();
            }
        }

        public double Vol
        {
            get { return _vol; }
            set
            {
                _vol = value;
                Update();
            }
        }

        public BlackScholes(double Spot, double Strike, double TVol, double ContinuousRate, double TDiscount, string CallPut)
        {
            _spot = Spot;
            _strike = Strike;
            _tVol = TVol;
            _rate = ContinuousRate;
            _tDiscount = TDiscount;
            _callPut = CallPut;

            _discountFactor = Math.Exp(-_rate * _tDiscount);

            _sqrtT = Math.Sqrt(_tVol);
        }

        public virtual double Premium()
        {
            double premium = 0;
            switch (_callPut.ToLower())
            {
                case "c":
                case "call":
```

```
                    premium = _spot * NormalDistribution.Phi(_d1) - _strike * _discountFactor * NormalDistribution.Phi(_d2);
                    break;
                case "p":
                case "put":
                    premium = _strike * _discountFactor * NormalDistribution.Phi(-_d2) - _spot * NormalDistribution.Phi(-_d1);
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return premium;
        }

        public virtual double Delta()
        {
            double delta = 0;
            switch (_callPut.ToLower())
            {
                case "c":
                case "call":
                    delta = NormalDistribution.Phi(_d1);
                    break;
                case "p":
                case "put":
                    delta = NormalDistribution.Phi(_d1) - 1;
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return delta;
        }

        public virtual double Gamma()
        {
            double gamma = 0;
```

```
            switch (_callPut.ToLower())
            {
                case "c":
                case "call":
                case "p":
                case "put":
                    gamma = NormalDistribution.PDF(_d1) / (_spot * _vol * _sqrtT);
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return gamma;
        }

        public virtual double Vega()
        {
            double vega = 0;
            switch (_callPut.ToLower())
            {
                case "c":
                case "call":
                case "p":
                case "put":
                    vega = _spot * NormalDistribution.PDF(_d1) * _sqrtT;
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return vega;
        }

        public virtual double Theta()
        {
            double theta = 0;
            switch (_callPut.ToLower())
            {
```

```
                case "c":
                case "call":
                    theta = -(_spot * NormalDistribution.PDF(_d1) * _vol) / (2 * _sqrtT) - _rate * _strike * _discountFactor * NormalDistribution.Phi(_d2);
                    break;
                case "p":
                case "put":
                    theta = -(_spot * NormalDistribution.PDF(_d1) * _vol) / (2 * _sqrtT) + _rate * _strike * _discountFactor * NormalDistribution.Phi(-_d2);
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return theta;
        }

        public virtual double Rho()
        {
            double rho = 0;
            switch (_callPut.ToLower())
            {
                case "c":
                case "call":
                    rho = _tVol * _strike * _discountFactor * NormalDistribution.Phi(_d2);
                    break;
                case "p":
                case "put":
                    rho = -_tVol * _strike * _discountFactor * NormalDistribution.Phi(-_d2);
                    break;
                default:
                    throw new ArgumentException("CallPut parameter not valid.");
            }

            return rho;
        }
```

```

        private void Update()
        {
            double volSqrtT = _vol * _sqrtT;

            _d1 = (Math.Log(_spot / _strike) + (_rate + 0.5 *
    _vol * _vol) * _tVol) / volSqrtT;
            _d2 = _d1 - volSqrtT;
        }
    }
}
```

—