
Análise e Implementação das Técnicas de Neely para Identificação das Ondas de Elliott

Bruno Feitosa

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Bruno Feitosa

Análise e Implementação das Técnicas de Neely para Identificação das Ondas de Elliott

Monografia apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, para o Exame de Qualificação, como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional.

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Vanderlei Bonato

**USP – São Carlos
Agosto de 2019**

Bruno Feitosa

Analysis and Implementation of Neely Techniques for Elliott Wave Identification

Monograph submitted to the Instituto de Ciências Matemáticas e de Computação - ICMC-USP, as part of the qualifying exam requisites of the Master Joint Graduate Program in Computer Science and Computational Mathematics.

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Vanderlei Bonato

**USP – São Carlos
August 2019**

Resumo

Observação de preços em mercados sempre foi do interesse daqueles que negociavam produtos, pois a capacidade de interpretar o mercado e decidir a melhor hora de comprar ou vender poderia ditar a diferença entre ganhar ou perder dinheiro. Durante a Era Meiji, no Século 19, um mercador de arroz chamado Muneshima Homma desenvolveu uma técnica de visualização e interpretação do preço do arroz, que flutuava constantemente durante um dia de negociações. Ele se tornou capaz de reconhecer os momentos em que o preço estaria mudando entre baixas e altas, reconhecendo o melhor momento para negociar o produto.

O ambiente tecnológico atual, onde a computação e a Internet facilitaram a velocidade de transmissão e acesso a informação contribuiu para fomentar o crescimento de *commodities*, derivativos e opções. Porém, com as rápidas flutuações dos preços dos ativos negociados em bolsa de valores, causadas pelos inúmeros investidores negociando os ativos, a bolsa de valores não é um ambiente de fácil entrada para novos investidores. Diversos economistas e analistas se especializaram em observar, entender e prever os movimentos dos preços no mercado de ações através da análise fundamentalista ou gráfica com a ajuda de indicadores derivados de preços.

Entre os especialistas em análise gráfica, em 1938 se destacou Ralph Nelson Elliott com o seu trabalho: O Princípio das Ondas. Sua teoria era capaz de interpretar o estado do mercado e prever movimentos futuros com base nos padrões das Ondas observadas. Porém, sua teoria foi considerada subjetiva por alguns estudiosos, o que pode ter evitado a disseminação de seus estudos. Mais tarde, em 1987, Glen Neely publicou um método científico capaz de implementar a teoria de Elliott, através de um algoritmo que gera resultados objetivos e replicáveis.

O objetivo deste trabalho é implementar o algoritmo de Neely para análise do preço de ativos, e analisar como diferentes configurações do método alteram a análise resultante.

Lista de Figuras

1	Gráfico do Preço do Dólar (2018-02-05) em <i>candlesticks</i>	2
2	Preço do Dólar (2018-02-05) as 12:00 e 13:00	2
3	Gráfico do Preço do Dólar (2018-02-05) em OHLC	3
4	Gráfico do Preço do Dólar (2018-02-05)	4
5	Gráfico do Preço do Dólar (2018-02-05) com maior Resolução	4
6	Gráfico do Preço do Dólar (2018-02-05) comparando resoluções	5
7	Onda Padrão 5-3 (adaptado de [1])	6
8	Onda Padrão 5-3 estendida (adaptado de [1])	7
9	Visualização de uma Monowave	11
10	Fluxograma do Método Elliott-Neely	12
11	Movimento Direcional	13
12	Movimento Horizontal do Preço	14
13	Regra de Neutralidade em forma gráfica	15
14	Execução da Regra de Neutralidade em forma gráfica	16
15	Diagrama de Casos de Uso	21
16	Visão Lógica	28
17	Visão de Processo para UC-1.1: Importar CSV	29
18	Visão de Processo para UC-2.1: Analisar Ação	30
19	Visão de Processo para UC-3.1: Exibir Resultado	30
20	Visão de Processo para UC-4.1: Mudar Configuração	31
21	Visão de Implementação	32
22	Visão de Implantação	34
23	Padrão de Onda Motivadora: Impulso	40
24	Padrão de Onda Motivadora: Extensão	40
25	Padrão de Onda Motivadora: Truncagem	41
26	Padrão de Onda Motivadora: Diagonal Final/Inicial	41
27	Padrão de Onda Corretiva: Zigzag	41
28	Padrão de Onda Corretiva: Zigzag Duplo	42
29	Padrão de Onda Corretiva: Plano	42
30	Padrão de Onda Corretiva: Plano Estendido	42
31	Padrão de Onda Corretiva: Plano Corrido	43
32	Padrão de Onda Corretiva: Triângulo	43

33	Padrão de Onda Corretiva: Combinação (exemplo 1)	44
34	Padrão de Onda Corretiva: Combinação (exemplo 2)	44

Lista de Tabelas

1	Preços do Dólar (2018-02-05) em Tabela	5
2	Notações das Ondas de Elliott de acordo com o Grau da Onda	8
3	Padrões das Ondas de Elliott	8
4	Representação Quantitativa dos Interesses dos <i>Stakeholders</i>	19
5	UC-1.1 - Importar .CSV	22
6	UC-1.2 - Importar de Terceiros	23
7	UC-2.1 - Analisar Ação	24
8	UC-3.1 - Exibir Resultado	25
9	UC-4.1 - Mudar Configuração	26
10	Requisitos Funcionais	27
11	Decisão - Linguagem de Implementação	35
12	Decisão - Análise de Diferentes Configurações	36
13	Cronograma de atividades	37
14	Progresso do algoritmo	38

Sumário

Resumo	i
Lista de Figuras	ii
Lista de Tabelas	iv
Sumário	v
1 Introdução	1
1.1 Contextualização: Mercado Financeiro e Negociação de Ações	1
1.2 Representações Gráficas: OHLC (<i>Open-High-Low-Close</i>)	1
2 Análise Gráfica	6
2.1 Ondas de Elliott	6
2.1.1 Regras para Formação das Ondas de Elliott	8
2.1.2 Críticas à Teoria de Elliott	10
2.2 Extensões de Neely	10
2.2.1 Limitação de Janela de Análise	13
2.2.2 Fatores de Estudo	16
3 Arquitetura	18
3.1 Contexto do Sistema	18
3.2 Requisitos	18
3.2.1 Visão arquitetural	18
3.2.2 <i>Stakeholders</i>	18
3.2.3 Atributos de Qualidade	19
3.2.4 Requisitos de Alto-Nível	19
3.2.5 Casos de Uso	20
3.2.6 Requisitos Funcionais	26
3.3 Análise	27
3.3.1 Suposições	27
3.3.2 Roteiro tecnológico	27
3.4 Arquitetura do Software	28
3.4.1 Visão Lógica	28

3.4.2	Visão de Processo	29
3.4.3	Visão de Implementação	31
3.4.4	Visão de Implantação	34
3.4.5	Decisões de Desenvolvimento de Software	34
4	Cronograma e Progressão	37
	Referências	39
	Apêndices	40
A	Padrões das Ondas de Elliott	40

1. Introdução

1.1. Contextualização: Mercado Financeiro e Negociação de Ações

Stringham e Curott em [2] pesquisaram e descobriram que negócios com múltiplos acionistas se tornaram populares com contratos *commenda* na Itália Medieval [3], e Malmendier [4] proveu evidências que empresas acionistas remontam à Roma antiga. No entanto, Stringham e Curott identificaram que o primeiro mercado de ações do mundo nasceu em Amsterdã do século 17, onde um mercado secundário ativo de ações emergiu. As duas maiores empresas eram a Dutch East India Company e a Dutch West India Company, fundadas em 1602 e 1621.

Com a fundação do primeiro mercado de ações eletrônico, a NASDAQ, em 1971, e a popularização da computação e Internet durante as últimas décadas, investir no mercado financeiro se torna cada vez mais fácil. Porém, preços de ativos no mercado financeiro flutuam de acordo com as negociações feitas no próprio mercado. A volatilidade deste tipo de negócio implica que, para se obter sucesso neste tipo de empreendimento, é necessário saber observar e entender as tendências dos preços e como eles refletem o otimismo e pessimismo dos negociadores.

1.2. Representações Gráficas: OHLC (*Open-High-Low-Close*)

De acordo com a pesquisa de Lo e Hasanhodzic [5], o primeiro registro de preços com o objetivo de predizê-los ocorreu na Babilônia, durante o primeiro milênio a.C.. Segundo Shimizu [6] “a abertura de trocas de *commodities* na Europa Ocidental (1561) e Japão (1654) proveram o ambiente necessário para o desenvolvimento do gráfico”. Conforme Kirkpatrick e Dahlquist [7], no momento em que estas negociações aconteciam, os mercados haviam se tornado sofisticados o bastante para produzirem múltiplos preços durante o decorrer do dia, e criando portanto, a necessidade de se registrar os preços de abertura, máximo, mínimo e de fechamento dos *commodities* sendo negociados.

Atualmente, a forma mais comum de representar o histórico do preço de uma ação é observando a dinâmica dos preços negociados através dos valores de abertura, máximo, mínimo, e fechamento de um dado período. Os quatro valores podem ser condensados em um único marcador gráfico, sendo os dois mais comuns as “velas japonesas” (*candelstick*), e o OHLC (*Open-High-Low-Close*, literalmente Abertura-Máximo-Mínimo-Fechamento). O marcador OHLC também é conhecido como gráfico de barras. O marcador OHLC divide seu nome com os tipos de dados de negociações de ativos.

Os valores podem ser observados durante diferentes períodos, como um dia ou um minuto. Na Figura 1 podem ser vistos os preços da cotação do Dólar (US\$) em Reais (R\$) durante o dia 5 de Fevereiro de 2018, utilizando as marcações de *candlesticks* com intervalos de tempo de 1 hora.

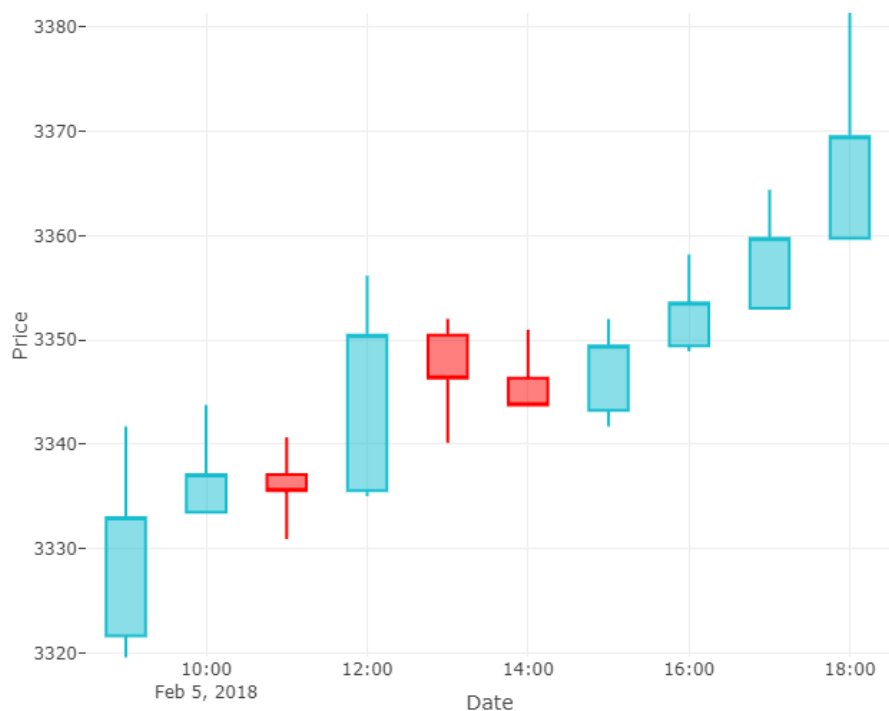


Figura 1. Gráfico do Preço do Dólar (2018-02-05) em *candlesticks*

Cada “vela” representa um conjunto de quatro valores: abertura (*open*), máximo (*high*), mínimo (*low*) e fechamento (*close*). A extensão do “pavio” da vela indica os valores de máximo (extensão superior) e mínimo (extensão inferior), e os valores de abertura e fechamento são indicados pelo “corpo” da vela. Os preços observados as 12:00 e 13:00 podem ser vistos na Figura 2.

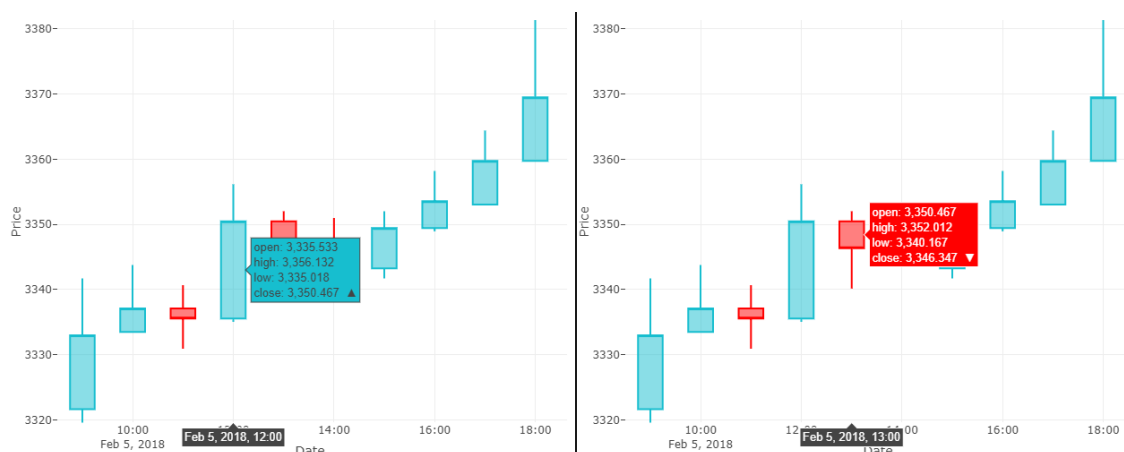


Figura 2. Preço do Dólar (2018-02-05) as 12:00 e 13:00

É notável observar que as 12:00 o preço aumenta durante o período (preço de fechamento é maior que o preço de abertura), enquanto as 13:00 ocorre o contrário (preço de fechamento é menor que o preço de abertura). Preços crescentes e decrescentes são distinguidos por cores (e.g. azul para crescente, vermelho para decrescente) ou por vela

sem cor interna (crescente) ou com cor cheia (decrecente) em representações monocromáticas. Essa distinção é necessária, caso contrário não é possível observar no corpo da vela se o preço de abertura/fechamento ocorreu no topo ou na base do corpo da vela.

O mesmo histórico de preço mostrado na Figura 1 pode ser visto na Figura 3 a seguir utilizando marcadores OHLC.

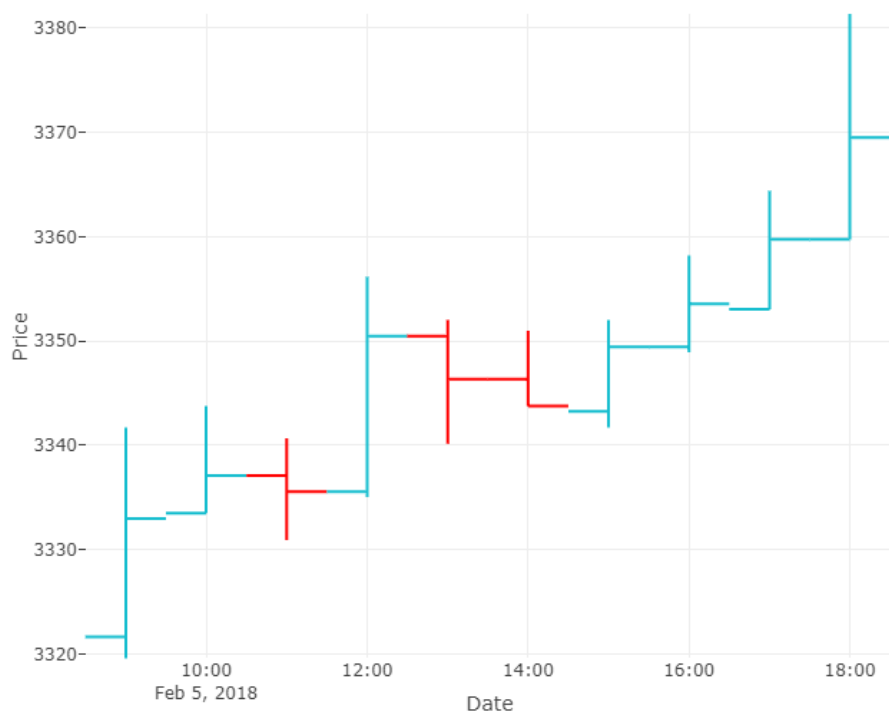


Figura 3. Gráfico do Preço do Dólar (2018-02-05) em OHLC

Na representação OHLC, o traço vertical indica os valores de máximo e mínimo, a extensão a esquerda indica o valor de abertura, e a extensão a direita indica o valor de fechamento. Em contrapartida às *candlesticks* não é necessário uma representação em cores para distinguir abertura de fechamento no indicador, porém, a visualização toma menos volume no gráfico, o que pode dificultar a leitura ao observador.

Os dois marcadores podem ser vistos lado a lado na Figura 4 para efeito de comparação.

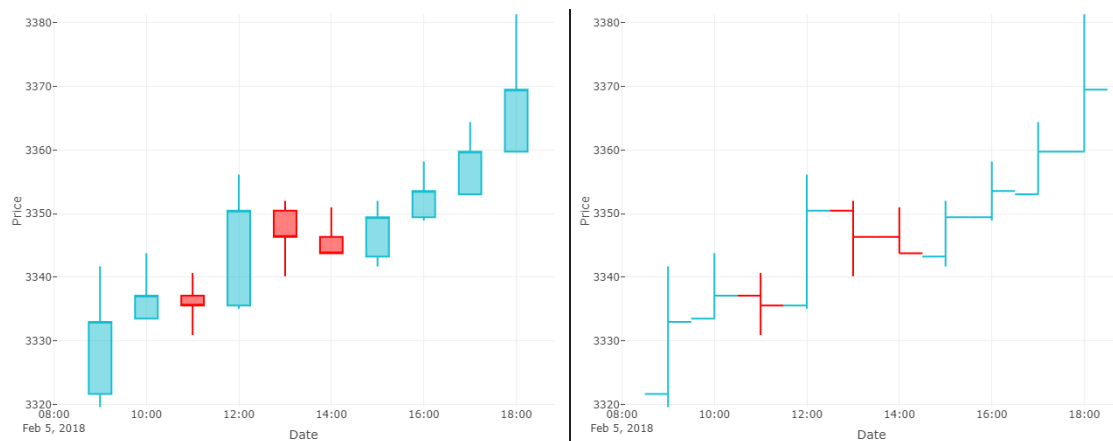


Figura 4. Gráfico do Preço do Dólar (2018-02-05)

É importante ressaltar também que gráficos de preços de ativos procuram representar a dinâmica do preço durante um período, já que o preço de um ativo muda constantemente. Os preços do Dólar observados na Figura 3 representam um período de 1 hora para cada ponto no gráfico. Os valores do Dólar para o mesmo dia, porém com período de 1 minuto, podem ser observados na Figura 5 a seguir.



Figura 5. Gráfico do Preço do Dólar (2018-02-05) com maior Resolução

Apesar de serem observados os mesmos preços no mesmo dia, a resolução pode mudar completamente a interpretação do gráfico. Uma comparação superimposta da Figura 1 e da Figura 5 pode ser vista na Figura 6 a seguir.

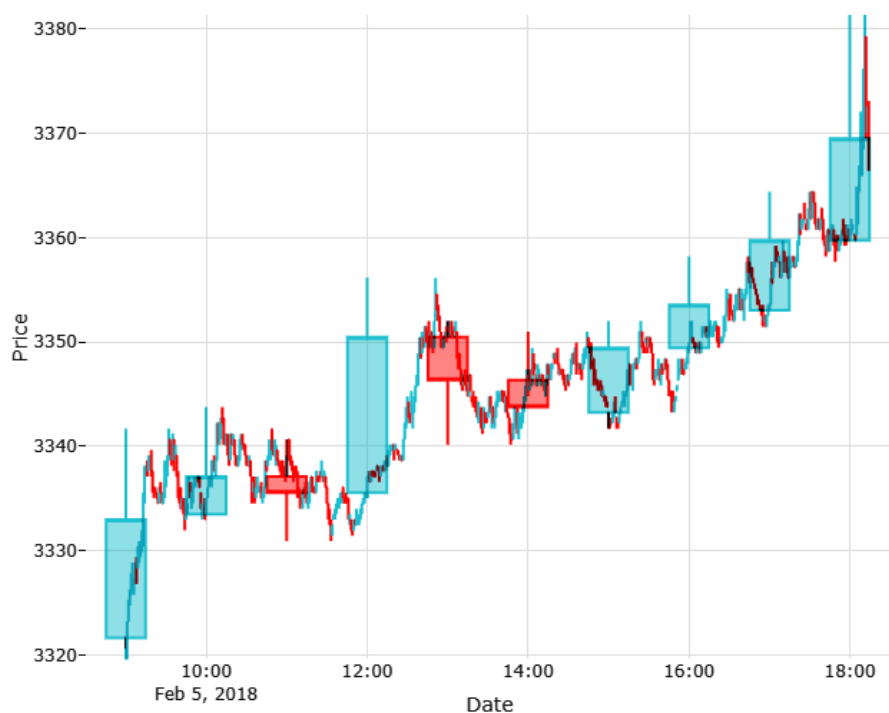


Figura 6. Gráfico do Preço do Dólar (2018-02-05) comparando resoluções

Os preços do Dólar utilizados nos gráficos desta seção podem ser vistos na Tabela 1 a seguir. Estes preços foram exportados do programa MetaTrader 5. Os gráficos desta seção foram gerados utilizando o programa em desenvolvimento neste projeto de pesquisa.

Tabela 1. Preços do Dólar (2018-02-05) em Tabela

Data	Abertura	Máximo	Mínimo	Fechamento
2018/02/05 9:00	3,321628	3,341712	3,319568	3,332958
2018/02/05 10:00	3,333473	3,343772	3,333473	3,337078
2018/02/05 11:00	3,337078	3,340682	3,330898	3,335533
2018/02/05 12:00	3,335533	3,356132	3,335018	3,350467
2018/02/05 13:00	3,350467	3,352012	3,340167	3,346347
2018/02/05 14:00	3,346347	3,350982	3,343772	3,343772
2018/02/05 15:00	3,343257	3,352012	3,341712	3,349437
2018/02/05 16:00	3,349437	3,358192	3,348922	3,353557
2018/02/05 17:00	3,353042	3,364372	3,353042	3,359737
2018/02/05 18:00	3,353042	3,381366	3,353042	3,369521

2. Análise Gráfica

2.1. Ondas de Elliott

A Teoria das Ondas de Elliott pode ser explicada em alguns princípios baseados nos fenômenos que ele observou nos preços. Estes princípios são:

- Preços no mercado de ações seguem padrões reconhecíveis;
- Estes padrões se repetem em “forma”, mas não necessariamente em “amplitude” e “tempo”;
- Estes padrões são chamados de **Ondas (Waves)**;
- Uma onda pode ser classificada como sendo **Motivadora/Motriz** ou **Corretiva**;
 - Ondas **Motivadoras (Motive)** seguem a **mesma tendência** do mercado;
 - Ondas **Corretiva (Corrective)** seguem a **tendência contrária** a do mercado;
- As **Ondas** podem ser combinadas entre si para formarem **novas Ondas** em **resoluções de tempo** diferentes (visão fractal);

A onda idealizada observada por Elliott foi o padrão de Onda 5-3, visto na Figura 7. As Ondas de Elliott motivadoras são identificadas com números e as corretivas são identificadas com letras.

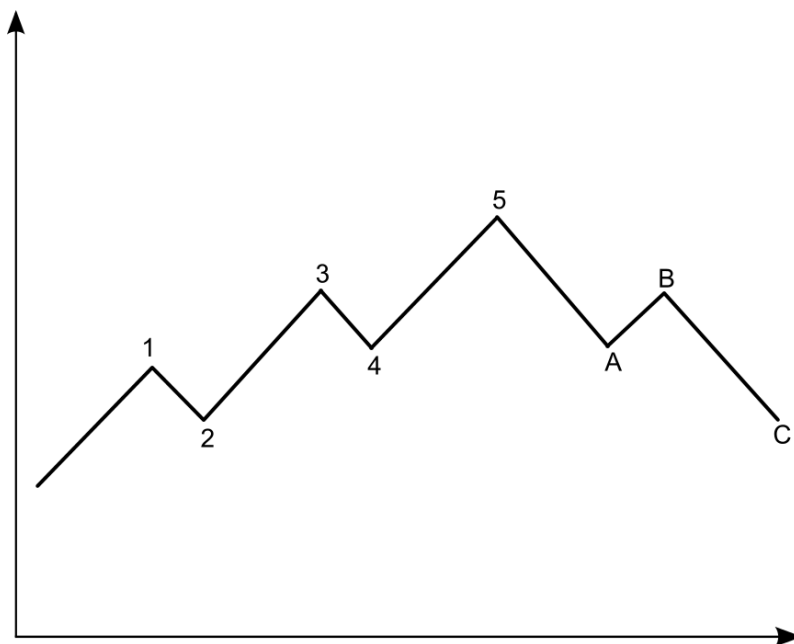


Figura 7. Onda Padrão 5-3 (adaptado de [1])

A tendência principal do mercado é que um Impulso (Onda 1-2-3-4-5 na Figura 7) sempre é seguido de uma Retração (Onda A-B-C), formando o padrão 5-3, onde há 5 Ondas para o impulso e 3 Ondas para a retração. Adicionalmente, como as Ondas podem ser combinadas entre si para formarem novas Ondas, o padrão de ondas 5-3 também poderia ser observado como na Figura 8.

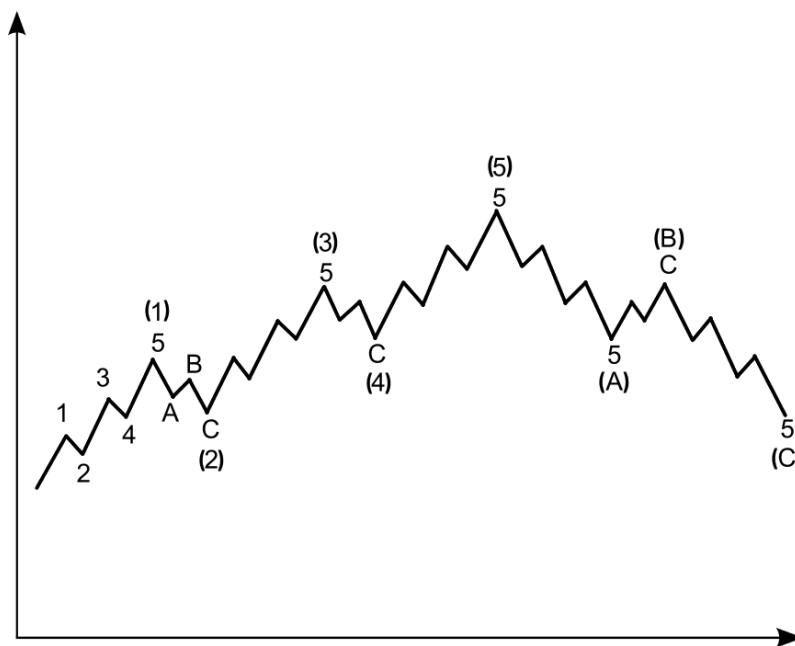


Figura 8. Onda Padrão 5-3 estendida (adaptado de [1])

Quando um conjunto de Ondas forma uma nova onda em uma resolução de tempo diferente, é dito que as Ondas formam uma onda de grau superior. Dessa forma, se torna necessária uma regra de identificação para Ondas de acordo com o grau (resolução do tempo) na qual elas se encontram. As marcações utilizadas para cada grau podem ser vistas na Tabela 2 a seguir. Os “5’s Seguindo a Tendência” são como são identificadas as ondas motivadoras (usando números), e os “3’s Contra a Tendência” são como são identificadas as ondas corretivas (usando letras).

Tabela 2. Notações das Ondas de Elliott de acordo com o Grau da Onda

Grau da Onda		5's Seguindo a Tendência					3's Contra a Tendência		
1	Supermilênio	①	②	③	④	⑤	Ⓐ	Ⓑ	Ⓒ
2	Milênio	(1)	(2)	(3)	(4)	(5)	(A)	(B)	(C)
3	Submilênio	1	2	3	4	5	A	B	C
4	SuperCiclo Maior	Ⓘ	Ⓜ	Ⓢ	Ⓣ	Ⓥ	ⓐ	ⓑ	ⓒ
5	SuperCiclo	(I)	(II)	(III)	(IV)	(V)	(a)	(b)	(c)
6	Ciclo	I	II	III	IV	V	a	b	c
7	Primário	①	②	③	④	⑤	Ⓐ	Ⓑ	Ⓒ
8	Intermediário	(1)	(2)	(3)	(4)	(5)	(A)	(B)	(C)
9	Menor	1	2	3	4	5	A	B	C
10	Minuto	⓪	Ⓛ	Ⓜ	Ⓨ	Ⓥ	ⓐ	ⓑ	ⓒ
11	Minuete	(i)	(ii)	(iii)	(iv)	(v)	(a)	(b)	(c)
12	SubMinuete	i	ii	iii	iv	v	a	b	c
13	Micro	①	②	③	④	⑤	Ⓐ	Ⓑ	Ⓒ
14	Submicro	(1)	(2)	(3)	(4)	(5)	(A)	(B)	(C)
15	Minúsculo	1	2	3	4	5	A	B	C

Os padrões identificados por Elliott podem ser vistos na Tabela 3, onde cada padrão pode ser visto na Figura de referência no Apêndice A. As regras para formação de cada padrão podem ser vistas na Seção 2.1.1 a seguir.

Tabela 3. Padrões das Ondas de Elliott

	Padrão	Nome Original	Referência
Ondas Motivadoras	Impulso	<i>Impulse</i>	Figura 23
	Extensão	<i>Extension</i>	Figura 24
	Truncagem	<i>Truncation</i>	Figura 25
	Diagonal Final/Inicial	<i>Ending/Leading Diagonal</i>	Figura 26
Ondas Corretivas	Zigzag	<i>Zigzag</i>	Figura 27
	Zigzag Duplo	<i>Double Zigzag</i>	Figura 28
	Plano	<i>Flat</i>	Figura 29
	Plano Estendido	<i>Extended Flat</i>	Figura 30
	Plano Corrido	<i>Running Flat</i>	Figura 31
	Triângulo	<i>Triangle</i>	Figura 32
	Combinação (3 Duplo e Triplo)	<i>Double/Triple 3</i>	Figuras 33 e 34

2.1.1. Regras para Formação das Ondas de Elliott

- Impulso
 - O Impulso sempre se divide em 5 ondas
 - Onda 2 nunca se move além do início da Onda 1
 - Onda 3 sempre se move além do final da Onda 1

- Onda 4 nunca se move além do fim da Onda 1
- Nunca todas as Ondas 1, 3 e 5 estão todas Estendidas
- Onda 1 sempre subdivide em Impulso ou (raramente) Diagonal
- Onda 3 sempre subdivide em Impulso
- Onda 5 sempre subdivide em Impulso ou Diagonal
- Onda 2 sempre subdivide em Zigzag
- Onda 4 sempre subdivide em Zigzag, Triângulo ou Combinação
- Diagonal
 - A Diagonal sempre se subdivide em 5 Ondas
 - A Diagonal Final sempre é a onda 5 de um Impulso, ou Onda C de um Zigzag ou Plano
 - A Diagonal Inicial sempre aparece como Onda 1 de um Impulso ou Onda A de um Zigzag
 - Ondas 1, 2, 3, 4, e 5 de uma Diagonal Final, e Ondas 2 e 4 de uma Diagonal Inicial, sempre se subdividem em Zigzags
 - Onda 2 nunca passa do início da Onda 1
 - Onda 3 sempre passa do fim da Onda 1
 - Onda 4 nunca passa do final da Onda 2
 - Onda 4 sempre termina dentro do território de preço da Onda 1
 - Em relação ao tempo, uma linha conectando o fim das Ondas 2 e 4 convergem em direção (caso de contração) ou divergem (caso de expansão) de uma linha conectando os fins das Ondas 1 e 3
 - A Onda 5 de uma Diagonal Inicial sempre termina além do final da Onda 3
 - Na Diagonal com Contração, a Onda 3 é sempre mais curta que a Onda 1, e a Onda 5 é sempre mais curta que a Onda 3
 - Na Diagonal com Expansão, a Onda 3 é sempre mais longa que a Onda 1, e a Onda 5 é sempre mais longa que a Onda 3
- Zigzag
 - A Zigzag sempre se subdivide em 3 Ondas
 - A Onda A sempre se subdivide em um Impulso ou em uma Diagonal Inicial
 - A Onda C sempre se subdivide em um Impulso ou em uma Diagonal
 - A Onda B sempre se subdivide em um Zigzag, Plano, Triângulo ou Combinação destes
 - A Onda B nunca se move além do início da Onda A
- Plano
 - Um Plano sempre se subdivide em 3 Ondas
 - Onda A nunca é um Triângulo
 - Onda C sempre é um Impulso ou um Triângulo
 - Onda B sempre regride/retraça pelo menos 90% da Onda A
- Triângulo
 - Contracionado
 - * Um Triângulo sempre se subdivide em 5 Ondas
 - * Pelo menos 4 ondas entre A, B, C, D e E se subdividem em um Zigzag ou Combinação de Zigzags
 - * A Onda C nunca se move além do final da Onda A

- * A Onda D nunca se move além do final da Onda B
- * A Onda E nunca se move além do final da Onda C
- * Consequência das 3 regras anteriores: uma linha ligando os fins das ondas B e D convergem e relação a uma linha ligando os fins das A e C
- * Um Triângulo nunca tem mais de uma sub-Onda complexa, a qual neste caso, sempre é uma combinação de Zigzag ou um Triângulo
- Barreira
 - * Mesmas regras do Triângulo Contracionado, exceto que B e D terminam no mesmo nível
 - * Quando uma Onda 5 ocorre após um Triângulo Barreira, ou a Onda 5 é breve, ou ela se estende por uma distância excepcionalmente longa
- Expandido
 - * Regras similares e antagônicas a do Triângulo Contracionado:
 - A Onda C sempre se move além do final da Onda A
 - A Onda D sempre se move além do final da Onda B
 - A Onda E sempre se move além do final da Onda C
 - Uma linha ligando os fins das ondas B e D divergem e relação a uma linha ligando os fins das A e C
 - * As Ondas B, C, e D cada retraçam pelo menos 100%, porém não mais do que 150%, da onda anterior
- Combinações
 - Combinações são formadas por dois (ou três) padrões Corretivos separados por um (ou dois) padrões Corretivos na direção oposta
 - Os padrões Corretivos são nomeados por ordem: W, X (direção oposta), Y, (X, Z (caso triplo))
 - Um Zigzag Duplo ou Triplo é a combinação de dois ou três Zigzags
 - Uma Combinação Plana "Duplo Três" é composta (em ordem) por um Triângulo e um Plano, um Plano e um Zigzag, um Plano e um Plano, um Zigzag e um Triângulo ou um Plano e um e um Triângulo
 - Um raro "Duplo Três" é composto por três Planos
 - Zigzags Duplos e Triplos tomam o lugar de um Zigzag
 - "Três Duplo" e "Três Triplo" tomam o lugar de Planos e Triângulos
 - Triângulo Expandido ainda não foi observado como parte componente de Combinações

2.1.2. Críticas à Teoria de Elliott

Aronson [8] criticou a teoria de Elliott por ser vaga, já que não identifica consistentemente onde uma Onda começa ou termina, e que previsões estão sujeitas a revisão subjetiva.

2.2. Extensões de Neely

Glenn Neely publicou em 1987 o livro "Mastering Elliott Wave" [9], introduzindo um método científico e objetivo para análise de mercado. O **Método Neely para Análise das Ondas de Elliott**, resumido neste trabalho como **Método Elliott-Neely**, cria regras

numéricas para classificação e formação das Ondas de Elliott, eliminando quaisquer subjetividade da análise das Ondas de Elliott.

Neely criou uma visualização intermediária para as Ondas chamadas **Monowaves**. As **monowaves** são a menor unidade de variação do preço de uma ação. Enquanto, por exemplo, na Teoria de Elliott uma Onda Impulso é o conjunto de movimentos numerados de 1 a 5 (Figura 23), para Neely cada movimento é uma única **monowave**. A Figura 9 mostra como visualizar uma **monowave**.

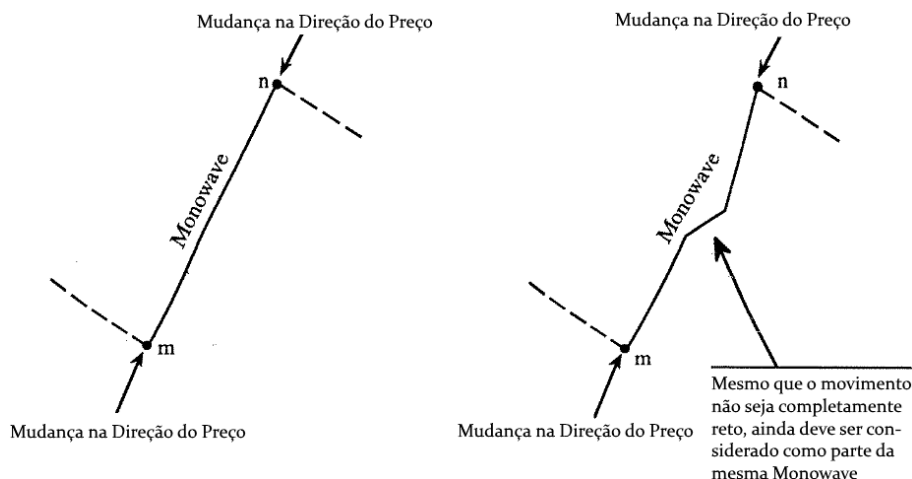


Figura 9. Visualização de uma Monowave

Utilizando a visualização intermediária das **monowaves**, Neely criou um algoritmo que combina **monowaves** em conjuntos que representam as Ondas de Elliott. Além disso, o algoritmo também recombina os conjuntos identificados entre si formando Ondas de grau superior (como listados na Tabela 2), e aplica conjuntos de regras chamadas “Extensões de Neely” que, segundo Neely, estendem a capacidade de identificação dos padrões de Elliott. Um fluxograma do algoritmo criado por Neely pode ser visto na Figura 10. Como o algoritmo ainda está em estudo, no momento só foram implementadas as funcionalidades relacionadas aos ‘Conceitos Gerais’ e o início da ‘Análise Preliminar’.

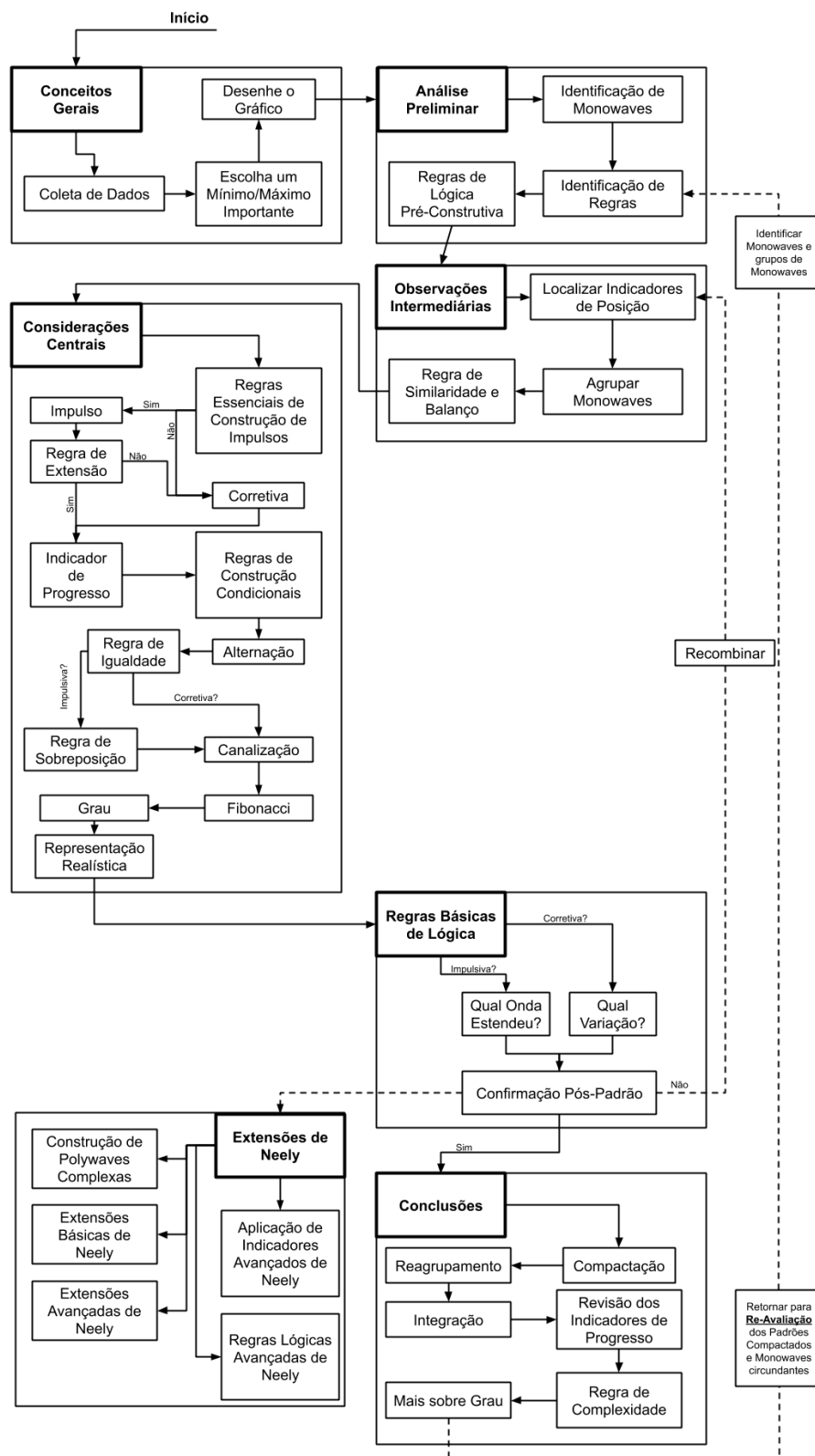


Figura 10. Fluxograma do Método Elliott-Neely

2.2.1. Limitação de Janela de Análise

Na etapa da Análise Preliminar do Método Elliott-Neely, é necessário aplicar a Regra de Neutralidade, que pode alterar o início e/ou fim das monowaves dependendo da relação entre os pontos do gráfico e o principal Movimento Direcional.

Movimentos Direcionais são conjuntos de monowaves que movimentam o preço obrigatoriamente para cima ou para baixo, e sua visualização pode ser vista na Figura 11 a seguir. Um Movimento Direcional termina quando a monowave seguinte sofre uma retração completa, e um Movimento Direcional só se inicia quando a segunda monowave não retrai mais do que 61,7% da primeira.

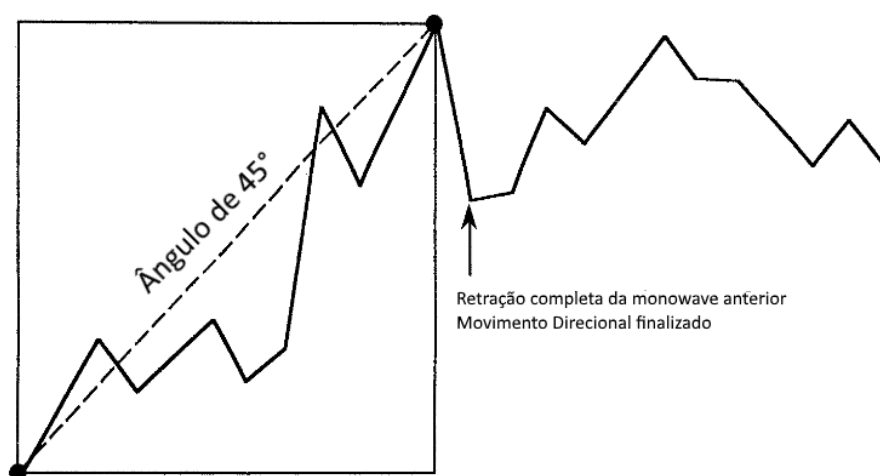


Figura 11. Movimento Direcional

A Regra da Neutralidade consiste em identificar movimentos horizontais do preço, e alterar os inícios e/ou fim das monowaves próximas ao movimento horizontal. Estes movimentos horizontais podem ser vistos na Figura 12 a seguir.

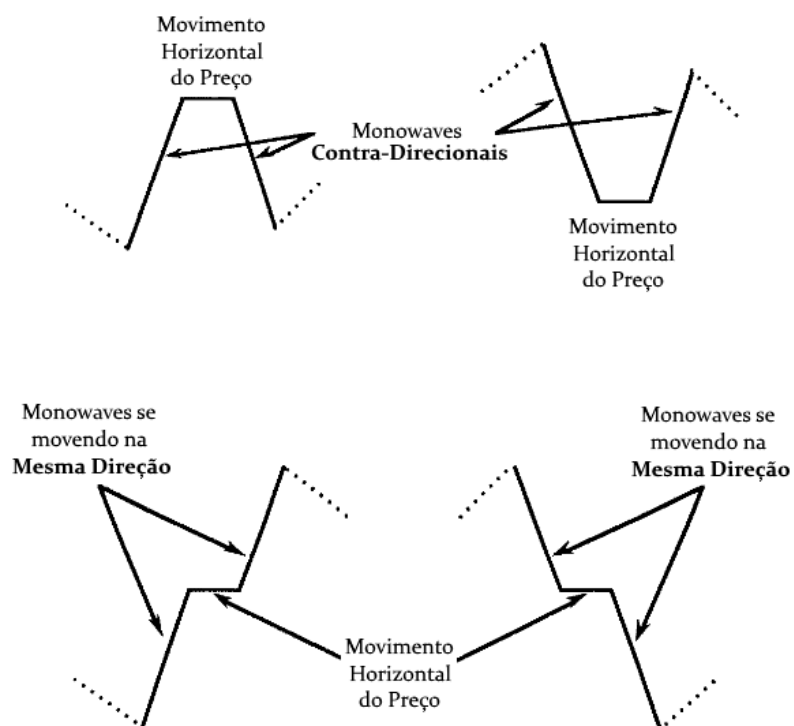


Figura 12. Movimento Horizontal do Preço

A Regra de Neutralidade não se aplica se o movimento horizontal tiver ângulo maior que 45° , o que implica que as monowaves mantêm seus inícios e fins originais. Quando o movimento horizontal do preço não possuir um ângulo maior do que 45° , a Regra de Neutralidade se aplica. Para movimentos contra-direcionais, o fim da monowave anterior ao movimento horizontal pode ser alterado para o fim do movimento horizontal. Para movimentos com a mesma direção, a monowave onde está o movimento horizontal pode ser separada em 3 monowaves distintas (a parte da monowave antes do movimento horizontal, a monowave com o movimento horizontal, e o restante da monowave). Essas regras podem ser visualizadas nas Figuras 13 e 14 a seguir.

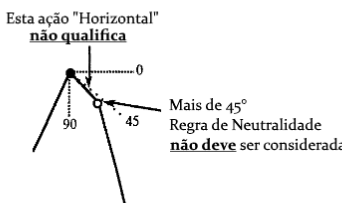
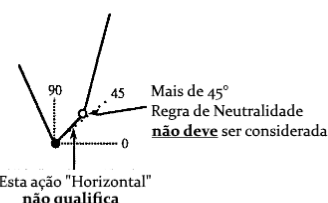
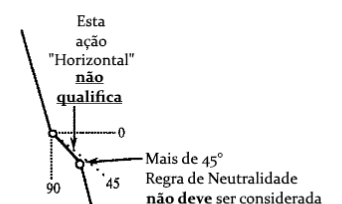
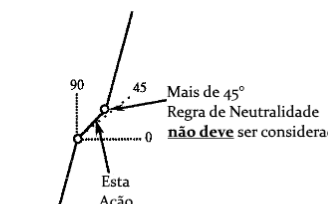
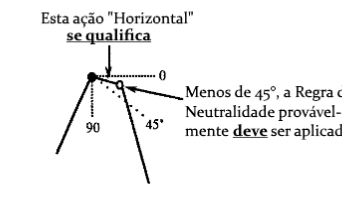
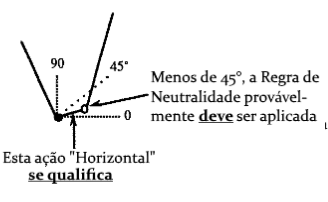
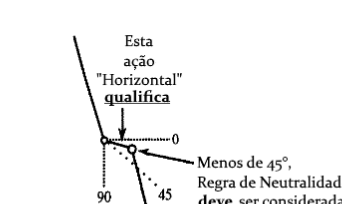
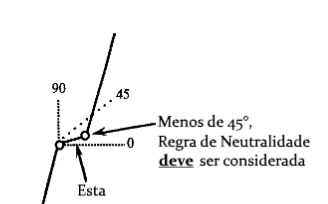
	Não se Aplica a Regra de Neutralidade	
Monwaves Contra-Direcionais	 <p>Esta ação "Horizontal" não qualifica</p> <p>Mais de 45° Regra de Neutralidade não deve ser considerada</p>	 <p>Mais de 45° Regra de Neutralidade não deve ser considerada</p> <p>Esta ação "Horizontal" não qualifica</p>
Monwaves com Mesma Direção	 <p>Esta ação "Horizontal" não qualifica</p> <p>Mais de 45° Regra de Neutralidade não deve ser considerada</p>	 <p>Mais de 45° Regra de Neutralidade não deve ser considerada</p> <p>Esta Ação "Horizontal" não qualifica</p>
	Se Aplica a Regra de Neutralidade	
Monwaves Contra-Direcionais	 <p>Esta ação "Horizontal" se qualifica</p> <p>Menos de 45°, a Regra de Neutralidade provávelmente deve ser aplicada</p>	 <p>Menos de 45°, a Regra de Neutralidade provávelmente deve ser aplicada</p> <p>Esta ação "Horizontal" se qualifica</p>
Monwaves com Mesma Direção	 <p>Esta ação "Horizontal" qualifica</p> <p>Menos de 45°, Regra de Neutralidade deve ser considerada</p>	 <p>Menos de 45°, Regra de Neutralidade deve ser considerada</p> <p>Esta ação "Horizontal" qualifica</p>

Figura 13. Regra de Neutralidade em forma gráfica

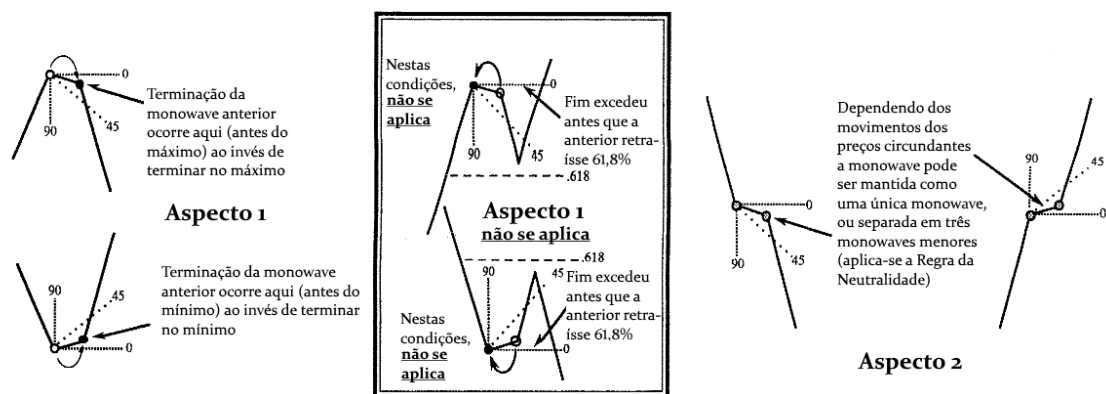


Figura 14. Execução da Regra de Neutralidade em forma gráfica

Na aplicação da Regra de Neutralidade, o primeiro Movimento Direcional dita quais monowaves devem ter seu início ou fim alterado de acordo com as regras descritas anteriormente, já que o primeiro Movimento Direcional deve alterar a escala de tempo de forma que o primeiro Movimento Direcional possua um ângulo de 45° . Porém, Neely diz que o primeiro Movimento Direcional dita a Regra de Neutralidade para todos os pontos de análise do gráfico, e que o gráfico ideal a ser analisado tem um total de cerca de 30 pontos de análise em um papel de 8 polegadas.

Visto que a implementação deste algoritmo em computador não possui a limitação física (tamanho do papel), a Regra de Neutralidade será modificada para atualizar o Movimento Direcional usado para julgar as monowaves a cada novo Movimento Direcional que ocorrer. Dessa forma, todas as monowaves serão julgadas pelo Movimento Direcional que ocorreu imediatamente anterior a elas.

2.2.2. Fatores de Estudo

As Extensões de Neely não especificam algumas configurações que podem alterar a análise resultante. O preço a ser utilizado é definido por Neely como sendo o de Negociação (*Cash Market*). Neely cita que Mercados de Negociação costumam gerar uma única cotação por dia, e essa cotação deve ser usada para compor as análises. Porém, Neely também cita que alguns Mercados de Negociação operam continuamente, e que nesses casos, uma opção é escolher um período de tempo para a análise, e usar a média entre o máximo e mínimo do período.

Uma ressalva feita por Neely é de que mercados internacionais devem ser observados somente durante o período em que o mercado está aberto no país onde se está fazendo a negociação.

- Preço da Ação
 - Média Agregada (HLC)
 - Mediana (HL)
 - Preço de Fechamento (C)
- Atualização da Regra de Neutralidade
 - Atualiza fator a cada novo Movimento Direcional
 - Não atualizar fator

- Não usar Regra de Neutralidade
- Escala de Tempo
 - Todas as escalas devem usar mesma configuração?
 - Escalas diferentes usam configurações diferentes?

Após a finalização do algoritmo, serão feitas análises de ações com diferentes configurações, e serão observadas as diferenças nos resultados obtidos para cada configuração. O objetivo destes testes será analisar o efeito e impacto de cada configuração sobre a análise resultante.

3. Arquitetura

Nesta seção será descrita a arquitetura do programa que implementa o algoritmo estudado.

3.1. Contexto do Sistema

Um programa capaz de implementar a classificação das Ondas de Elliott de forma consistente e replicável pode se tornar uma forte ferramenta na análise de variações de preços de ativos no mercado financeiro. A característica programática das Extensões de Neely tornam o método de Elliott-Neely um ótimo candidato para elaboração de um sistema de análise das variações de preços de ativos no mercado financeiro.

3.2. Requisitos

Nesta seção são delineados os requisitos para o sistema a ser implementado. Inicialmente, a visão arquitetural e as partes interessadas são apresentadas, seguidas dos requisitos de alto nível e casos de uso. Finalmente, os requisitos funcionais, não-funcionais e evolutivos são apresentados.

3.2.1. Visão arquitetural

Nesta parte os principais conceitos arquiteturais serão destacados e explicados.

A entrada utilizada no sistema são os preços de **ativos**, explicados na Seção 1.2. O sistema em desenvolvimento neste projeto de pesquisa atualmente utiliza os dados exportados de outros softwares em um arquivo .CSV (*comma separated values*). Há planos de configurar o programa para obter os dados atualizados em tempo real, assim permitindo uma análise dinâmica. A implementação atual, estática através de um arquivo .CSV, foi escolhida por ser mais simples de se implementar, além de reduzir a dependência de programas de terceiros.

A **interface gráfica** é a parte da aplicação através da qual o usuário final interage com o sistema e acessa suas funcionalidades.

O **algoritmo Elliott-Neely** é a parte crítica do sistema, sendo o principal responsável pela análise dos preços de ativos alimentados no sistema. O algoritmo é discutido na Seção 2.2 e sua implementação programática é o objeto de estudo deste Mestrado.

O **relatório final** é o resultado da análise fornecida pelo sistema. Ele será implementado na forma gráfica, em notações de Elliott explicadas na Seção 2.1. Caso a conclusão do algoritmo seja antecipada ao cronograma, será implementada a emissão de um relatório escrito.

3.2.2. Stakeholders

O **usuário final** é o pesquisador/analizador de mercado interessado em realizar uma análise das Ondas de Elliott nos preços de um ativo do mercado financeiro. Seus interesses são obter um resultado confiável dos dados analisados e conseguir utilizar o sistema sem grandes dificuldades.

O **desenvolvedor** é um pesquisador/programador interessado em atualizar e/ou modificar o sistema, podendo por exemplo, adicionar novos métodos de análise. Seus interesses são a manutenibilidade do sistema e a testabilidade do sistema e de seus elementos.

Para determinação dos atributos chaves de qualidade do sistema em desenvolvimento neste projeto de pesquisa, foi utilizado um sistema de pontuação onde cada *stakeholder* é igualmente importante e recebe 100 pontos para distribuir entre os atributos de qualidade que ele considera importantes. A Tabela 4 mostra a distribuição dos interesses de cada *stakeholder*, assim como o interesse total de cada atributo do sistema.

Tabela 4. Representação Quantitativa dos Interesses dos Stakeholders

Stakeholders	Confiabilidade	Manutenibilidade	Usabilidade	Testabilidade
Usuário Final	80		20	
Desenvolvedor	30	40		30
Total	110	40	20	30

3.2.3. Atributos de Qualidade

Os atributos chaves de qualidade são aqueles que concentram o maior interesse dos *stakeholders* (Tabela 4).

Confiabilidade é o atributo chave deste sistema. Por ser um sistema de análise de dados, a confiabilidade dos resultados está diretamente ligada a qualidade do sistema em si.

Manutenibilidade é um atributo secundário deste sistema. A medida que novos pesquisadores se interessem no sistema, a facilidade de extensão, otimização e manutenção se toram chaves em manter o interesse dos novos desenvolvedores.

Testabilidade permitirá que desenvolvedores tenham mais confiança a cerca do sistema e das modificações feitas nele.

Usabilidade facilitará a interação dos usuários com o sistema.

3.2.4. Requisitos de Alto-Nível

Os seguintes requisitos representam as funcionalidades de alto-nível do sistema. Cada requerimento de alto-nível será refinado em detalhe por vários requisitos específicos e

técnicos.

Análise de Dados

HL-1	Obrigatório	O sistema deve permitir a importação dos dados de análise.
------	-------------	--

Análise de Dados

HL-2	Obrigatório	O sistema deve fazer uma análise das Ondas de Elliott de acordo com as Extensões de Neely de forma precisa e confiável.
------	-------------	---

Configuração da Análise

HL-3	Obrigatório	O sistema deve exibir os resultados da análise de forma legível para o usuário.
------	-------------	---

Exibição de Resultados

HL-4	Obrigatório	O sistema deve permitir a mudança da configuração da análise a ser feita sobre os dados.
------	-------------	--

3.2.5. Casos de Uso

O diagrama de casos de uso é ilustrado na Figura 15, demonstrando os mais importantes casos de uso e seus atores.

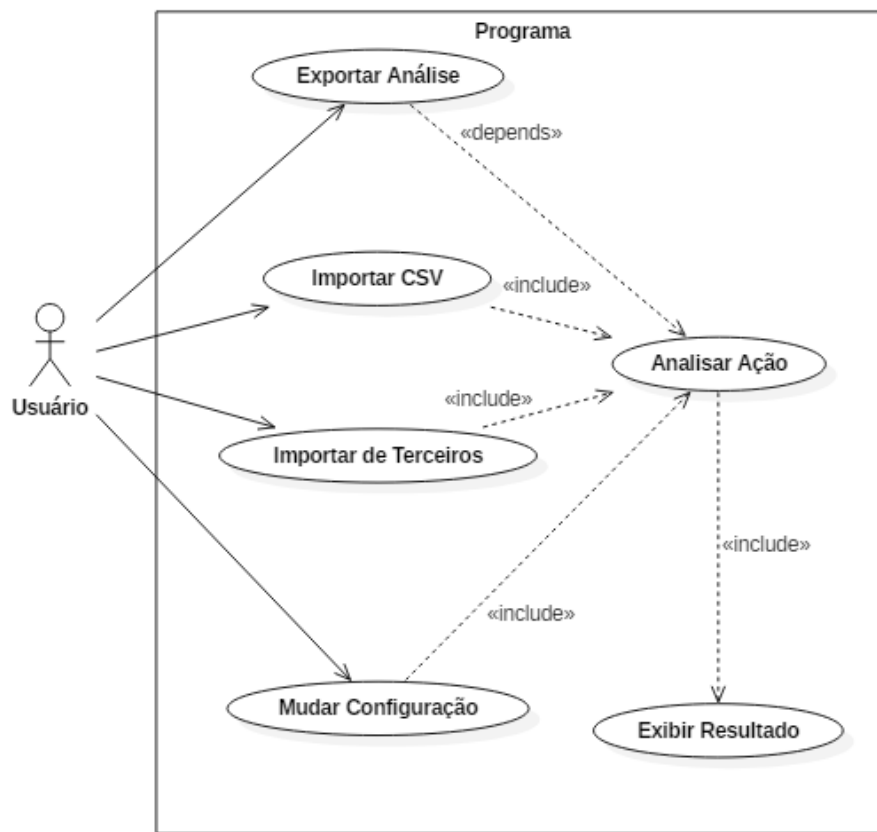


Figura 15. Diagrama de Casos de Uso

Além disso, as Tabelas a seguir mostram os casos de uso arquiteturalmente significantes. A numeração refere-se ao requerimento de Alto-nível, por exemplo, UC-1.x refere-se a HL-1.

Tabela 5. UC-1.1 - Importar .CSV

UC-1.1 - Importar .CSV

Ator Primário	Usuário Final
Objetivo	Importar um Arquivo .CSV com preços de ativo a serem analisados
Condições Iniciais	O sistema está operacional, e o usuário final possui o arquivo com o histórico da Ação a ser analisada
Cenário Principal de Sucesso	
	<ol style="list-style-type: none"> 1. O usuário clica na opção de Inserir o Arquivo .CSV com os dados da Ação 2. O usuário localiza o arquivo .CSV a ser analisado 3. O sistema valida o arquivo .CSV 4. O sistema realiza a análise do histórico de preços da Ação (UC-2.1)
Extensões	
	<ol style="list-style-type: none"> 3.a O sistema identifica que o arquivo .CSV não tem as colunas necessárias <ol style="list-style-type: none"> 3.a.1 O sistema não analisa o arquivo .CSV 3.a.2 O sistema emite um aviso de que o arquivo .CSV não é válido 3.a.3 O usuário obtém um arquivo .CSV válido 3.a.4 Ir para o passo 1
Sub-variações	
Condições Finais	O histórico de preços da ação é analisado e o usuário final pode alterar as configurações de análise.
Requisitos Relacionados	HL-1, FR-1, FR-2

Tabela 6. UC-1.2 - Importar de Terceiros

UC-1.2 - Importar de Terceiros

Ator Primário	Usuário Final
Objetivo	Importar os preços de ativo a serem analisados via terceiros
Condições Iniciais	O sistema está operacional, e o usuário final possui uma fonte de dados alternativa
Cenário Principal de Sucesso	<ol style="list-style-type: none"> 1. O usuário configura o programa para estabelecer a conexão com o terceiro 2. O sistema valida as configurações da conexão com o terceiro 3. O sistema requisita os dados do terceiro 4. O sistema realiza a análise do histórico de preços da Ação (UC-2.1)
Extensões	<ol style="list-style-type: none"> 2.a O sistema não consegue realizar a conexão com o terceiro <ol style="list-style-type: none"> 2.a.1 O sistema não realiza a análise 2.a.2 O sistema emite um aviso de que a conexão não foi estabelecida 2.a.3 Ir para o passo 1
Sub-variações	
Condições Finais	O histórico de preços da ação é analisado e o usuário final pode alterar as configurações de análise.
Requisitos Relacionados	HL-1, FR-6

Tabela 7. UC-2.1 - Analisar Ação

UC-2.1 - Analisar Ação

Ator Primário	Sistema
Objetivo	Analisar o Histórico de Preços de uma Ação
Condições Iniciais	O sistema está operacional, e o arquivo foi inserido pelo usuário e validado pelo sistema
Cenário Principal de Sucesso	<ol style="list-style-type: none">1. O sistema realiza a análise do histórico de preços da Ação.2. O sistema exibe os resultados para o usuário (UC-3.1)
Extensões	
Sub-variações	
Condições Finais	A interface exibe o resultado da análise e o usuário pode alterar as configurações de análise.
Requisitos Relacionados	HL-2, FR-3

Tabela 8. UC-3.1 - Exibir Resultado

UC-3.1 - Exibir Resultado

Ator Primário	Sistema
Objetivo	Exibir o resultado da análise feita
Condições Iniciais	O sistema finalizou uma análise (UC-2.1)
Cenário Principal de Sucesso	<ol style="list-style-type: none">1. O sistema deve exibir o gráfico do histórico do preço da Ação2. O sistema deve sobrepor a este gráfico as notações das Ondas de Elliott
Extensões	
Sub-variações	
Condições Finais	O resultado da análise é visível para o usuário final.
Requisitos Relacionados	HL-3, FR-4

Tabela 9. UC-4.1 - Mudar Configuração

UC-4.1 - Mudar Configuração

Ator Primário	Usuário Final
Objetivo	Alterar as configurações de análise
Condições Iniciais	Um conjunto de dados já foi inicializado no sistema
Cenário Principal de Sucesso	
<ol style="list-style-type: none"> 1. O usuário altera uma das configurações disponíveis na interface 2. O sistema atualiza a análise de acordo com as novas configurações (UC-2.1) 	
Extensões	
Sub-variações	
Condições Finais	A análise é atualizada para a nova configuração.
Requisitos Relacionados	HL-4, FR-3, FR-4, FR-5

O Caso de Uso UC-1.2 descrito na Tabela 6 que importa dados de terceiros é uma funcionalidade planejada. O objetivo é que os dados sejam obtidos, por exemplo, diretamente através de uma interface TCP/IP, ou através de uma conexão com um programa como o MetaTrader 5. Se esta funcionalidade for implementada, a atualização dos dados e da análise poderá ser feita em tempo real, a medida que novos dados sejam disponibilizados.

3.2.6. Requisitos Funcionais

Nesta seção, os requisitos que capturam o comportamento esperado do sistema serão apresentados (Tabela 10). Este comportamento pode ser expresso como serviços, tarefas ou funções que o sistema deve executar. Estes são derivados a partir dos requisitos de Alto-nível e dos casos de uso.

Tabela 10. Requisitos Funcionais

FR-1	Obrigatório	O sistema deve validar se a entrada é válida.
FR-2	Obrigatório	O sistema deve informar o usuário caso a entrada seja inválida.
FR-3	Obrigatório	O sistema deve realizar a análise automaticamente.
FR-4	Obrigatório	O sistema deve exibir o resultado ao usuário automaticamente.
FR-5	Obrigatório	O sistema deve atualizar automaticamente a análise quando o usuário alterar as configurações.
FR-6	Desejável	E desejável que o sistema consiga obter os dados através de uma conexão com um terceiro.

3.3. Análise

Esta seção foca em esclarecer o processo de desenvolvimento do sistema. Aqui, todas as suposições que são relevantes ao processo são mostradas, assim como o roteiro tecnológico de desenvolvimento e as decisões de alto-nível tomadas durante o projeto.

3.3.1. Suposições

- O usuário final tem conhecimento sobre o mercado de ações (Seção 1.1);
- O usuário final tem conhecimento sobre a representação dos preços de ações (Seção 1.2);
- O usuário final tem conhecimento sobre as Ondas de Elliott (Seção 2.1);
- O usuário final tem a sua disposição um software capaz de exportar os preços da ação que quer analisar em formato .CSV;

3.3.2. Roteiro tecnológico

A primeira versão do sistema foi desenvolvida somente para gerar a representação gráfica do histórico de preços e implementar mudanças de configurações básicas. Esta versão teve como objetivo demonstrar que o ambiente de desenvolvimento era favorável e compatível com a aplicação a ser criada.

A segunda versão do sistema está sendo desenvolvida para implementar a análise de preços de ações no mercado financeiro conforme o método de Elliott-Neely. Esta versão será capaz de realizar uma análise das Ondas de Elliott de forma automática, com mínima ou nenhuma intervenção do usuário.

Versões futuras do sistema serão capazes de realizarem previsões futuras dos preços das ações a partir, primariamente, da verificação dos padrões das Ondas de Elliott.

3.4. Arquitetura do Software

A arquitetura do software é descrita nesta seção. Esta seção é composta dos principais motivadores arquiteturais, seguidos das visões arquiteturais. A representação foi criada baseada no modelo 4+1 [10], *i.e.*, consistindo na visão lógica, visão de processo, visão de desenvolvimento, visão física e visão de caso de uso. Para fortalecer o conceito original do modelo 4+1, cada visão foi projetada usando UML [11], conforme recomendado em [12].

3.4.1. Visão Lógica

Esta visão mostra as abstrações chaves e mecanismos que são usados dentro do programa para realizar suas funcionalidades. A interação entre estes elementos também é descrita. Os elementos estão alocados em camadas de acordo com suas responsabilidades. A representação gráfica desta visão pode ser observada na Figura 16. Foi utilizado o padrão MVC para esta Visão [13] [14].

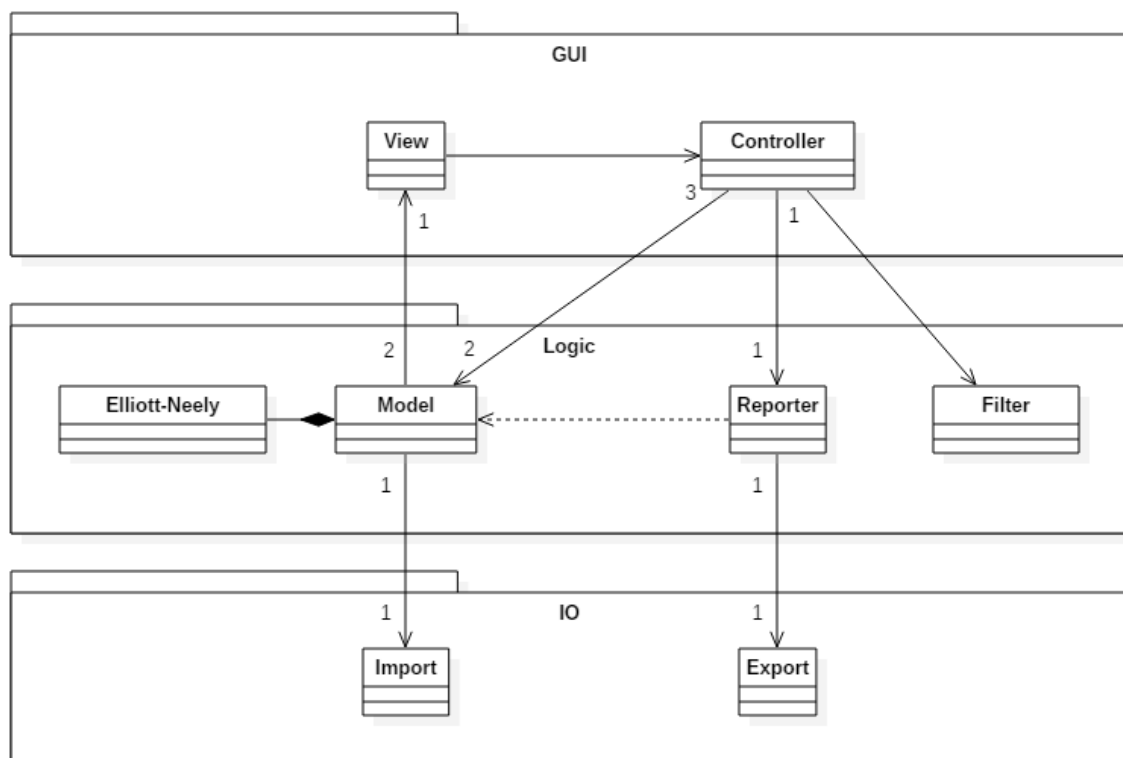


Figura 16. Visão Lógica

- **GUI:** Interface com Usuário
 - A **Visualização (View)** é a parte da camada de interface com a qual o usuário interage diretamente, acessando as funcionalidades do programa.
 - O **Controlador (Controller)** são os dispositivos lógicos e de controle que realizam as funcionalidades do programa. O programa utiliza 3 controladores: Global (para funcionalidades gerais), Data (para processamento de dados) e UI (para interfaceamento gráfico).

- **Logic:** Lógica
 - O **Modelo** (*Model*) é a parte da camada lógica onde se encontram os dados sendo processados pelo programa ou visualizados na interface. Estão sendo utilizados dois Modelos: um de dados (sob o qual o Método de **Elliott-Neely** é aplicado), e um de visualização, que é enviado para a **GUI**.
 - O **Relator** (*Reporter*) é a parte responsável por criar uma relatório para o usuário.
 - O **Filtro** (*Filter*) é responsável por filtrar quais informações serão disponibilizadas para o usuário de acordo com a configuração escolhida.
- **IO:** Entradas e Saídas
 - A **Importação** (*Import*) é responsável por receber os dados do usuário.
 - A **Exportação** (*Export*) é responsável por enviar dados para o usuário.

3.4.2. Visão de Processo

Esta visão mostra a representação abstrata do processo principal no sistema. Para apresentar estas representações, os casos de uso descritos na Seção 3.2.5 foram utilizados para retratar as partes envolvidas na realização de cada caso de uso.

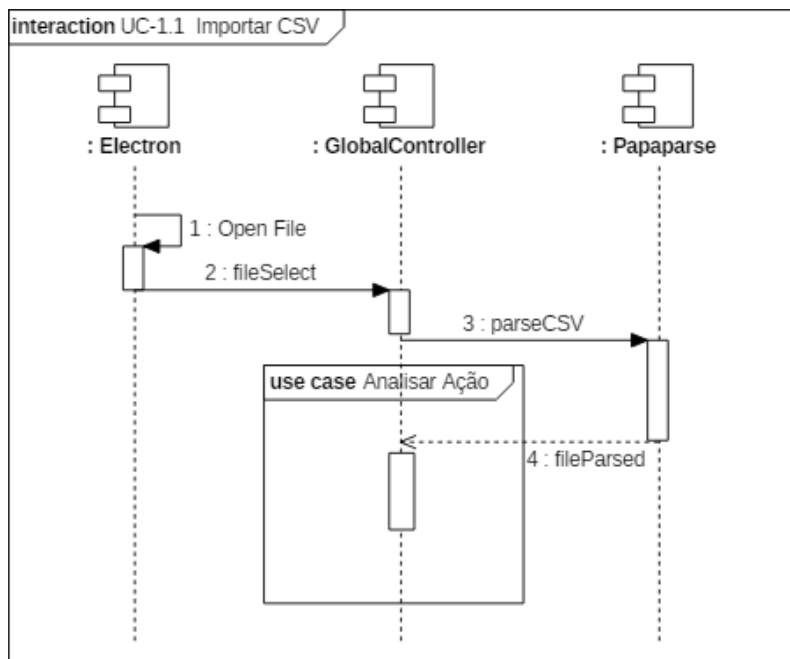


Figura 17. Visão de Processo para UC-1.1: Importar CSV

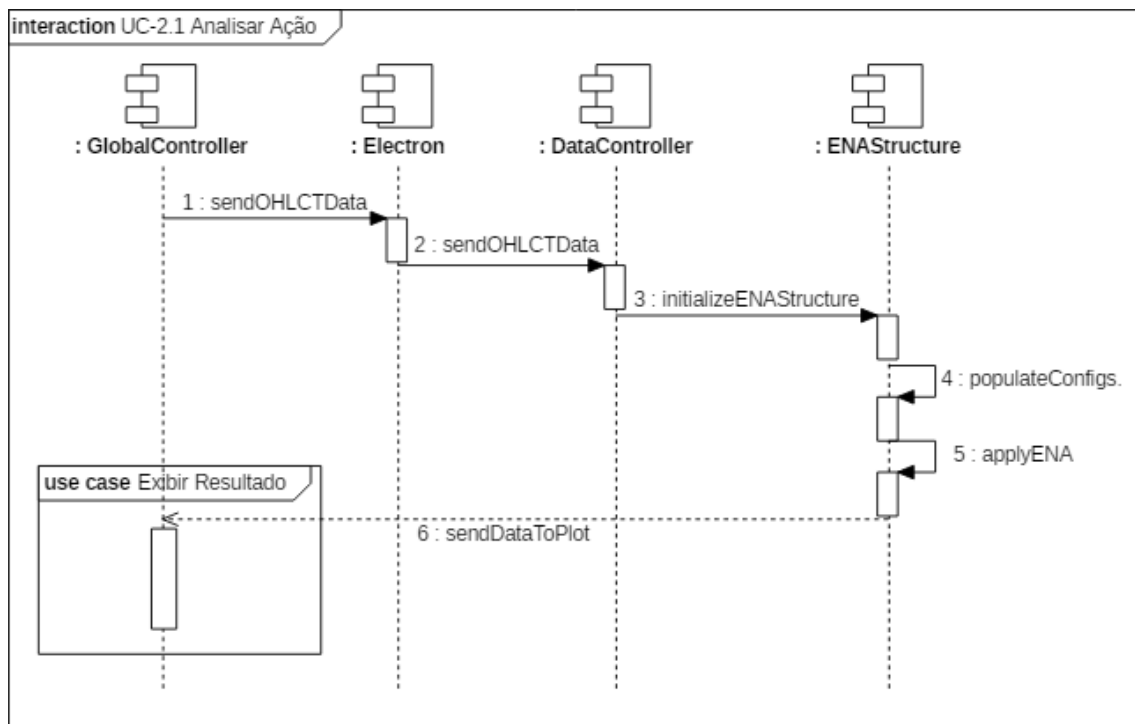


Figura 18. Visão de Processo para UC-2.1: Analisar Ação

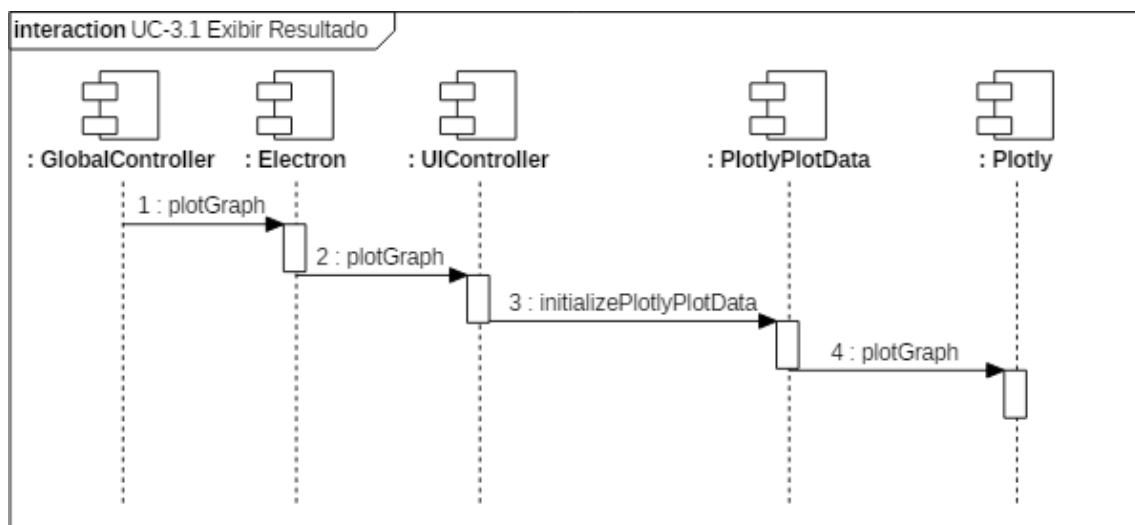


Figura 19. Visão de Processo para UC-3.1: Exibir Resultado

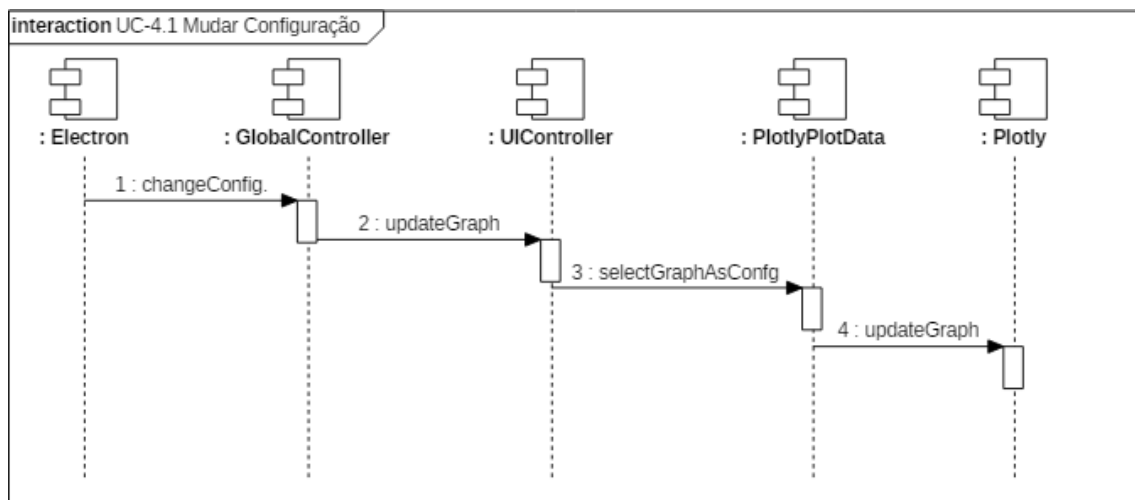


Figura 20. Visão de Processo para UC-4.1: Mudar Configuração

3.4.3. Visão de Implementação

Essa visão foca no gerenciamento de configuração e organização real do módulo de software no ambiente de desenvolvimento, englobando os principais componentes usados para montar e gerar o sistema físico. Nas seções a seguir, esses componentes são decompostos e descritos em termos de responsabilidades e interfaces. A representação gráfica desta visão pode ser observada na Figura 21.

- * Receber os dados de entrada do usuário a partir da interface **FileHandler**.
 - * Converter o arquivo .CSV do usuário usando a biblioteca **Papaparse** via interface **CSVHandler**.
 - * Gerenciar os outros Controladores por meio do **Electron**.
- **DataController**
 - Gerenciador dos dados da aplicação.
 - Responsabilidades:
 - * Receber os dados entrados pelo usuário via interface **HTML**.
 - * Salvar os dados entrados pelo usuário no modelo **ENAStructure** via interface **ENAInterface**.
- **UIController**
 - Gerenciador da visualização e interface com usuário.
 - Possui uma estrutura chamada **PlotKey** que registra as modificações feitas nas configurações de exibição quando o programa solicitar uma atualização do gráfico.
 - Responsabilidades:
 - * Receber os dados para plotar o gráfico via interface **ENAInterface**.
 - * Plotar o gráfico usando a biblioteca **Plotly** via interface **PlotControl**.
 - * Enviar os dados para geração do relatório da análise para o componente **ENAREport** via interface **ReportInterface**.
- **ENAStructure**
 - Organiza os dados do usuário de acordo com as configurações possíveis.
 - Os dados do usuário são salvos na estrutura **OHLCTStructure**.
 - A análise gerada é feita e salva na estrutura **MonowaveVector**.
 - A estrutura **MonowaveVector** realiza a análise através da organização de um vetor de **Monowave(s)** e de uma estrutura de **DirectionalAction**.
 - Para cada combinação de configurações disponíveis, são gerados um par de **OHLCTData** e **MonowaveVector**
- **ENAREport**
 - Gera um relatório da análise para o usuário.
 - O relatório gerado é exportado para o usuário através do módulo **ENA_IO** via interface **ExportInterface**.
- **Relayout**
 - Guarda o histórico de visualizações do gráfico, permitindo desfazer e refazer comandos de Zoom.
 - Permite aplicar um comando de Fit vertical para melhor utilizar o espaço horizontal do gráfico.
- **PlotlyPlotData**
 - Guarda os dados da **ENAStructure** formatados para visualização usando a biblioteca **Plotly**.

3.4.4. Visão de Implantação

Esta visão foca na distribuição do software no hardware. Ela apresenta o hardware e as peças de software que o compõem. A seguir, cada parte do software e hardware serão explicadas. A representação gráfica desta visão pode ser observada na Figura 22.

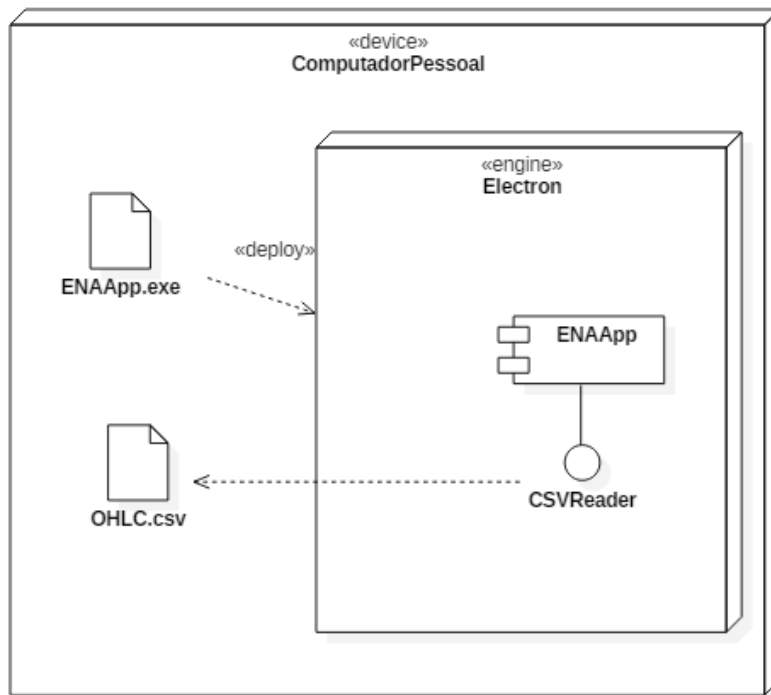


Figura 22. Visão de Implantação

O **computador pessoal** representa qualquer computador pessoal rodando um sistema operacional Windows, Linux ou MacOS.

O **ENAAp.exe** é o arquivo executável gerado a partir do projeto. Ele está listado como “.exe” a título de exemplo e simplificação, pois o Electron é capaz de gerar programas para vários sistemas operacionais utilizando o mesmo código fonte.

O executável cria uma instância do **Electron**, na qual a aplicação “Elliott-Neely Analysis App” (**ENAAp**) executa. A aplicação é capaz de importar arquivos .csv através da interface **CSVReader**. Na visualização, o arquivo OHLC.csv é usado como exemplo.

3.4.5. Decisões de Desenvolvimento de Software

Esta seção lista as decisões relacionadas ao desenvolvimento do software e implementação. Elas podem ser vistas nas Tabelas seguintes. O modelo utilizado para estas decisões foi baseado em [17].

Tabela 11. Decisão - Linguagem de Implementação

Nome	Linguagem de Implementação
Decisão	1
Estado	Aprovada
Problema	Qual linguagem de programação utilizar para implementar o algoritmo
Decisão	Electron (<i>framework</i>) + JavaScript (lógica) + HTML5 & CSS (interface)
Alternativas	C, Java, C#, Ruby, Python
Argumentos	<p>JavaScript é uma linguagem interpretada, com interpretadores disponíveis em vários sistemas operacionais, muitos destes possuindo código aberto com licenças permissivas (como o Chrome V8 [18]).</p> <p>Node.js [19] é um <i>runtime</i> para JavaScript que implementa uma modificação da <i>engine</i> Chrome V8, permitindo a execução de código em JavaScript sem a necessidade de um navegador web.</p> <p>NPM [20] é um catálogo de bibliotecas e projetos em JavaScript, reunindo mais de 60 mil módulos e controle de versão para cada módulo.</p> <p>Electron [21] é um <i>framework</i> que permite a criação de programas em JavaScript + HTML5 & CSS capazes de rodar em diversos sistemas operacionais sem problemas de compatibilidade.</p> <p>JavaScript possui tipificação dinâmica de variáveis, o que permite uma maior flexibilidade na criação e manuseio de dados.</p> <p>Este ecossistema permite a criação de aplicações altamente flexíveis e compatíveis, com acesso a um grande acervo de bibliotecas e módulos para serem integrados, sem problemas de licenciamento e com controle de versão individual por módulo.</p>

Tabela 12. Decisão - Análise de Diferentes Configurações

Nome	Análise de Diferentes Configurações
Decisão	2
Estado	Aprovada
Problema	Como fazer a análise para as diferentes configurações, conforme listadas e descritas na Seção 2.2.2
Decisão	Fazer a análise para todas as configurações possíveis e somente alterar qual está visível
Alternativas	<ol style="list-style-type: none"> 1. Realizar a análise somente para a configuração atual e refazer a análise para cada mudança na configuração 2. Registrar cada análise feita e somente refazer a análise caso uma nova configuração seja escolhida 3. Fazer a análise para todas as configurações possíveis e somente alterar qual está visível
Argumentos	<p>A opção 1 implica em um menor gasto inicial de recursos, já que seria feita somente uma análise, mas causaria desperdício de recursos quando fosse solicitada a análise de uma configuração já visualizada anteriormente.</p> <p>A opção 2 levaria implicaria na necessidade de se implementar um controle adicional para verificar se a análise já foi feita antes.</p> <p>A opção 3 implica em um maior gasto inicial de recursos e facilita a mudança entre configurações.</p> <p>Visto que a implementação deste sistema visa a exploração das diferentes configurações, foi decidido pela terceira opção.</p>

4. Cronograma e Progressão

Nesta seção são listadas as atividades previstas para realização do trabalho proposto. Na Tabela 13 é apresentado o cronograma para estas atividades. Na Tabela 14 são listadas as atividades individuais envolvidas na elaboração e programação do algoritmo estudado.

1. Obtenção dos créditos referentes as disciplinas do Mestrado;
2. Pesquisa bibliográfica;
3. Redação da monografia para o Exame Geral de Qualificação;
4. Exame Geral de Qualificação;
5. Elaboração do algoritmo proposto e programação;
6. Geração e execução de casos de testes, e avaliação do sistema;
7. Análise dos efeitos das configurações sobre os resultados do algoritmo;
8. Redação final da Dissertação; e
9. Defesa do Mestrado.

Atividades	2018					2019										2020								
	Agosto	Setembro	Outubro	Novembro	Dezembro	Janeiro	Fevereiro	Março	Abril	Maior	Junho	Julho	Agosto	Setembro	Outubro	Novembro	Dezembro	Janeiro	Fevereiro	Março	Abril	Maior	Junho	Julho
1. Créditos																								
2. Pesquisa bibliográfica																								
3. Escrita da qualificação																								
4. Exame de qualificação																								
5. Elaboração do prog.																								
6. Testes																								
7. Análise de config.																								
8. Escrita da dissertação																								
9. Defesa do mestrado																								

Tabela 13. Cronograma de atividades

Tópico	Atividade	Descrição	Progresso	Estimado
Aquisição de Dados	Importação de Dados	Importação dos Dados OHLC em arquivos .CSV exportados a partir do Programa MetaTrader 5. Utilizando a biblioteca Papaparse .	100%	-
Pré-Tratamento de Dados	Passagem dos Dados para Classe OHLCT	Transferência dos dados importados do MetaTrader 5 para a Classe OHLCT	100%	-
	Escalas de Tempo	Replica dados originais em diferentes escalas de tempo. Implementados: 1 min, 1 hora, 1 dia. Escalável para fácil configuração de novas escalas.	100%	-
	Valor Típico	Cálculo do Valor Típico sobre o qual a análise será feita. Implementados: Valor Agregado (HLC), Mediana (HL). Escalável para fácil configuração de novos valores típicos.	100%	-
Método Elliott-Neely Análise Preliminar (1ª parte)	Regra de Neutralidade (1º passo)	Aplicação da primeira passagem da Regra de Neutralidade . Implementado através das Classes Monowave e Monowave Vector .	100%	-
	Regra de Proporção (Seleção da Janela de Análise)	Atualizada para ambiente digital e alta quantidade de dados. Afeta a segunda passagem da Regra de Neutralidade e a Regra de Retração	100%	-
	Regra de Neutralidade (2º Passo)	Une monowaves e/ou altera início/fim de monowaves de acordo com a Regra de Proporção .	100%	-
	Regra de Observação	Aplicada de forma emergente através da Classe Monowave Vector .	100%	-
	Regra de Retração	Classificação de Monowaves .	0%	3 semanas
Método Elliott-Neely Observações Intermediárias (2ª parte)	Regra de Similaridade e Balanço	Indica quais Monowaves pertencem ao mesmo grupo/grau para efeito de agrupamento.	0%	2 semanas
Método Elliott-Neely Considerações Centrais (3ª parte)	Construção de Polywaves	Classifica conjuntos de Monowaves nos padrões das Ondas de Elliott .	0%	3-4 semanas
	Extensões de Neely	Aplica regras de compactação que transformam grupos de Monowaves em Monowaves de grau maior.	0%	3-4 semanas
Interface com Usuário	Visualização em Gráfico	Utilizando a biblioteca Plotly.js	100%	-
	Configurações	Eixo Vertical (Log ou Linear), Eixo Horizontal (Tempo ou Sequencial), Tipo de Marcador (OHLC, Candelstick ou Nenhum), Resolução do Tempo (1 Minuto, 10 Minutos, 1 Hora, 1 Dia), Valor Típico (HLC ou HL), Monowave (Visualizar ou Esconder)	100%	-

Tabela 14. Progresso do algoritmo

Referências

- [1] Masur. R.N. Elliott, “The Basis of the Wave Principle”, October 1940. URL <https://commons.wikimedia.org/w/index.php?curid=1599960>. Acessado em 2019-07-17.
- [2] N. A. Curott E. P. Stringham. *The Oxford Handbook of Austrian Economics*. Oxford University Press, 2005.
- [3] A. Greif. *Institutions and the Path to the Modern Economy: Lessons from Medieval Trade*. Cambridge University Press, 2006.
- [4] U. Malmendier. Law and finance “at the origin”. *Journal of Economic Literature*, 47(4):1076–1108, 2009. doi: 10.1257/jel.47.4.1076. URL <http://dx.doi.org/10.1257/jel.47.4.1076>.
- [5] J. Hasanhodzic A. W. Lo. *The Evolution of Technical Analysis: Financial Prediction from Babylonian Tablets to Bloomberg Terminals*. John Wiley & Sons, 2010.
- [6] S. Shimizu. *The Japanese Chart of Charts*. Tokyo Futures Trading Publ. Co., 1986.
- [7] J. A. Dahlquist C. D. Kirkpatrick. *Technical Analysis: The Complete Resource for Financial Market Technicians*. FT Press, 2006.
- [8] D. R. Aronson. *Evidence-Based Technical Analysis: Applying the Scientific Method and Statistical Inference to Trading Signals*. John Wiley and Sons, 2006.
- [9] E. Hall G. Neely. *Mastering Elliott Wave*. Windsor Books for the Elliott Wave Institute, 1990.
- [10] P. Kruchten. The 4+1 view model of architecture. *IEEE Softw.*, 12(6):42–50, November 1995. ISSN 0740-7459. doi: 10.1109/52.469759. URL <http://dx.doi.org/10.1109/52.469759>.
- [11] OGM. Unified Model Language. URL <https://www.uml.org/>. Acessado em 2019-07-16.
- [12] *Applying 4+1 View Architecture with UML 2*. First FCG Software Services, 2007. URL http://www.sparxsystems.com/downloads/whitepapers/FCGSS_US_WP_Applying_4+1_w_UML2.pdf.
- [13] S. T. Pope G. E. Krasner. A cookbook for using the model-view controller user interface paradigm in Smalltalk-80. *Journal of Object-Oriented Programming*, 1(3):26–49, 1988.
- [14] F. Buschmann. *Pattern-Oriented Software Architecture: A System of Patterns*. John Wiley & Sons, 1996.
- [15] Plot.ly. Plotly JS. URL <https://plot.ly/javascript/>. Acessado em 2019-07-15.
- [16] Matt Holt. Papa Parse 5. URL <https://www.papaparse.com/>. Acessado em 2019-07-15.
- [17] A. Ackerman J. Tyree. Architecture Decisions: De-mystifying Architecture. *IEEE Software*, pages 19–27, 2005.
- [18] Google. Engine Chrome V8. URL <https://v8.dev/>. Acessado em 2019-07-15.
- [19] Node.js Foundation. Node.js. URL <https://nodejs.org/>. Acessado em 2019-07-15.
- [20] npm, Inc. Node Pack Manager. URL <https://www.npmjs.com/>. Acessado em 2019-07-15.
- [21] Electron. Electron. URL <https://electronjs.org/>. Acessado em 2019-07-15.

Apêndices

A. Padrões das Ondas de Elliott

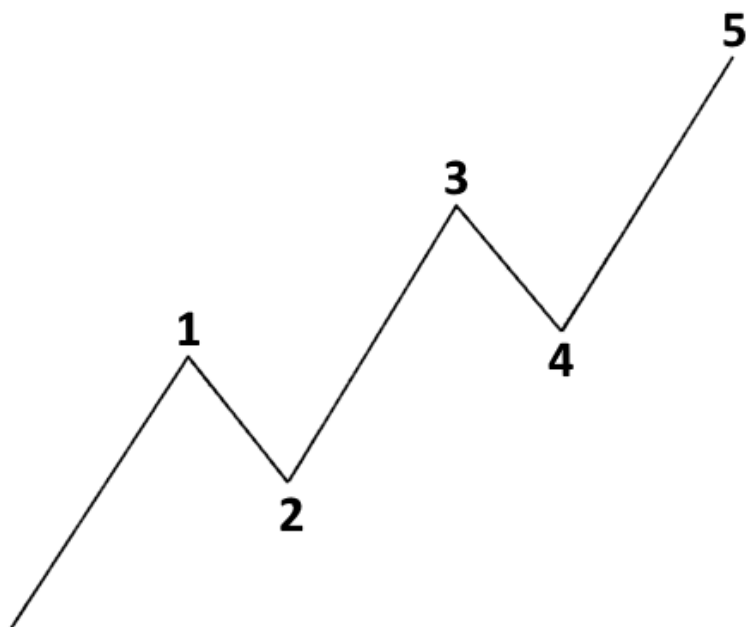


Figura 23. Padrão de Onda Motivadora: Impulso

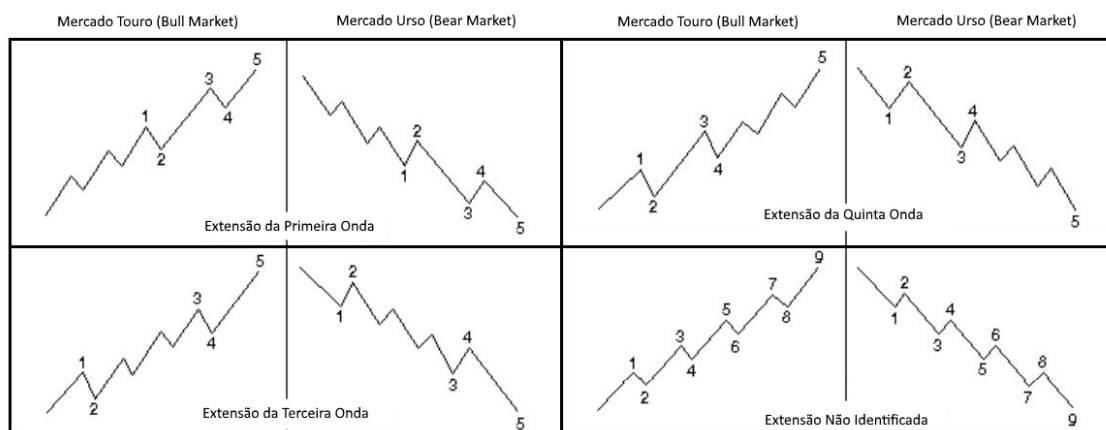


Figura 24. Padrão de Onda Motivadora: Extensão

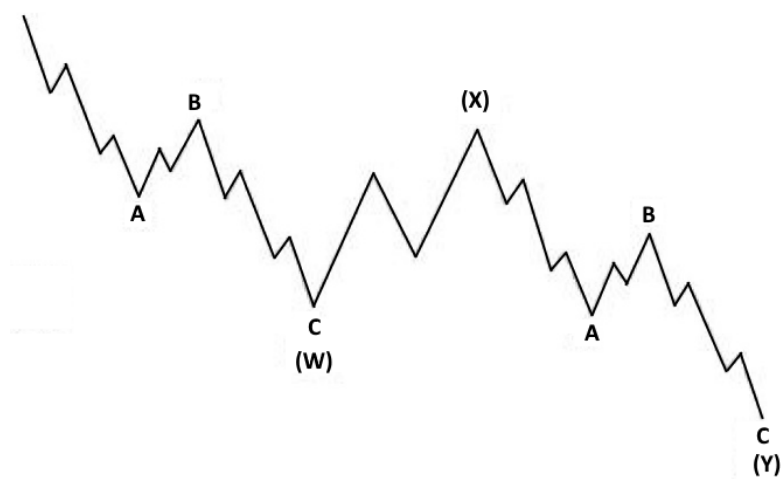


Figura 28. Padrão de Onda Corretiva: Zigzag Duplo

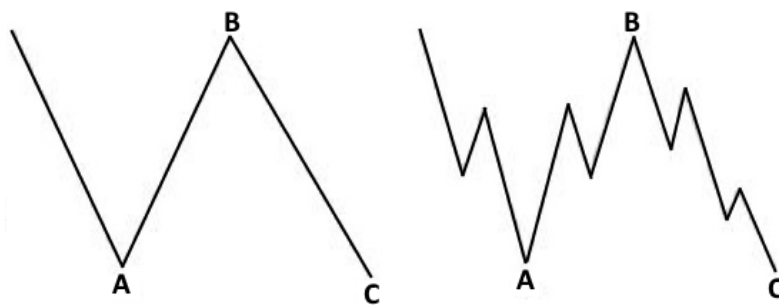


Figura 29. Padrão de Onda Corretiva: Plano

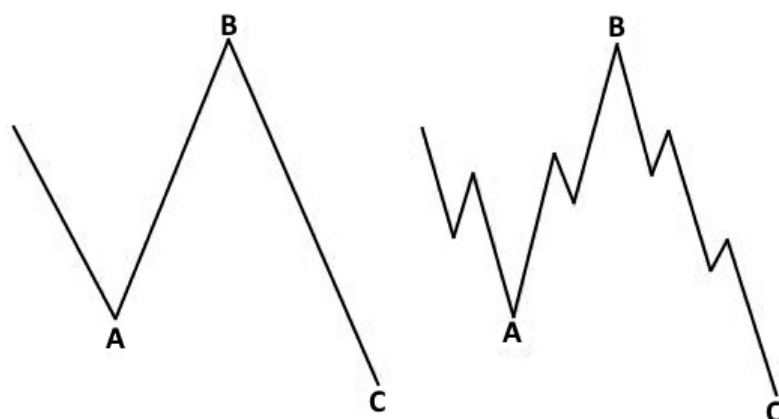


Figura 30. Padrão de Onda Corretiva: Plano Estendido

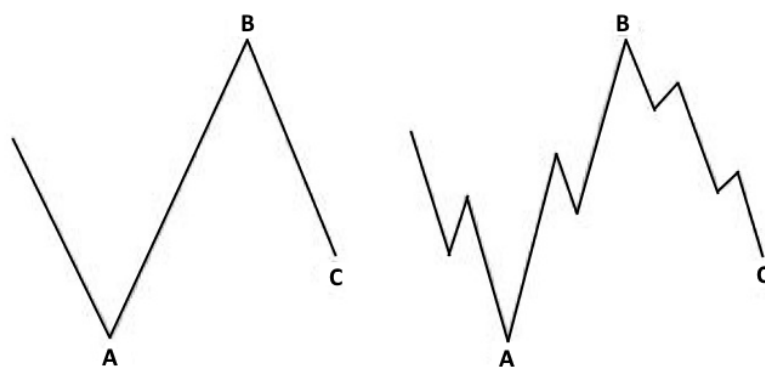


Figura 31. Padrão de Onda Corretiva: Plano Corrido

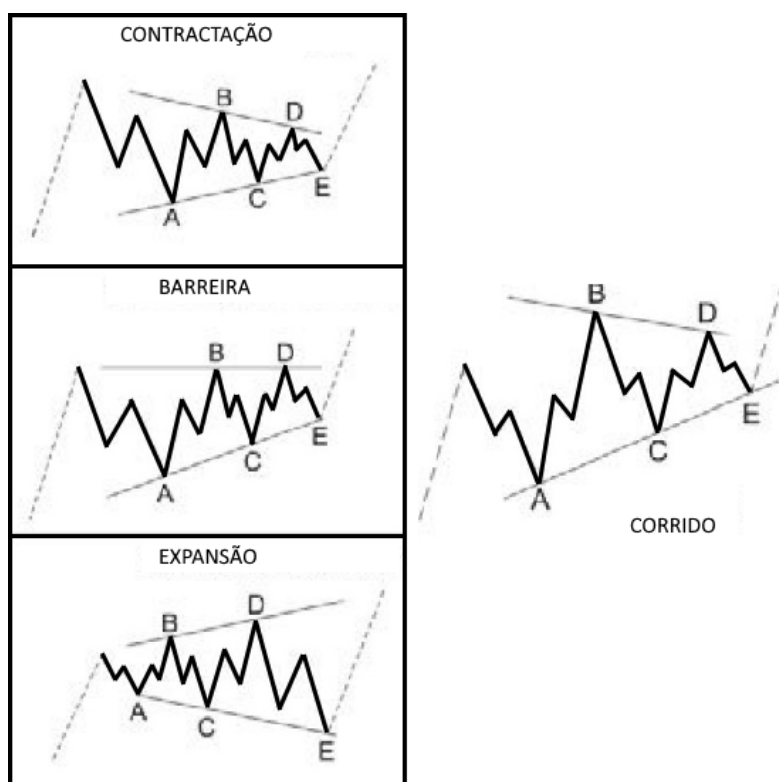


Figura 32. Padrão de Onda Corretiva: Triângulo

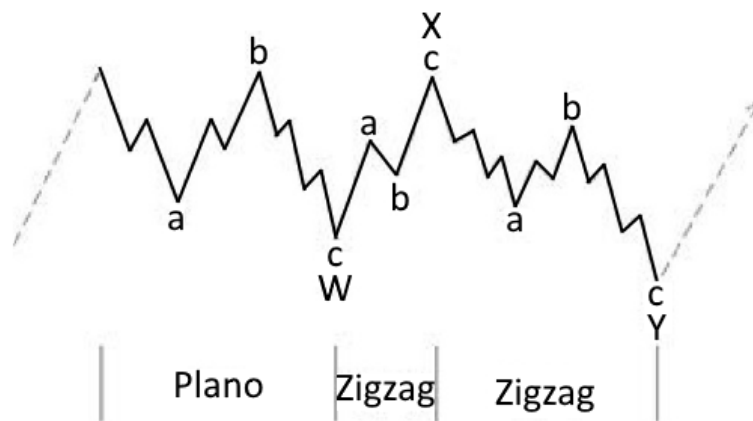


Figura 33. Padrão de Onda Corretiva: Combinação (exemplo 1)

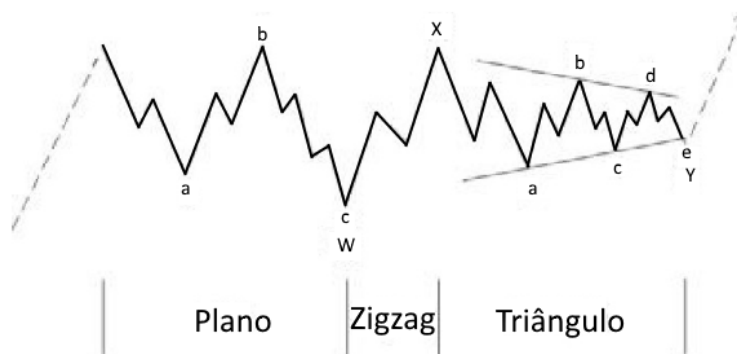


Figura 34. Padrão de Onda Corretiva: Combinação (exemplo 2)