

**Sapientia Erdélyi Magyar Tudományegyetem**  
**Műszaki és Humántudományok Kar,**  
**Marosvásárhely**  
**Szoftverfejlesztés Szak**

## **MESTERI DISSZERTÁCIÓ**

**Témavezető:**  
**Dr. Kupán A. Pál,**  
**egyetemi docens**

**Végzős hallgató:**  
**Keresztesi Tekla**

UNIVERSITATEA SAPIENTIA  
FACULTATEA DE ȘTIINȚE TEHNICE ȘI UMANISTE DIN TÂRGU-MUREȘ  
SECȚIA DEZVOLTARE DE APLICAȚII SOFTWARE

VIZAT DECAN  
Dr. Kelemen András

VIZAT ȘEF CATEDRĂ  
Dr. Kátai Zoltán

### LUCRARE DE DISERTAȚIE

Conducătorul temei:  
**Dr. KUPÁN PÁL**

Candidat: **KERESZTESI TEKLA**  
Anul absolvirii: 2015

#### 1. Conținutul proiectului

a) **Tema proiectului de diplomă:** *Sistem de tranzacționare pe piața acțiunilor bazat pe identificarea automatizată a valurilor Elliott*

b) **Problemele principale care vor fi tratate în proiect:**

- prezentarea necesității unui sistem automatizat de recunoașterea valurilor Elliott și a unui sistem de tranzacționare

- noțiuni de bază despre Teoria valurilor Elliott

- analiza identificării valurilor Elliott și descrierea strategiei de tranzacționare

- efectuarea testelor și evaluarea aplicației

c) **Desene obligatorii:**

- schema bloc al aplicației

- figuri descriptive

- figuri cu rezultate experimentale

d) **Softuri obligatorii:**

- aplicație de identificarea automatizată a valurilor Elliott și de tranzacționare automată

#### Bibliografie recomandată:

1. Rober Rougelot Prechter, Jr., A. J. Frost: Elliott Wave Principle, New Classics Library, 2005.
2. Bruce Vanstone, Tobias Hahn: Designing Stock Market Trading Systems, Harriman House, 2010.

**Termene obligatorii de consultații:** - săptămânal

**Locul practicii:** Universitatea Sapientia

Primit la data de:

Termen de predare:

Semnătura șefului de catedră

Semnătura îndrumătorului științific

Semnătura candidatului

**UNIVERSITATEA SAPIENTIA  
FACULTATEA DE ȘTIINȚE TEHNICE ȘI UMANISTE, TÂRGU-MUREȘ  
SPECIALIZAREA DEZVOLTAREA APLICAȚIILOR SOFTWARE**

**Sistem de tranzacționare pe piața acțiunilor bazat pe  
identificarea automată a valurilor Elliott**

**Lucrare de disertație**

**Coordonator științific:  
Dr. Kupán A. Pál  
conferențiar universitar**

**Absolvent:  
Keresztesi Tekla**

## Declarație

Subsemnata/Subsemnatul....., funcția....., titlul științific..... declar pe propria răspundere că absolventul specializării de ..... a întocmit prezenta lucrare cu îndrumarea mea.

Forma finală a lucrării a fost verificată de mine și acesta corespunde cu cerințele de formă și conținut precizate Consiliului Facultății de ..... în baza reglementărilor Universității Sapiientia. Lucrarea/proiectul corespunde și cerinșelor de Legea Educației Naționale 1/2011 cu modificările ulterioare, Codului de etică și deontologie profesională a Universității Sapiientia referitoare la furt intelectual.

Sunt de acord cu susținerea lucrării în fața comisiei de examen de disertație.

Localitatea,

Data:

Semnătura îndrumătorului,

## Declarație

Subsemnata/Subsemnatul....., absolvent(ă) al/a specializării....., promoția ....., cunoscând prevederile Legii Educației Naționale 1/2011 și a Codului de etică și deontologie profesională a Universității Sapienția cu privire la furt intelectual, decla pe propria răspundere că prezenta lucrare de disertație se bazează pe activitatea personală, cercetarea/proiectare este efectuată de mine, informațiile și datele preluate din literatura de specialitate sunt citate în mod corespunzător.

Localitatea,

Data:

Absolvent

Semnătura.....

## EXTRAS

Scopul disertației mele a fost investigarea recunoașterii valurilor Elliott în piața de acțiuni și crearea unei studii privind succesul unei strategii de tranzacționare bazat pe acesta. Teoria valurilor Elliott reprezintă un sistem de reguli, menit să sistematizeze mișcarea prețurilor pe piața bursieră. Ralph Nelson Elliot a observat sistemul mișcării prețurilor în anii 1930 și teoria valurilor este încă folosit cu succes și în zilele noastre. Acesta este aplicat cu cel mai mare succes în cazul indicilor, de aceea și eu tot pe indici fac recunoașterea, iar crearea strategiei de tranzacționare este alcătuit în consecință.

Există sisteme similare de tranzacționare dar în general folosesc rețele neuronale, care fac ca strategia de tranzacționare să fie neclară. Acesta prezintă cel mai mare risc posibil pentru cel care folosește un sistem de tranzacționare existent.

În prima parte a lucrării prezint teoria valurilor și detaliez deciziile pe care este construită strategia de tranzacționare. Aceasta este alcătuită în așa fel, încât parametrii de tranzacționare să nu influențeze deciziile luate pe baza recunoașterii valurilor.

În următorul capitol prezint programul și parametrii de funcționare, precum și procesul de realizare al acestuia. Programul poate fi folosit în platforma de tranzacționare Metatrader 4.

Mai departe am sintetizat rezultatele obținute. Testul strategiei l-am efectuat pe patru indici diferiți, iar succesul obținut poate fi văzut în tabelul din capitolul 4.

În ultimul capitol am vorbit despre posibilitățile de dezvoltare al sistemului, precum schimbarea timpului folosit la recunoașterea valurilor sau schimbarea parametrilor de tranzacționare.

***Cuvinte cheie:*** Elliott, val, serie de timp, interpolare lineară, automatizat, tranzacționare

**SAPIENTIA ERDÉLYI MAGYAR TUDOMÁNYEGYETEM  
MŰSZAKI ÉS HUMÁNTUDOMÁNYOK KAR, MAROSVÁSÁRHELY  
SZOFTVERFEJLESZTÉS SZAK**

**Elliott-hullámok automatizált felismerésén alapuló részvény  
kereskedési automata**

**Mesteri disszertáció**

**Témavezető:  
Dr. Kupán A. Pál  
egyetemi docens**

**Végzős hallgató:  
Keresztesi Tekla**

## KIVONAT

Dolgozatom célja az Elliott-hullámok felismerésének vizsgálata a tőzsdei árakban és egy erre épülő kereskedési stratégia sikerességének felmérése. Az Elliott féle hullámelmélet megfigyelésen alapuló szabályrendszer, mely a tőzsdei árak mozgását hivatott rendszerezni. Ralph Nelson Elliott az 1930-as években figyelte meg a tőzsdei mozgások szabályszerűségét, és hullámelméletét mai napig használják a tőzsdei kereskedés során. Legsikeresebben indexekre alkalmazható, ezért én is ezen az instrumentumon végzem a felismerést és a kereskedési stratégiát is ennek megfelelően állítottam össze.

Léteznek hasonló céllal létrehozott kereskedési rendszerek, de ezek általában neuronháló segítségével hoznak döntéseket, ami a kereskedési stratégiát átláthatatlanná teszi, ez pedig a legnagyobb kockázat az a személy számára, aki egy meglévő rendszert használ.

A dolgozatom első részében bemutatom a hullámelmélet szabályait és kifejtem, hogy a kereskedési stratégia milyen feltételek szerint működik. A kereskedési stratégiát úgy alakítottam ki, hogy minél kisebb mértékben befolyásolják a kereskedési paraméterek a hullámfelismerés alapú döntéseket.

Az ezt követő részben bemutatom a létrehozott programot és ennek működési paramétereit, valamint a megvalósítás folyamatát. A programot a Metatrader 4 kereskedési platform segítségével lehet futtatni.

Ezután az elért eredményeket írom le. A stratégia sikerességének vizsgálatát négy indexen végeztem el, az összefoglalt táblázatban szemléltettem az eredményeket.

Az utolsó fejezet a továbbfejlesztési irányokról szól, a felismerés időbeosztásának kezelésétől a kereskedési stratégia paramétereinek változtatásáig.

**Kulcsszavak:** Elliott, hullám, idősor, lineáris interpoláció, automata, kereskedés



**SAPIENTIA HUNGARIAN UNIVERSITY OF TRANSYLVANIA  
FACULTY OF TECHNICAL AND HUMAN SCIENCES  
DEPARTMENT OF MATHEMATICS-INFORMATICS  
SPECIALIZATION SOFTWARE DEVELOPMENT**

**A Stock Trading System Based on Automated Identification  
of Elliot Waves**

**Master's Thesis**

**Supervisor:  
Dr. Kupán A. Pál**

**Student:  
Keresztesi Tekla**

## ABSTRACT

The aim of my thesis is to test the recognition of Elliott waves in the stock market and to create an automatic trading system which makes trades based on the recognized waves. The Elliott Wave Theory is a guideline based on observation, which systemizes the movement of stock market price fluctuation. Ralph Nelson Elliott observed the rules by which the prices moved in the 1930s and his wave theory is still used successfully in trading. It is most successfully applied to indexes, that is why I used indexes when performing the recognition and creating the trading strategy.

Similar trading systems exist, but they usually use neural networks which makes the decisions made by the trading system unclear, which is a huge risk for someone using an existing trading system.

In the first part of my thesis I have presented the rules of the wave theory and presented the rules of the trading. I created the trading strategy so that the trading parameters do not interfere with the decisions made based on the recognized waves.

In the next part I have presented the program and its functional parameters, as well as the creation process. The expert advisor can be used in the Metatrader 4 trading platform.

Next, I have synthesized the results. The efficiency of the trading system can be viewed in chapter 4.

In the last chapter I have talked about the possibilities of further development, ranging from changing the time period of the trading to changing the parameters of the trading strategy itself.

***Keywords:* Elliott, wave, trading, time series, linear interpolation, system**

# Tartalomjegyzék

|  |           |
|--|-----------|
| <b>1. fejezet</b>  | <b>13</b> |
| Bevezetés  | 13        |
| 1.1. Történelmi áttekintés   | 14        |
| 1.2. Kapcsolódó kutatás  | 15        |
| 1.3. Célkitűzés  | 15        |
| <b>2. fejezet</b>  | <b>17</b> |
| Elméleti megalapozás   | 17        |
| 2.1. Alapfogalmak  | 17        |
| 2.2. A hullám irányelv   | 17        |
| 2.2.1. Az 5-hullámos minta   | 18        |
| 2.2.2. Alap hullám típusok   | 18        |
| 2.2.3. Jelölési konvenció  | 19        |
| 2.2.4. Kiterjesztett hullámok  | 19        |
| 2.2.5. Megcsonkított hullám  | 20        |
| 2.2.6. Elliott-hullámok és a Fibonacci-számok                            | 21        |
| 2.3. A tesztelt kereskedési stratégiák                                   | 21        |
| <b>3. fejezet</b>  | <b>24</b> |
| Rendszer specifikáció és architektúra                                    | 24        |
| 3.1. A MetaTrader 4 bróker platform                                      | 24        |
| 3.2. A módszer lépései   | 25        |
| 3.3. A legkisebb négyzetek módszer alkalmazása a szakaszok előállítására | 26        |
| 3.4. Elliott-hullámok felismerése  | 28        |
| 3.5. A kereskedési stratégia implementálása és tesztelése                | 31        |
| 3.5.1. A WaveSeq osztály   | 31        |
| 3.5.2. A hullámfelismerő modul   | 31        |
| 3.5.3. A kereskedő modul   | 32        |
| 3.5.4. A program állapotdiagramja  | 33        |
| 3.5.5. A kereskedési stratégia tesztelése                                | 34        |
| 3.6. Felmerült nehézségek és megoldásaik                                 | 35        |
| 3.6.1. Az egymás után következő szakaszok irányának problémája           | 35        |
| 3.6.2. Nagyobb időbeosztású hullámok megjelenítése kisebb időbeosztáson  | 36        |
| 3.6.3. A kereskedési stratégia megválasztása                             | 38        |

## **Tartalomjegyzék**

|                                    |           |
|------------------------------------|-----------|
| <b>4. fejezet</b>                  | <b>40</b> |
| Eredmények és következtetések      | 40        |
| 4.1. Eredmények                    | 40        |
| 4.2. Következtetések               | 43        |
| 4.3. Továbbfejlesztési lehetőségek | 43        |
| Ábrák jegyzéke                     | 45        |
| Táblázatok jegyzéke                | 46        |
| Irodalomjegyzék                    | 47        |
| Függelékek                         | 49        |

# 1. fejezet

## Bevezetés

A tőzsdei világban igen sokat emlegetett Elliott-hullámok elmélete az egyik olyan elemző eszköz, aminek a működése minden időben igazolható volt. Ez a jelenre is igaz.

Ralph Nelson Elliott az 1930-as években állapította meg, hogy a tőzsdei árak egy felismerhető minta alapján mozognak egy adott irányba és húzódnak vissza. A felismert minta formailag ismétlődő, de időben és erősségben nem. Elliott öt ilyen mintát határozott meg, hullámoknak hívta őket, amik a piaci árakon többször előfordulnak. Elnevezte, meghatározta és szemléltette a mintákat és variációikat. Majd leírta, hogy miként kapcsolódnak egymáshoz, hogy nagyobb mintákat alkossanak, majd azok is nagyobb mintákat alkossanak, és így tovább, egy strukturált előre haladó mozgást alakítsanak. Ezt a jelenséget a **hullám irányelv**nek nevezte el [1] (Robert Rougelot Prechter, A. J. Frost, 2005).

Eredetileg kézzel készített ábrákon próbálták meghatározni az Elliott-hullámokat a tőzsdei árakból, később külön programokat hoztak létre erre a célra. Mivel ma már maga a kereskedés is otthonunk kényelmében történik, a brókerek által kiadott programok segítségével, és ezek azt is lehetővé teszik, hogy az elemzést a valós árakon elvégezzük. Sőt, mi több, a népszerűbb platformokon saját programjainkat is futtathatjuk, amik elemzést vagy automatizált kereskedést végeznek.

Habár a technikai körülmények adottak az elemzéshez, a hullámok meghatározása így is kihívást jelent, főleg a kezdők számára. Az elvárás általában az, hogy egy pontos szabályrendszer alapján tudjuk meghatározni a hullámok dimenzióit az ábrákon. Ha ez valóban lehetséges lenne, akkor egyrészt mindenki ezt használná, másrészt pedig már létezne olyan program, ami ezt a feladatot elvégzi. Az Elliott hullámelmélet alkalmazásában pontosan az jelenti a kihívást, hogy nincs egy teljesen pontosan meghatározott szabályrendszere. Megfigyeléseken alapszik, néhány szabályt felállít ugyan, de a pontosítást inkább irányelvek formájában határozza meg. Elvégre ezért hívta Elliott elméletnek és nem algoritmusnak.

Az elmélet alkalmazása akkor lesz hasznos egy kereskedő számára, ha ezeket a laza szabályokat elfogadja. A múlt adatai alapján megfigyelhetjük, hogy miként mozognak az Elliott

hullámok egy adott instrumentumon, és ez alapján következtethetünk, hogy a jelenben mi történik.

A hullámok automatizált felismerésében pont az jelenti a legnagyobb kihívást, hogy nincsenek konkrét szabályok. Erre megoldásként az szolgál, hogy a technikai lehetőségeket kihasználva, a különböző időbeosztások hullámait összehasonlítsuk. A legjobb pozitív visszacsatolás a hullámelméletben, ha egy kitörési hullám egy adott időbeosztáson kisebb kitörési hullámokból áll, a kisebb időbeosztáson. Az elemző programban azt igyekeztem szem előtt tartani, hogy a különböző időbeosztásban megjelenő szakaszokat egymással harmonizáljanak.

A program a felismert hullámok alapján kereskedik. A döntést egyszerű feltételek alapján hozza meg, szem előtt tartva a felismert állapotot és a potenciális nyereség/veszteség arányt. A döntés amiatt nem történik mesterséges intelligencia alapján, mert nagy annak a veszélye, hogy a betanított neuronháló a múltban jól teljesítsen, a múlt adataira legyen optimalizálva, a jelenben pedig, más körülmények között rossz döntéseket hozzon.

## **1.1. Történelmi áttekintés**

A tőzsdei árakat általában az idősor analízis eszközeivel próbálják előrejelezni, majd az elemzés eredménye építeni egy kereskedést stratégiát. Ez azt jelenti, hogy a meglévő adatok alapján egy következtetést vonnak le a jövő adatokra vonatkozóan, és összefüggést keresnek a múlt és a jövő adatai között. Az idősor négy része a trend, a ciklus, a szezonális ingadozás és a véletlen ingadozás [3] (Michelberger P., Szeidl L., Várlaki P.). Ha megfigyeljük ezeket az elemeket és összevetjük az Elliott-hullámokkal, akkor mindenre találunk megfeleltetést:

- a trend, hogy milyen irányba mozog a piac, milyen irányba haladnak a mozgató hullámok,
- ciklikus, mivel több hullám követi egymást
- a szezonális ingadozás ott ismerhető fel, hogy a hullámok különböző árterületen jelentkeznek
- a véletlen ingadozás pedig a javító hullámok formájában jelentkezik

Elliott megfigyelés alapján azonosította a fenti összefüggéseket és olyan következtetéseket vont le, amiket az 1930-as évektől napjainkig használnak.

Az elején az elemzés hosszú időt vett igénybe, papíron történt, és nem állt annyi adat az elemzők rendelkezésére. Ehhez képest ma minden lehetőség megvan rá, hogy az elemzést

számítógépen végezzük el, több évre visszamenőleg rendelkezésünkre állnak az adatok és ezt a folyamatot automatizálhatjuk is. Sok hasonló rendszer létezik is már, ami az Elliott féle hullámelméletet használja a tőzsdei előrejelzés alapjául.

## 1.2. Kapcsolódó kutatás

Egyik ilyen rendszer, a WASP (Wave Analysis Stock Prediction<sup>1</sup>) a neuronhálók és fuzzy logika segítségével végez előrejelzéseket. A neuronhálót naponta újra tanítják, egy megbízhatósági indexet használ, hogy megállapítsa, hogy mennyire biztos az előrejelzés. A kiértékelést 400 napos időperióduson keresztül végezték, és egy egyszerű “Buy and hold”<sup>2</sup> stratégiát sikerült túlteljesítenie. Az eredmények azt mutatják, hogy a 10.000 dolláros teszt számlán 400 nap alatt körülbelül 80%-os nyereséget sikerült elérnie [5] (George S. Atsalakis, Emmanouil M. Dimitrakakis, Constantinos D. Zopounidis). Az eredmény hitelessége megkérdőjelezhető, mivel megtörténhet az, hogy rövid időn belül, vagy a kiválasztott időszakon belül jól teljesít az rendszer, de hosszú távon veszteséges lenne a stratégia.

Egy másik hasonló kutatás a jelzések alapjaként a technikai analízist vette, azaz az árak mozgását különböző mutatók viselkedésével próbálja megjósolni, és ez egészül ki az Elliott-hullámok jelzésével. Ez nem kereskedési rendszer, hanem csak jelzéseket szolgáltat, hogy mikor milyen irányba fog mozogni a piac [4] (M. Tirea, I. Tandau, V. Negru).

Egy Metatrader környezetben használható rendszer nem végez kereskedést, hanem csak a hullámok felismerését implementálja [6] (R. Martiruk).

A piacon rengetek olyan kereskedelmi rendszer létezik, amely az Elliott-hullám elméletre alapuló előrejelzést vagy kereskedést ígér, de egyrészt nem elérhető áron vásárolható meg, másrészt pedig nem vállalnak semmilyen garanciát arra, hogy a rendszer valóban jól működik. A neuronháló alkalmazása pedig általában áttekinthetlenné teszi a kereskedési stratégiát.

## 1.3. Célkitűzés

Korábban egy olyan rendszerrel próbálkoztam, ami neuronhálót használt az árak mozgásának előrejelzésére, viszont nem bizonyult elég megbízhatónak a rendszer. Egyes

---

<sup>1</sup> Hullám analízis alapú részvény előrejelzés

<sup>2</sup> Buy and hold stratégia: részvény megvásárlása és hosszú távon való megtartása.

esetekben a nyereség nagy volt, más esetekben viszont a veszteség túl nagy volt. A döntési folyamat és a belépési pont sem volt átlátható.

A neuronhálóra épülő rendszerek nagy hátránya (és veszélye) az, hogy a tanítás során a tesztelési periódusban jól viselkedik a rendszer, de túl rá van hangolva a tanításhoz használt adatokra, más helyzetben pedig nem hoznak jó eredményt.

Célom egy olyan rendszer létrehozása, ami egyszerű, átlátható döntések alapján kereskedik és felügyelhető. Azt szeretném vizsgálni, hogy kizárólag az Elliott-hullámok alapján meghozott döntések mennyire megbízhatóak, lehet-e ez alapján döntést hozni, és ennek a döntésnek milyen eredménye lesz.



## 2. fejezet

### Elméleti megalapozás

Az elméleti rész megértéséhez szükségesek egyes tőzsdei alapfogalmak ismerete, ezeket a következőkben fogom ismertetni.

#### 2.1. Alapfogalmak

**Gyertya:** a tőzsdén gyertyákkal jelölik egy adott időperióduson belüli ár változását. Ennek van nyitási, zárási értéke, maximum és minimum értéke. Példaként a május, 2015-höz tartozó havi gyertya nyitóára a május elsejei nyitási ár, a zárási ára a május 31.-i zárási ár, a maximum és a minimum pedig e két dátum közötti maximum, illetve minimum érték.

**Időbeosztások:** a tőzsdei adatokat különböző időbeosztásban jeleníthetjük meg egy grafikonon, ez a legnagyobbtól a legkisebbig lehet: havi, heti, napos, 4 órás, 1 órás, 30 perces, 15 perces, 5 perces, 1 perces vagy minden egyes adatot megjelenítő.

**Trend:** az általános irány, amerre mozog a piaci ár, ez lehet növekvő vagy csökkenő.

**Leverage:** tőkeáttét. A tőzsdeszámlák esetében a bróker hitelezőként működik, és egy pozíció megnyitásához nincs szükségünk a teljes kereskedésben használt összegre, hanem csak egy letétre. A kereskedés során a nyereség vagy veszteség áttéttel jelenik meg.

**Pip:** pont, a legkisebb egység, amennyivel az ár elmozdulhat.

**Long:** hosszú pozíció, először vásárolunk, majd eladunk. A cél, hogy alacsonyabb áron adjunk el, magasabban vásároljunk. Növekvő trendben nyitunk hosszú pozíciót.

**Short:** rövid pozíció, először eladunk, majd utána vásárolunk. A cél, hogy magas áron adjuk el, majd alacsony áron vásároljunk. Csökkenő trendben nyitunk rövid pozíciót.

**Stop loss:** nyitott pozíciónk esetén a stop loss szintnél vágjuk el a veszteséget.

**Take profit:** ennél a szintnél zárjuk le a pozíciót nyereségben.

#### 2.2. A hullám irányelv

A hullám irányelv megfigyelésen alapszik, a technikai elemzés egy eszköze. Azt feltételezi, hogy a tőzsdei árak minden addig elérhető információt tartalmaznak. A fundamentális elemzőkkel ellentétben a technikai elemzők azt tartják igaznak, hogy nem a hírek mozgatják a

piacot, hanem az emberek reakciója a hírekre. Ezek a trend változásainak sorozataiból, a továbbiakban hullámoknak nevezett formákat alkotnak.

### 2.2.1. Az 5-hullámos minta

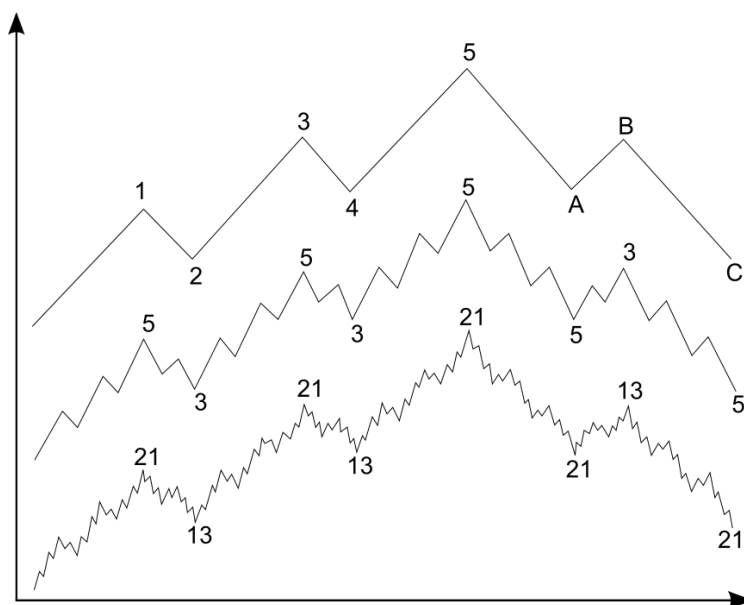
A tőzsdei piacon az árak haladása öt, jellegzetes alakú hullámból áll. Ezekből háromnak, névileg az első, harmadik és ötödik a kihatása az egy irányba történő haladás. Ezek két olyan hullámmal vannak megszakítva, amik a trenddel szembe haladnak, és megszakítják az előrehaladást. Ezek a kettes és négyes hullám. Ezek szükségesek a trend előrehaladásához.

Elliott három fontos szabályt állított fel az 5 hullámos mintáról:

- a második hullám sose megy vissza az első hullám elejére
- a harmadik hullám sose a legrövidebb
- a negyedik hullám nem megy az egyes hullám árterületére [1] (R. R. Prechter, Jr., A. J. Frost)

### 2.2.2. Alap hullám típusok

Kétféle alaphullám létezik: mozgató<sup>3</sup> vagy javító<sup>4</sup>. Az mozgató hullámok öt kisebb hullámból állnak, míg a javító hullámok háromból. A teljes hullám mozgás könnyen megérthető az alábbi ábra alapján:



2.1. ábra [9]

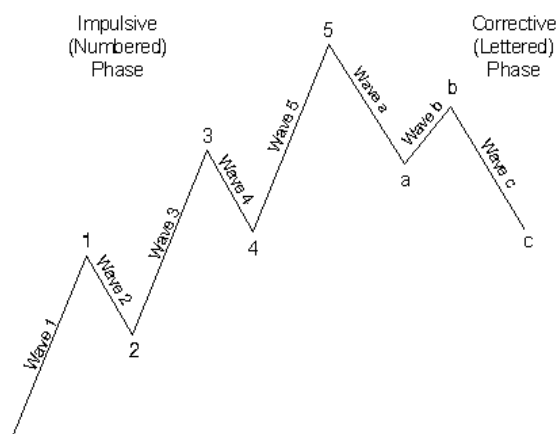
<sup>3</sup> angolul: motive

<sup>4</sup> angolul: corrective

Az ábrán a legfelső hullám egy teljes 8-as ciklus, a következők pedig már fel vannak bontva kisebb hullámokra. Megfigyelhető, ahogy az mozgató hullámok (az 1-es, 3-as és 5-ös) 5 részből álló hullámra bonthatóak fel, a javító hullámok pedig (2-es, 4-es) pedig három részből álló hullámokra bonthatóak fel kisebb időbeosztáson [1] (R. R. Prechter, Jr., A. J. Frost).

### 2.2.3. Jelölési konvenció

A 2.2-es ábrán a két féle hullám típusra találunk példát. Fontos jelölési konvenció, hogy az előrehaladó hullámokat számokkal jelöljük, a javító hullámokat pedig betűkkel.

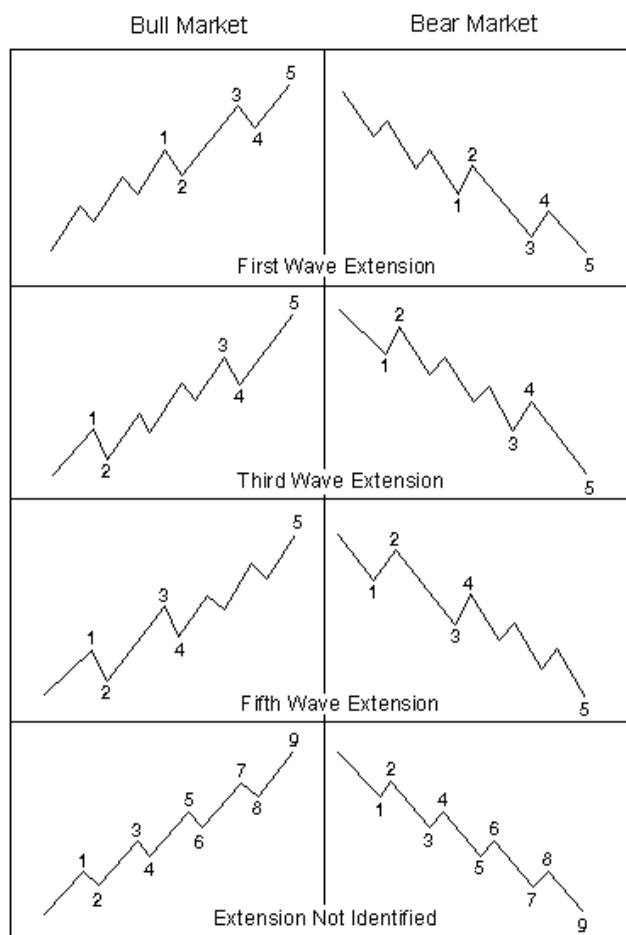


2.2. ábra

### 2.2.4. Kiterjesztett hullámok

A kétféle hullámon kívül megfigyelhetünk **kiterjesztett**<sup>5</sup> hullámokat is. A kiterjesztések csak a mozgató hullámoknál jelentkezhetnek, akár egyszerre többnél is. Leggyakrabban a 3-as hullám lehet kiterjesztett, ez a szabályokból következik, miszerint a 3-as sosem lehet a legrövidebb hullám, a 4-es pedig nem lép be az 1-es hullám ártartományába [1] (R. R. Prechter, Jr., A. J. Frost). A 2-es ábrán a kiterjesztéseket figyelhetjük meg. A feltételezéseim alapján azt érdemes szem előtt tartani, hogy a hullámok pontos beosztása nem is annyira fontos, a mozgató és javító hullámok elejének és végének a lehető legpontosabb meghatározása a cél.

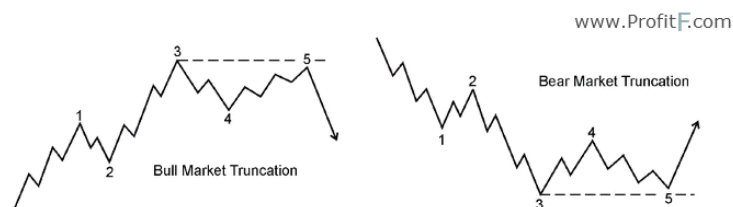
<sup>5</sup> angolul: extension



2.3. ábra

### 2.2.5. Megcsonkított hullám

A következő hullám típus a **megcsonkított**<sup>6</sup>, amely az ötödik hullám esetén fordul elő. Úgy lehet ellenőrizni, hogy valóban az ötödik hullám ez, hogy felbontva kisebb részekre, kiderül, hogy mozgató hullámról van szó, vagy ő maga is 5 kisebb hullámból áll. Viszont, amint a 2.4.-es ábrán is megfigyelhető, nem sikerül túllépnie a 3-as hullám árterületét [1] (R. R. Prechter, Jr., A. J. Frost):



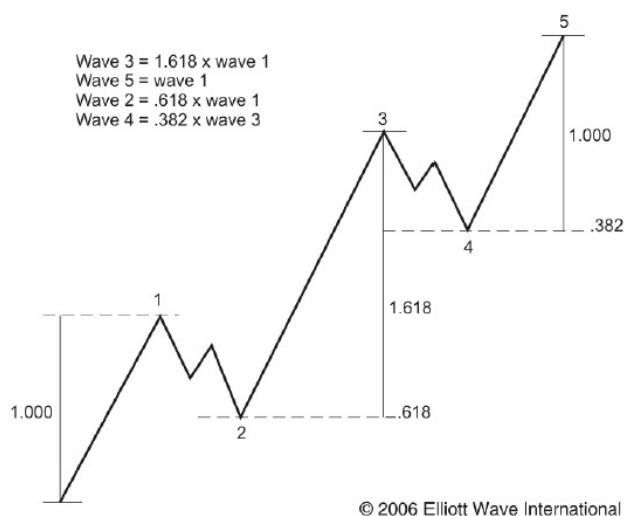
2.4. ábra

<sup>6</sup> angolul: truncation

### 2.2.6. Elliott-hullámok és a Fibonacci-számok

Az eddigiekben összefoglaltam a hullámok formáját, de azok egymáshoz viszonyított méretéről még nem esett szó.

Elliott azt állította, hogy a hullámelmélete alapját a Fibonacci-számok jelentették. A hullámok számosságai is Fibonacci-számok: 5 részből áll a mozgató, 3 részből a javító, összesen 8 részből egy egész hullám. A hullámok méretei pedig a Fibonacci-számokkal arányosak. A 2.5. ábra ezt szemlélteti:



2.5. ábra

Az ábrán természetesen egy ideális helyzet jelenik meg. Az általános alapelv azt mondja ki, hogy ha az 1-es hullám nem kiterjesztett, akkor a 4-es hullám egy mozgató alakzatot pont az aranymetszet arányában fogja elválasztani. Ha pedig az 5-ös hullám kiterjesztett, akkor az arányok megfordulnak. A Fibonacci-számok nem csak a hullámok magasságában figyelhetők meg, hanem egy-egy hullám időtartama is arányos a Fibonacci-számokkal [1] (R. R. Prechter, Jr., A. J. Frost).

### 2.3. A tesztelt kereskedési stratégiák

Céлом nem csupán a hullámok minél pontosabb felismerése, hanem az eredmények felhasználása egy kereskedési stratégia kialakításában és automatizálásában. A stratégia implementálásában több olyan paraméter van, amit hangolni lehet különböző helyzetekre. A

dolgozat céljába nem tartozik bele ezen paraméterek optimalizálása, hiszen az valamilyen szinten az adott tesztadathoz való hangolást jelentené. Ezek a paraméterek változtathatóak, különböző indexeken jobb vagy rosszabb eredményeket fogunk kapni ennek függvényében.

A teszteket indexek tőzsdei árfolyamán fogom elvégezni, mivel a tapasztalat azt mutatja, hogy az Elliott hullámok ezeken az instrumentumokon ismerhetőek fel a legkönnyebben. Ezek árfolyama a legstabilabb, mivel több cég vagy termék árfolyamából tevődik össze, ezért stabilabban is mozognak.

A kereskedési stratégia négy fontos részből tevődik össze, éspedig: időbeosztás, belépési pont, maximális veszteség szintje és az elvárt nyereség szintje.

Az időbeosztást két szempontból is meg kell, hogy határozzam. Az első, hogy a hullámokat melyik időbeosztáson ismerem fel, a második pedig, hogy milyen időbeosztású grafikonon fogom futtatni a programot. A hullámok időbeosztásának választása a hetesre esett, mivel az adatoknak a hetes grafikonon nincsenek nagy kilengései, ezért elég pontosan fel lehet ismerni a hullámokat. A program futtatása pedig napos grafikonon fog történni. Ez azt jelenti, hogy minden nap nyitásakor fut le a hullámfelismerő algoritmus és naponta egyszer próbál pozíciót nyitni a kereskedési automata.

A stratégia belépési pontját a hullámok felismert állapota fogja adni: a leghosszabb hullámok a 3-as és az 5-ös, ezek előrehaladó hullámok, és ezek előtti hullámok felismerésekor fogok pozíciót nyitni. Természetesen abba az irányba, amelybe a felismert hullám fog haladni.

A stratégiához egy egyszerű kockázatmenedzsmenti részt implementálok, 10.000 dolláros demó számlával fogok tesztelni, és minden pozíciót egy lot mennyiséggel nyitok meg. A maximális veszteséget a következő logika szerint határozom meg:

- **vétel esetén** a napi grafikonon az aktuális nap előtti nap mélypontja alatt legyen 100 ponttal, vagy ha az azelőtti napi árterjedelem magasabb, akkor pontosan 100 ponttal a nyitási ár alatt
- **eladás esetén** pedig a napi grafikonon az aktuális nap előtti nap csúcspontja fölött legyen 100 ponttal, vagy ha az azelőtti napi árterjedelem alacsonyabb, akkor pontosan 100 ponttal a nyitási ár fölött határozom meg.

Minden kereskedésnél meghatározom azt a szintet, aminél elvágom a nyereséget. A nyereség/veszteség szintek arányát 1:6-nak határozom meg. Így arra törekszem, hogy ha a

nyereséges pozíciók száma kisebb is, mint a veszteségesé, összesítve a nyereség értéke nagyobb legyen, mint a veszteségé.

A belépési stratégiának két variációját fogom vizsgálni. Az **első egy kockázatosabb**, amikor kizárólag egy időbeosztást figyelek, és az azon felismert hullámok szerint lépek be. A **második, kisebb kockázatú** stratégia esetén azt is vizsgálom, hogy a kisebb időbeosztású adatokon felismert hullámok egy irányba mozognak-e a legnagyobb időbeosztás trendjével.

### 3. fejezet

#### Rendszer specifikáció és architektúra

A kereskedő automata egy programozható bróker platformon készült el. Ennek az a nagy előnye, hogy valós adatokkal dolgozhatunk demó számlán. Lehetőség van arra is, hogy az időben visszamenőleg is teszteljük a programot.

##### 3.1. A MetaTrader 4 bróker platform

Az általam választott bróker platform a MetaTrader 4, ami C alapú, mql4 nevű nyelven programozható. A program népszerűségére való tekintettel és a programozhatóság egyszerűsége miatt választottam ezt. A platformnak csak azon részeit fogom a következőkben ismertetni, amelyeket használtam a program elkészítésében.

A MetaTrader 4 kliens terminál saját fejlesztésű szerkesztő használatát biztosítja. A programokban a régi és aktuális adatokhoz a különböző mql4 függvényeken keresztül férünk hozzá. Négy típusú fájlt hozhatunk létre [11] (MQL4 Reference):

- szaktanácsadó program (Expert Advisor)
- személyre szabott mutató (Custom Indicator)
- szkript (Script)
- könyvtár (Library)

Az általam írt program szaktanácsadó típusú, amely segítségével egy kereskedési stratégiát implementálhatunk. Ez egy adott grafikonhoz (Chart) kapcsolódik. A szaktanácsadói program különböző részei akkor hajtódnak végre, mikor egy adott esemény bekövetkezik. Az események lehetnek [7] (MQL5 Limited):

- Init - a program belépési pontja
- Deinit - a program kilépési pontja
- NewTick - új adat érkezett
- Timer - a programban definiált időzítő esemény
- Tester - historikus adatokon való tesztelés után bekövetkező esemény
- ChartEvent - grafikon mozgás, méretezés esemény



Az eseményeket a következő esemény kezelő függvényeken keresztül tudjuk figyelni [8] (MQL5 Limited):

- void OnInit()
- void OnDeinit(const int reason)
- void OnTick() - új adat érkezik, NewTick eseményt kezeli
- void OnTimer()
- double OnTester()
- void OnChartEvent(...)

### 3.2. A módszer lépései

Mivel az adatokat gyertyákként kapom meg, ami négy diszkrét adatot jelent, ezért át kell alakítsam őket idősorra, így a zárási adatokat használom az elemzéshez.

Első lépésben a legnagyobb (havi) időhöz tartozó adatokat felvágom szakaszokra, a Gauss féle regressziós illesztés módszerrel. Ugyanezt elvégzem a hetes adatokra is, majd a különböző periódushoz tartozó szakaszok végeit összeillesztem. Erre azért van szükség, mivel a havi adat 30 gyertyát foglal magába, ha hetes időbeosztáson nézem a szakaszokat, akkor ezek végei nem érnek össze, minden két szakasz között marad egy hónapnyi üres idő. Az összeillesztést a felmerült problémák és azok megoldásaik cím alatt tárgyalom.

Második lépésként a havi adatban 1-2-3 felépítésű hullámokat keresek. Ha nem találtam 1-2-3-as alakú hullámot, akkor a módszer nem alkalmazható erre az adatsorra. Vagy nincs elég adat vagy annyira változó az adat, hogy nem lehet elég jó felismerést végezni.

Ha találtam 1-2-3 felépítésű hullámokat, akkor kiválasztom időben a legutolsónak a kezdőpontját, és onnan próbálom tovább építeni a hullám sorozatot. Ezzel az a cél, hogy meghatározzam, hogy a jelen pillanatban, vagyis a hullám sorozat utolsó eleme után mi következhet, milyen állapotban lehetek. Egy egyszerű szabállyal határozom meg a hullám típust, amiben vagyok, ami az iránya szerint lehet növekvő vagy csökkenő, a sebessége szerint pedig mozgató (motive) vagy javító (corrective). Ha az előző hullám csökkenő volt, akkor a mostani növekvő lesz, másképp a szakaszokat előállító algoritmus egybeolvasztotta volna a szakaszokat. A hullám elmélet leírása szerint pedig az 1-es, 3-as és 5-ös hullámok lesznek az mozgóak, az összes többi pedig javító hullám lesz.

A legnagyobb elérhető időbeosztást arra használok, hogy meghatározzam azt az általános irányt, amibe fog haladni a piac, mivel a második stratégia szerint nem szeretnék a trenddel szemben kereskedni.

### 3.3. A legkisebb négyzetek módszer alkalmazása a szakaszok előállítására

#### A legkisebb négyzetek módszer [10] (Algorithm for Finding the Best Fit Line):

Keressük azt az egyenest, amely az X-Y síkban elhelyezkedő pontokra a legjobban illeszkedik. Az egyenes egyenlete:

$$y = a \cdot x + b, \text{ ahol}$$

$$a = \frac{\Sigma(x \cdot y) - \frac{\Sigma(x) \Sigma(y)}{\Sigma 1}}{\Sigma(x^2) - \frac{(\Sigma(x))^2}{\Sigma 1}}$$

és

$$b = \frac{\Sigma y}{\Sigma 1} - a \cdot \frac{\Sigma x}{\Sigma 1}$$

A tőzsdei árakra való alkalmazásban az x-tengelyen az adott időegységhez tartozó gyertya indexét jelenti, az y adatot pedig az adott időegység alatti átlagolt ár. Az átlagolt ár az időegységen belüli legkisebb és legnagyobb ár átlaga.

A szakaszok felosztásához, minden időbeosztásban meghatározok egy minimális hosszúságot:

| Időfelosztás              | Minimális hosszúság (gyertya) |
|---------------------------|-------------------------------|
| Havi, heti, napos, 4 órás | 3                             |
| 1 órás                    | 5                             |
| 30 perces, 15 perces      | 8                             |
| 5 perces, 1 perces        | 13                            |

3.1. táblázat

Ezután kiindulok a teljes hosszúságú idősorból, meghatározom a rá legjobban illeszkedő szakaszt, aminek a négyzetes hibája lesz a kiinduló érték a minimumkereséshez. A hibát a következő képlettel számítom ki:

$$h = \Sigma((a \cdot x + b - y)^2)$$

Azért van szükség a négyzetre emelésre, mivel az eltérések különböző előjelűek lesznek, és ha csak az összegüket számolnám, akkor ezek semlegesítenék egymást.

Miután megkaptam a teljes hosszúságú szakaszra a hibát, akkor elindulok az idősor elejétől, és megnézem, a megfelelő minimális hosszúságú szakasztól felfele haladva, hogy ha két szakaszra választanám az adott pontban a teljes idősort, akkor a szakaszok együttesen vett négyzetes hibája kisebb lenne-e a teljes szakasz négyzetes hibájánál. Ha igen, akkor levágom az elejéről a szakaszt és a következő iteráció kezdőpontja a levágott szakasz utáni első index lesz. Ezt addig végzem, amíg vagy már akkora lesz a szakasz, hogy nem tudom feldarabolni, vagy darabolással nem kapok kisebb négyzetes hibát.

Ez a módszer általában kicsi szakaszokat eredményez, amik gyakran azonos előjelű iránytényezővel rendelkeznek. Ezért úgy folytatom a szakaszok felosztásának algoritmusát, hogy végigmegyek a szakaszokon és az azonos monotonitású szakaszokat lineáris interpolációval összevonom. Az összevonást a következőképpen végzem: az első szakasz kezdőpontja lesz az új szakasz kezdőpontja, a második szakasz lesz az új szakasz végpontja és az új kezdő és végpontnak megfelelően meghatározom a rá legjobban illeszkedő szakaszt. A módszer szemléltetése lépésenként: az 3.1. ábrán látható, amint az NQ100 index havi felbontású adataira elvégeztem a szakasz illesztést. Ennek következményeként olyan szakaszokat illesztettem az adatra, amelyek irányai felváltva követik egymást.

A grafikonon egy gyertya alakzat egy havi adatnak felel meg. A fehér szakaszok formájában jelenik meg a szakaszra bontás.



3.1. ábra

Ami után a havi adatokon meghatároztam a szakaszokat, elvégzem a felbontást a heti adatokra is. Ez a 3.2. ábrán látható. Itt a fehér vonalak a havi adatok felbontását jelentik, a piros vonalak pedig a heti adatok felbontását.



3.2. ábra

### 3.4. Elliott-hullámok felismerése

Következő lépésként elvégzem az állapotok felismerését. Első lépésben 1-2-3 hullámokat keresek. Ehhez kiszámítom minden szakasznak az időbeni hosszát, vagyis meghatározom minden szakasz esetén, hogy hány gyertyát ölel át. Nevezzük ezt a következőkben időbeni hosszúnak. Ez arra fog szolgálni, hogy azt is vizsgáljam, hogy az 1-es hullám ne legyen túl rövid.

A következő szabályokat alkalmazom az 1-2-3 hullámok felismeréséhez:

- az 1-es hullám időbeni hossza nagyobb, mint az átlag időbeni hossz fele
- a 2-es hullám kialakulásáig nem megy vissza az 1-es hullám elejére
- a 3-as hullám nem a legrövidebb e három hullám közül.

Ha a fenti szabályok teljesülnek, akkor felismertnek tekintem az 1-2-3 alakzatot. Ennek eredménye látható a 3.3. ábrán:



3.3. ábra

Az 1-2-3 hullám alakzat kezdeti pontját C-vel jelölöm. Ennek csak abból a szempontból van jelentősége, hogy ha a hullámok A-B-C és 1-2-3-4-5 alakzatai váltogatják egymást, akkor feltételezem, hogy C-vel végződött az 1-2-3 előtti javító hullám. Mivel a javító hullámokat nehezebb követni és a kereskedési stratégiának nem képezik részét, ezért ezeknek a felismerése nem tartozik a dolgozat célkitűzései közé. Mivel a javító hullámokról csak akkor derül ki, hogy javítóak, hogyha nem alakul ki a 4-es, 5-ös hullámuk, de a kereskedési stratégia szerint a 3-as hullámon is akarunk kereskedni, ezért a felismert állapotok között lesz olyan 1-2-3-as hullám, amelyik tulajdonképpen A-B-C alakzat. Mivel ez csak utólag derül ki, ezért a kereskedési stratégia szempontjából nem fontos.

Miután megvannak az 1-2-3-as alakzatok, megpróbálom kiterjeszteni 1-2-3-4-5-ös állapotokkal is. A hullámok felismerését elvégzem a heti adatokra is. A 4-es és 5-ös hullámok vizsgálatát az alábbi logika szerint végzem:

- a 4-es hullám nem kerülhet be az 1-es hullám árterületére
- az 5-ös hullám a 3-as hullám végső árán túl kell haladjon

Természetesen az egymásutáni hullámok alap követelménye az, hogy két egymást követő szakasz monotonitása különbözzön, vagyis az iránytényezője ellentétes előjelű legyen.

Az alábbi két ábrán már teljes C-1-2-3-4-5 felismert sorozatokat figyelhetünk meg:



3.4. ábra

Nagyon szépen követhető a nagyobb időbeosztáson felismert hullámok felépítése a kisebb időbeosztáson. A fehér, havi adathoz tartozó 3-as szakasz egy piros 1-2-3-4-5 sorozatból áll a heti adatok alapján. Ez azt is bizonyítja, hogy a fehér hármas valóban egy előrehaladó hullám. A 3.5. ábra időben a 3.4-es folytatása.

A 3.5-ös ábrán megfigyelhető, hogy miért is mondta Elliott, hogy nem szigorú szabályokat határoz meg, hanem irányelveket a hullámok felismeréséhez. Itt ugyanis az figyelhető meg, hogy a fehér 1-es hullám a kisebb időbeosztáson 1-2-3 hullámra bomlik fel, és a fehér 2-es hullám egybeesik a piros 4-es hullámmal. Utána a fehér 3-as a piros sorozatban az 5-ös hullámmal folytatódik. Tehát az, hogy a nagyobb hullámok kisebbekre oszthatóak csak elvileg értendő, nincs garancia arra, hogy a kisebb hullámok sorozatának kezdő- és végpontjai egybeesnek a nagyobb hullámok kezdő- és végpontjaival. Az magától értetődő következmény, hogy ha ezek valóban egybeesnek, akkor a felismerés folyamata biztosabb lehet. Minél több megerősítést kapunk különböző időbeosztáson, annál pontosabbnak mondhatjuk a felismerést.



3.5. ábra

### 3.5. A kereskedési stratégia implementálása és tesztelése

A Metatrader platform alatt futó szakértői tanácsadó implementáltam a felismerést és a kereskedési stratégiát is. A teljes program több modulra épül:

#### 3.5.1. A WaveSeq osztály

A WaveSeq osztály tartalmazza azokat az adatokat, amelyek a szakaszok felbontásához szükségesek, és a felismert állapotokat is tartalmazzát. Minden használt időbeosztásnak megfelel egy példány a fenti osztályból.

#### 3.5.2. A hullámfelismerő modul

Ez a rész tartalmazza az összes olyan függvényt, ami segítségével elvégzem a legkisebb négyzetek módszerét és kirajzolom a szakaszokat a grafikonra. A hullámok felismerése is ebben a modulban történik.

Több segédfüggvényt használva implementáltam a szakaszokra bontást. Az IT\_GetMinBreakingPosForward függvényben fut le a szakaszokra bontó algoritmus. Ez paraméterként kapja, hogy milyen periódusra végezze a felosztást, mennyi az egy szakaszhoz tartozó minimális adatok száma, illetve az idősor első és utolsó indexét, amelyre a felosztást el kell végezze. Ebben a függvényben hívom meg a ConstructAndGetTrend függvényt, ami

átalakítja az idősort diszkrét adatsorrá. A diszkrét adatsorom x komponense az adott gyertyához tartozó adat indexe lesz a sorban. Az y komponense pedig az adott időponthoz tartozó ár.

Az iteráció során azt vizsgálom, hogy ha a teljes adatsort megtöröm egy i pozíciónál, akkor a két külön szakasz négyzetes hibája kisebb lesz-e, mint a teljes adatsoré. A GetTrend függvényben határozom meg egy adott szakaszra vonatkozó négyzetes hibát. Ez a függvény számítja ki a szakaszok iránytényezőjét és a hozzá tartozó eltolást.

A szakaszokhoz tartozó adatot a törési pontok indexeinek listájában tárolom, és ennek megfelelően tárolom az iránytényezőt és az eltolást is. Ezeket az adatokat használom a szakaszok megjelenítésére is.

A felbontott adatsor szakaszain végigmenve, a fent említett szabályok szerint vizsgálom, hogy melyik szakasz együttes alkothat először 1-2-3 hullámsorozatot, majd ezekből melyek lehetnek 1-2-3-4-5 sorozat kezdőpontjai.

### **3.5.3. A kereskedő modul**

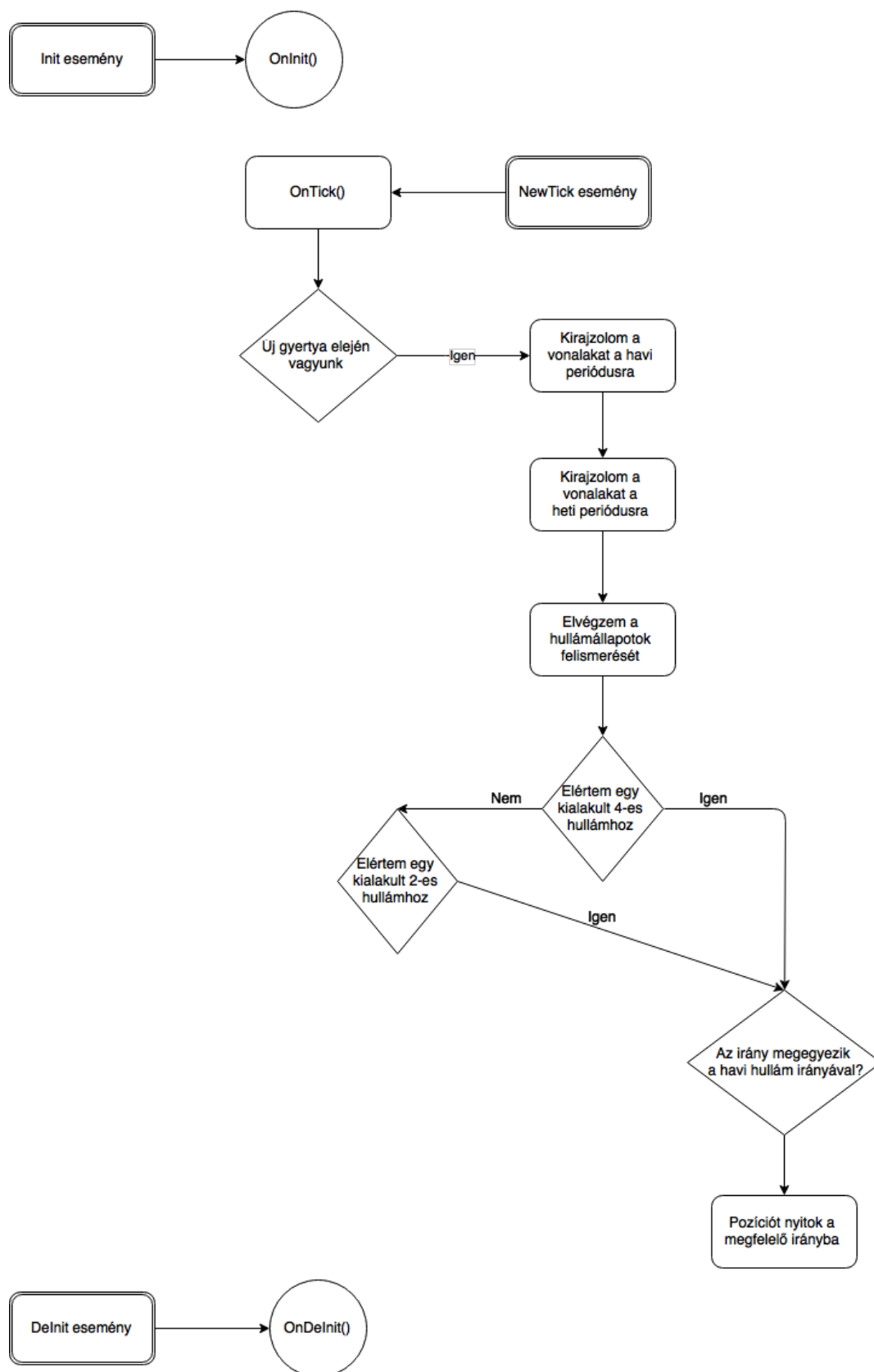
Amint a neve is sugallja, ez a modul tartalmazza a kereskedéssel kapcsolatos függvényeket. Itt nyitok pozíciót és ebben számítom ki a stop loss és a take profit értékét.

Az openNewBuy függvény segítségével hosszú pozíciót nyitok, az openNewSell segítségével pedig rövidet. Mindkét függvény paraméterként kapja az időperiódust, amin történik a kereskedés. E két függvényben standard változók (Ask, Bid) és függvények (iLow, iHigh) segítségével határozom meg a maximális veszteség és elvárt nyereség szintjét. A hosszú pozíciót az Ask árszinten kell nyitni, a rövid pozíciót pedig a Bid szinten. Az iLow és iHigh függvények egy gyertyához tartozó minimális, valamint maximális értéket adja vissza.

A pozíciók lezárása automatikusan történik, a meghatározott stop loss vagy take profit szinten.



### 3.5.4. A program állapotdiagramja



3.6. ábra

A 3.6-os ábrán a program állapotdiagramja látható. A program futása az Init eseménnyel kezdődik, amikor hozzáadjuk a programot egy, a brókerfelületen levő instrumentum grafikonjához. Az Init függvényben elvégzem az inicializálást, ami annyiból áll, hogy a futás idejére egy véletlenszerű számot generálok, amit megadok paraméterként a vonalaknak, így mikor újra kell rajzolni őket, akkor fogom tudni, hogy melyiket kell letörölni.

A vonalak rajzolása és a hullámok felismerése akkor fog végrehajtódni először, amikor új adatot kapok. Ezt a NewTick esemény váltja ki, és az OnTick függvényben kezelem le. Miután elvégeztem a hullámfelismerést, megvizsgálom, hogy a jelenben milyen hullámban vagyok a havi beosztáson. Ezt az adatot csak irányként használom fel, mivel nem soha nem szeretnénk a fő trenddel szemben kereskedni. Ezután megnézem, hogy a heti hullámokban milyen hullámnál tartok. Ha a 4-es hullám végén vagyok, és a következő, 5-ös hullám iránya megegyezik a havi iránnyal, akkor pozíciót nyitok. Stop loss és take profit szinteket állítok be, a piaci mozgás majd végzi a dolgát és a pozíció vagy a meghatározott veszteséggel, vagy a meghatározott nyereséggel fog zárulni. Ha nem 4-es pozícióban vagyok, akkor megvizsgálom, hogy 2-es hullámnál tartok-e. Ha igen, és az irány is megegyezik a havi iránnyal, akkor szintén pozíciót nyitok. Ellenkező esetben várok egy új eseményre.

Bár az OnTick függvény minden egyes új adatra végrehajtódik, ami másodperceken belüli több hívást is jelenthet, a szakaszok újrarajzolását és a kereskedési logikát csak minden egyes nap elején hajtom végre, mivel a kereskedési stratégia hosszú távú, és ha a heti adatok alapján kereskedek, akkor naponta várhatok olyan helyzetre, hogy eleget változzon az adat ahhoz, hogy ne az azelőtti állapotban maradjak.

A program végrehajtása a DeInit függvénnyel végződik, ahol a létrehozott objektumok felszabadítását végzem el, és az esetlegesen előfordult hibát iratom ki.

### **3.5.5. A kereskedési stratégia tesztelése**

A kereskedési stratégia tesztelését a Metatrader platformon belüli Strategy Tester modullal végeztem. Ez lehetőséget ad arra, hogy történelmi adatokon teszteljük egy stratégiát. Egy grafikonon szimulálja a program az idő múlását és a program futását, és a nyitott pozíciókat és azok fejleményét kirajzolja a grafikonra.

Egy listát is kapunk a tesztelés végén az összes nyitott pozícióról és eredményeiről. Ezen kívül egy jelentést is készít, amiben összefoglalja a teljes nyereséget, veszteséget, pozíciók

számát nyertes/vesztes és hosszú/rövid típusok szerint, és az átlag és maximális veszteséget és nyereséget is láthatjuk.

A tesztelést több indexre elvégeztem, mindegyiket az összes elérhető adaton futtattam, napos grafikonon. Az eredményeket a következő fejezetben ismertetem.

### **3.6. Felmerült nehézségek és megoldásaik**

A felmerült nehézségek nem feltétlenül technikai dolgok voltak. Arra az elvi kérdésre kerestem a választ, hogy mi az az elemi kereskedési stratégia, ami legkevésbé befolyásolja a hullámok állapotán alapuló döntést.

#### **3.6.1. Az egymás után következő szakaszok irányának problémája**

A szakaszokra bontás elvégzése során több minimális hosszúságot próbáltam meghatározni, de mivel az adatok gyertya formában vannak, egy diszkrét ponthoz tartozó adat tulajdonképpen egy intervallumot ölel át, ezért ha nagynak választottam volna a minimális adatok számát, amiből állhat egy szakasz, akkor az olyan mozgások, amik nagyon rövid időn belül történnek, de nagy árintervallumot ölelnek át, elvesztődtek volna. Ezért a nagyobb időbeosztásokon 3-nak vettem a szakaszhoz szükséges adatok minimális számát.

Ezután a következő felmerülő probléma az volt, hogy a felismert szakaszok, amik nagyjából csak a minimális számú hosszúak voltak, úgy követték egymást, hogy az irányuk megegyező volt. Ezeket szükséges volt összeolvasztani. Első lépésben azzal próbálkoztam, hogy divide et impera módszerrel próbáljam meg felosztani a szakaszokat, mindegyik szakaszt két különböző irányú szakaszra. Belátható, hogy már a második lépésnél, mikor két szakaszt kellett négyre bontsák, annak ellenére, hogy a két szakasz két olyan másik szakaszra bomlott, amelyeknek az iránya ellentétes volt, az összeillesztési pontnál nem biztos, hogy kisebb szakaszok irányai váltották egymást.

A megoldás az lett, hogy a felismerést követően végigmegyek a szakaszokon és azokat a szakaszokat, amik egymás utániak és az irányuk is megegyezik, olyan módon olvasztom egybe, hogy az új szakasz minkét eredeti szakasz adataira legyen a legjobban fekvő szakasz. Ez megoldotta azt a problémát, hogy a szakaszok iránya nem változott, mivel az egymásutáni azonos irányú szakaszok egy nagyobb szakasszá alakultak át.

A 3.7-es ábrán az összeolvasztás nélkül látható a szakaszok felosztása, a 3.8-as ábrán pedig az összeolvasztás elvégzése után.



3.7. ábra

Megfigyelhető a 2008 szeptemberétől a 2011 áprilisáig terjedő perióduson, hogy a 3.7-es ábrán rengetek hasonló irányú szakasz van. De ezt havi viszonylatban nyugodtan vehetjük egy növekvő szakasznak. A 3.8-as ábrán a fent említett periódushoz egyetlen szakasz tartozik. A kisebb változások majd a kisebb időbeosztáson fognak érvényesülni.

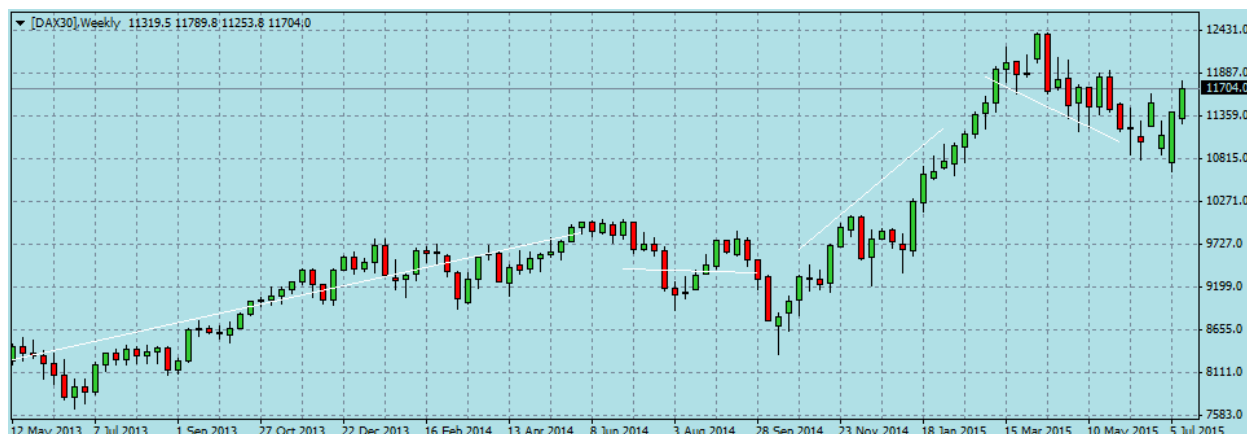


3.8. ábra

### 3.6.2. Nagyobb időbeosztású hullámok megjelenítése kisebb időbeosztáson

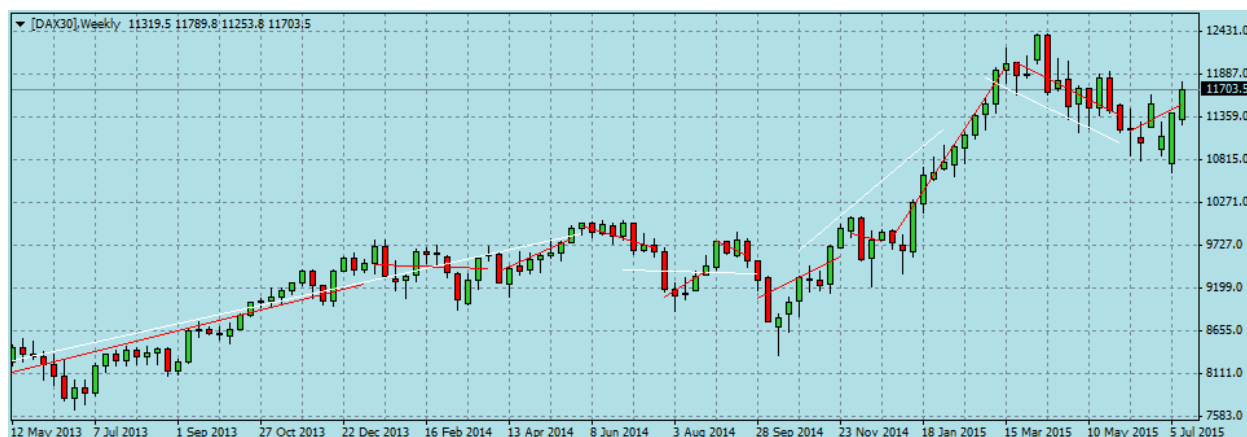
A második probléma, amibe beleütköztem az az volt, hogy amikor a nagy időbeosztású szakaszokkal akartam dolgozni kisebb időbeosztáson, akkor a szakaszok végénél kiesést észleltem. Egy példával szemléltetve fejtem ki a problémát. A havi grafikonon felismert két egymásutáni szakasz esetében az első szakasz például június elsejéig tart, a következő július

elsejétől kezdődik. Ez a havi grafikonon nem okoz kiesést, mivel június elseje és július elseje egymást követő gyertyákon helyezkednek el. De ha ugyanezeket a szakaszokat jelenítem meg a hetes grafikonon, akkor megfigyelhető, hogy a június elseje és július elseje közé bekerül három heti gyertya, amihez nem tartozik a szakasznak megfelelő érték. Ez a június 8.-tól kezdődően a július 1.-ig heteknek megfelelő három hét. A 3.9-es ábrán megfigyelhető a kiesés:



3.9. ábra

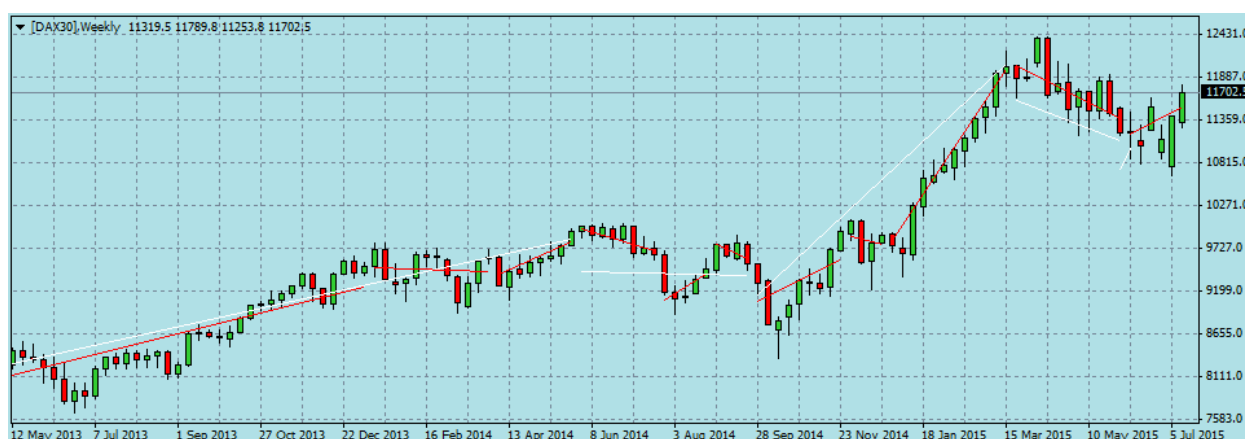
A 3.10-es ábrán a heti szakaszok is láthatóak:



3.10. ábra

Ezt a problémát úgy sikerült megoldani, hogy a nagyobb időbeosztású szakaszokat az egyel kisebb időbeosztású szakaszok végéhez igazítom. Mivel tudom, hogy azok részletesebb adatból állnak össze, és végpontjai pontosabbak, ezért a havi adathoz tartozó szakaszokat a hetes adatúak végpontjaihoz illeszttem a következő logika szerint: végigmegyek a nagyobb

időbeosztású szakaszokon. Ha a szakasz kezdőpontja nem egyezik meg két kisebb időbeosztású szakasz találkozási pontjával, akkor a nagyobb időbeosztású szakasz kezdőpontját az irányától függően vagy visszafele húzom vagy előre tolom, meghosszabbítva vagy lerövidítve magát a nagyobb időbeosztású szakaszt. Ugyanezt elvégzem a szakasz végpontjára is. A rövidítést vagy hosszabbítást az irány alapján döntöm el, oly módon, hogy a kezdőpont a legközelebbi olyan kisebb időbeosztású szakasz kezdőpontjával essen egybe, amelynek iránya megegyezik a nagyobb időbeosztásúval. A 3.11-es ábrán láthatjuk az eredményt:



3.11. ábra

### 3.6.3. A kereskedési stratégia megválasztása

A kereskedési stratégiát olyan módon törekedtem megválasztani, hogy minél kevesebb befolyása legyen az eredményre. A stratégia kiegészülhetne követő stoppal, ami a nyereséges pozíciókat bebiztosítaná. Vagy azt is lehetne figyelni a kereskedés során, hogy ha egy pozíció rég nyitva van, de se nem vesztes, se nem nyertes, akkor zárjam le [2] (B. Vanstone, T. Hahn). Ezeket a plusz szempontokat akarattal hagytam ki a kereskedés folyamatából, hogy azon maradjon a hangsúly, hogy a hullámok felismerése mennyire mutat biztos irányt.

A 3.12-es ábrán megfigyelhető az ideális belépési helyzet, mivel a grafikon elején látható, hogy a 3-as hullám véget ért a havi (fehér) beosztásban és a heti (piros) beosztásban is. A grafikonon megjelenik a napos beosztás is (késsel), de ezt jelenleg nem használom, viszont szemléltetésként szolgál, hogy mind a három időbeosztásban a 4-es hullámban vagyunk. A piros 4-es hullám pedig 3 részre oszlik, tehát valóban egy javító hullámról van szó. A belépési pontot a

fekete nyíl jelöli, a piros egyenes a maximális veszteség szintjét jelöli, a kék egyenes pedig azt a



nyereség szintet, amit elvárunk ettől a pozíciótól.

### 3.12. ábra

A fenti ábra alapján a kereskedés során a rendszer több zöld gyertya esetén fog pozíciót nyitni, mindig az azelőtti gyertya aljához igazítva a maximális veszteség szintjét.

## 4. fejezet

### Eredmények és következtetések

#### 4.1. Eredmények

A stratégia tesztelését négy indexre végeztem el, ezek közül háromnak nagyobb a forgalma. A teszteléshez egy 10.000 dolláros demó számlát használtam, tehát ez volt a kiinduló összeg a számlán. A teszteléshez az összes elérhető adatot felhasználtam. Ezek a következő időintervallumot ölelték át:

| Index   | Név                          | Kezdeti dátum | Végső dátum |
|---------|------------------------------|---------------|-------------|
| NQ100   | Nasdaq 100                   | 2005.10.21    | 2015.07.10  |
| SP500   | Standard and Poor's 500      | 2005.10.19    | 2015.07.10  |
| DJI30   | Dow Jones Industrial Average | 2008.11.25    | 2015.07.10  |
| STOXX50 | Euro Stoxx 50                | 2007.11.08    | 2015.07.13  |

#### 4.1. táblázat

Az NQ100 esetében teszteltem a stratégia mindkét variánsát. Az elsőnél nem vettem figyelembe a havi irányt, csupán a heti hullámok alapján nyitottam pozíciót. A tesztelés eredményéből kitűnik, hogy ez kockázatosabb variáns. Habár a nyereség összességben nagyobb volt ebben az esetben, a veszteségek mértéke is nagyobb volt. A második variáns biztonságosabbnak bizonyult, és a nyeresége sem elhanyagolható. A 4.2-es táblázatban az NQ100 indexen futtatott stratégia eredménye látható.

Megfigyelhető, hogy az időbeosztásokon felismert hullámok monotonitása közötti összefüggés vizsgálatával sikerült biztonságosabbá tenni a stratégiát. A veszteséget majdnem felére sikerült csökkenteni, míg a nyereség arányosan a háromnegyedére csökkent. A nyitott pozíciók száma is csökkent, hiszen a különböző időbeosztásokon felismert hullámok monotonitása közötti összefüggés vizsgálatával kiszűrtünk egyes pozíciókat. Így egyetlen rövid pozíció sem került megnyitásra, mivel az index hosszú távú trendje felfele irányuló volt.

Még egy fontos szempont, hogy sikerült növelni a nyereségek pozíciószámok arányát, habár ez az átlag vesztes pozíció méretének a növekedését vonta maga után.



|   | Monotonitások közötti<br>összefüggés vizsgálata nélkül | Monotonitások közötti<br>összefüggés vizsgálatával |
|---|--|--|
| <b>Teljes nyereség (\$)</b>                 | 61,646.50  | 46,291.46  |
| <b>Teljes veszteség (\$)</b>                | 12,428.73  | 6,842.00   |
| <b>Nyitott hosszú pozíciók száma</b>        | 54   | 39   |
| <b>Nyitott rövid pozíciók száma</b>         | 15   | 0  |
| <b>Nyertes pozíciók aránya (%)</b>          | 39.13  | 51.28  |
| <b>Vesztes pozíciók aránya (%)</b>          | 60.76  | 48.72  |
| <b>Legnagyobb nyereség (\$)</b>             | 5,063.77   | 5,063.77   |
| <b>Legnagyobb veszteség (\$)</b>            | 987.47   | 987.47   |
| <b>Átlag nyereség (\$)</b>                  | 2,283.20   | 2,314.57   |
| <b>Átlag veszteség (\$)</b>                 | 295.92   | 360.11   |
| <b>Eredmény (nyereség - veszteség) (\$)</b> | <b>49,217.77</b>                                       | <b>39,449.46</b>                                   |
| <b>Eredmény (nyereség/alaptőke) (%)</b>     | <b>492.18</b>  | <b>394.49</b>                                      |

4.2. táblázat

Az eredmények kiértékeléséhez a banki kamatot veszem viszonyítási alapként, hiszen az lehet egy biztonságosabb alternatíva a tőzsdei befektetésre. A monotonitások közötti összehangolás vizsgálatával elért nyereséget banki letét esetén, évi kamatos kamattal számolva

| Év | Tőke (\$) | Éves kamat (%) |
|----|-----------|----------------|
| 0  | 10,000.00 | 14.71%         |
| 1  | 11,471.00 | 14.71%         |
| 2  | 13,158.38 | 14.71%         |
| 3  | 15,093.98 | 14.71%         |
| 4  | 17,314.31 | 14.71%         |
| 5  | 19,861.24 | 14.71%         |
| 6  | 22,782.83 | 14.71%         |
| 7  | 26,134.18 | 14.71%         |
| 8  | 29,978.52 | 14.71%         |
| 9  | 34,388.36 | 14.71%         |
| 10 | 39,446.89 | 14.71%         |

4.3. táblázat

az éves banki kamat mértéke a 4.3-as táblázat alapján 14.71% kellene, hogy legyen ahhoz, hogy ugyanezt az eredményt hozza 10 év alatt. Jelenleg a banki kamatok messze elmaradnak ettől az értéktől (kb. 2% lejre és 0.5% euró esetén).

Az indexek eredményeit a 4.4-as táblázatban tekinthetjük át. Az eredmények a monotonitás megtartás vizsgálatával készültek.

|   | DJI30            | NQ100            | SP500            | EURO STOXX50  |
|---|------------------|------------------|------------------|---------------|
| <b>Hibás adatok száma</b>                   | 542.00           | 7,973.00         | 153.00           | 293,571.00    |
| <b>Teljes nyereség (\$)</b>                 | 20,345.36        | 46,291.46        | 45,937.17        | 2,004.82      |
| <b>Teljes veszteség (\$)</b>                | 2,530.16         | 6,842.00         | 6,768.97         | 1,378.84      |
| <b>Nyitott hosszú pozíciók száma</b>        | 29               | 39               | 25               | 35            |
| <b>Nyitott rövid pozíciók száma</b>         | 0                | 0                | 1                | 0             |
| <b>Nyertes pozíciók aránya (%)</b>          | 55.17            | 51.28            | 50.00            | 25.71         |
| <b>Vesztes pozíciók aránya (%)</b>          | 44.83            | 48.72            | 50.00            | 74.29         |
| <b>Legnagyobb nyereség (\$)</b>             | 3,380.17         | 5,063.77         | 14,823.43        | 375.70        |
| <b>Legnagyobb veszteség (\$)</b>            | 548.00           | 987.47           | 1,906.41         | 123.84        |
| <b>Átlag nyereség (\$)</b>                  | 1,271.59         | 2,314.57         | 3,533.63         | 222.76        |
| <b>Átlag veszteség (\$)</b>                 | 194.63           | 360.11           | 520.69           | 53.03         |
| <b>Eredmény (nyereség - veszteség) (\$)</b> | <b>17,815.20</b> | <b>39,449.46</b> | <b>39,168.20</b> | <b>625.98</b> |
| <b>Eredmény (nyereség/ alaptőke) (%)</b>    | <b>178.15</b>    | <b>394.49</b>    | <b>391.68</b>    | <b>6.26</b>   |

4.4. táblázat

Látható, hogy mind a négy vizsgált index esetében pozitív az eredmény, a negyedik indexnél figyelhető meg kisebb nyereség, mivel ebben az esetben a tesztelés folyamatában az adatok

nem voltak helyesek, ezt a stratégia tesztelő modul jelentése mutatja ki a tesztelés végén. A többi indexhez viszonyítva a STOXX 50 index esetében a hiba nagyságrendekkel nagyobb, ezért ezt az esetet nem tekinthetjük jellemzőnek. A kereskedés során a különböző indexek esetében a számlán levő tőke és az egyenleg változása a függelékben található meg.

## 4.2. Következtetések

A tesztelés eredménye sikeresnek mondható, mivel az Elliott-hullámok felismerésén alapuló kereskedési stratégia sokkal nagyobb nyereséget hozott az indexek esetében, mint a viszonyítási alapul szolgáló banki letét lehetősége.

Kimondható, hogy a hullámfelismerésen alapuló döntéshozás egy kereskedési rendszer kialakításában jó választás. Összehasonlítva a létező Elliott-hullám alapján végzett kereskedési rendszerekkel, kimagasló nyereségeknek köszönhetően (**NQ100 index esetében 390%-os 10 év alatt**), sokkal jobban teljesített ez a rendszer, mint a többi.

## 4.3. Továbbfejlesztési lehetőségek

Rengeteg irányban el lehet indulni, akár a hullámfelismerés, akár a kereskedési stratégia finomítása útján. Az első lépés az lehetne, hogy ne csak a belépéshez megfelelő hullámokat határozzuk meg, hanem minden egyes hullámot. Ez abban segítene, hogy a belépési hullámokat finomítani tudjuk. Így különbséget lehetne tenni a visszahúzódó A-B-C alakzat előrehaladó hullámai és az 1-2-3-4-5 előrehaladó hullámai között.

Egy második továbbfejlesztési irány a Fibonacci-számok elemzése és alkalmazása lehetne. A maximális veszteség, elérhető nyereség szintjeihez is fel tudnánk használni az arányokat, ismerve a különböző hullámok terjedelmét és a közöttük fennállható arányokat.

A harmadik lehetőség az lehetne, hogy időben is kövessük a hullámok alakulását, és a lezárási feltétel meghatározásához használjuk fel ezt az információt. Ha például egy pozíció már régóta nyitott, de sem a veszteséget, sem a nyereséget nem érte el, akkor úgy dönthetünk, hogy lezárjuk. A kialakult hullám állapota szerint valószínűsíthetjük, hogy a hullám mennyire lehet hosszú, és az ellenkező irányú hullám kialakulása előtt lezárhatjuk a pozíciót.

A negyedik ötlet az lehet, hogy egy pozíció fejleményét egy kisebb időbeosztáson kövessük, az ott kialakuló hullámok segítséget nyújthatnak abban, hogy arról kapjunk képet, hogy az általunk “meglovagolt” hullám a kialakulásának melyik fázisában van.

Ezekon kívül a használt paramétereket is lehet finomítani, a használt indexre hangolni. Ha pedig a mostani állapotában akarjuk használhatóbbá tenni a kereskedési rendszert, akkor olyan irányba indulhatunk el, hogy az összes elérhető indexen végigfusson és belépési lehetőséget keressen. Mivel a tesztek több évnnyi adaton futottak végig, ezzel a megoldással megtartható a hosszú időtávú elemzés, de nem kell éveket sem várni ahhoz, hogy eredményt lássunk.

## **Ábrák jegyzéke**

- 2.1. ábra - Alap hullám típusok
- 2.2. ábra - Jelölési konvenció
- 2.3. ábra - Kiterjesztett hullámok
- 2.4. ábra - Megcsonkított hullámok
- 2.5. ábra - Elliott-hullámok és a Fibonacci számok
- 3.1. ábra - Szakaszokra bontás havi adaton
- 3.2. ábra - Szakaszokra bontás havi és heti adaton
- 3.3. ábra - 1-2-3 alakzatok havi adaton
- 3.4. ábra - Felismert 1-2-3-4-5 sorozat heti adaton
- 3.5. ábra - Hullámok egybeesése különböző időbeosztásokon
- 3.6. ábra - A program állapotdiagramja
- 3.7. ábra - Szakaszok felosztása összeolvasztás nélkül
- 3.8. ábra - Szakaszok felosztása összeolvasztással
- 3.9. ábra - Csonkolt havi szakaszok heti grafikonon
- 3.10. ábra - Csonkolt havi szakaszok heti grafikonon a heti szakaszokkal együtt megjelenítve
- 3.11. ábra - Szakaszok meghosszabbítása
- 3.12. ábra - Belépési pont meghatározása

## **Táblázatok jegyzéke**

3.1. táblázat - Szakaszok felosztásához különböző időbeosztásnak megfelelő minimális hosszúság

4.1. táblázat - A tesztelés során használt idősor időintervalluma

4.2. táblázat - Az NQ100 indexen tesztelt kétféle stratégia eredménye

4.3. táblázat - Banki kamat kiszámítása

4.4. táblázat - Indexekre végzett tesztek eredményei

## Irodalomjegyzék

- [1] Robert Rougelot Prechter, Jr., A. J. Frost: "Elliott Wave Principle", New Classics Library, (2005).
- [2] Bruce Vanstone, Tobias Hahn: "Designing Stock Market Trading Systems", Harriman House, (2010).
- [3] Michelberger Pál, Szeidl László, Várlaki Péter: "Alkalmazott folyamatstatisztika és idősor analízis", Typotex Kiadó, (2001).
- [4] Monica Tirea, Ioan Tandau, Viorel Negru: "Stock market Multi-Agent Recommendation System Based on the Elliott Wave Principle", (2012)  
URL: [http://link.springer.com/chapter/10.1007/978-3-642-32498-7\\_25](http://link.springer.com/chapter/10.1007/978-3-642-32498-7_25),  
Utoljára látogatva: 2015.07.04.
- [5] George S. Atsalakis, Emmanouil M. Dimitrakakis, Constantinos D. Zopounidis: "Elliott Wave Theory and neuro-fuzzy systems, in stock market prediction: The WASP system", (2011), Expert Systems with Applications, Volume 38/8, 9196-9206.  
URL: <http://www.sciencedirect.com/science/article/pii/S0957417411000881>  
Utoljára látogatva: 2015.07.05.
- [6] Roman Martiruk: "The Implementation of Automatic Analysis of the Elliott Waves in MQL5", (2011)  
URL: <https://www.mql5.com/en/articles/260>  
Utoljára látogatva: 2015.07.10.
- [7] MQL5 Limited: MQL4 Documentation - Client Terminal Events  
URL: [http://docs.mql4.com/runtime/event\\_fire](http://docs.mql4.com/runtime/event_fire)  
Utoljára látogatva: 2015.07.15.
- [8] MQL5 Limited: MQL4 Documentation - Event Handling Functions  
URL: <http://docs.mql4.com/basis/function/events>  
Utoljára látogatva: 2015.07.15.
- [9] Masur - Ralph Nelson Elliott: "Elliott wave", "The Basis of the Wave Principle"  
URL: [https://commons.wikimedia.org/wiki/File:Elliott\\_wave.svg#/media/File:Elliott\\_wave.svg](https://commons.wikimedia.org/wiki/File:Elliott_wave.svg#/media/File:Elliott_wave.svg)  
Utoljára látogatva: 2015.07.17.

[10] Algorithm for Finding the Best Fit Line,

URL: <http://faculty.cs.niu.edu/~hutchins/csci230/best-fit.htm>

Utoljára látogatva: 2015.07.17.

[11] MQL5 Limited: MQL4 Reference

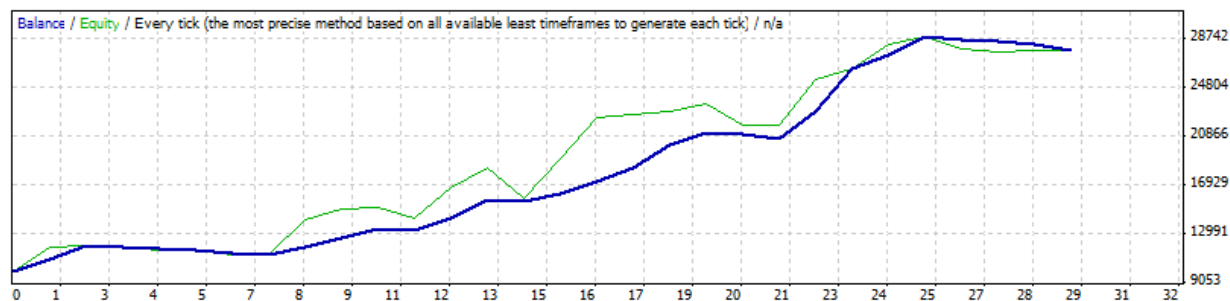
URL: <http://docs.mql4.com>

Utoljára látogatva: 2015.07.17.

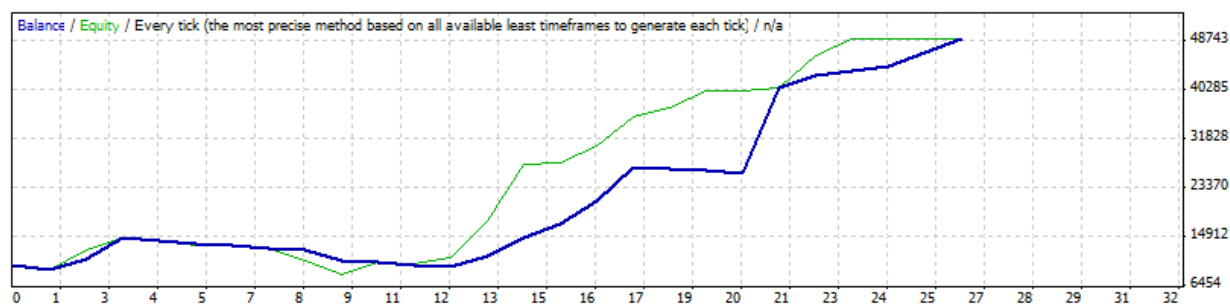


## Függelékek

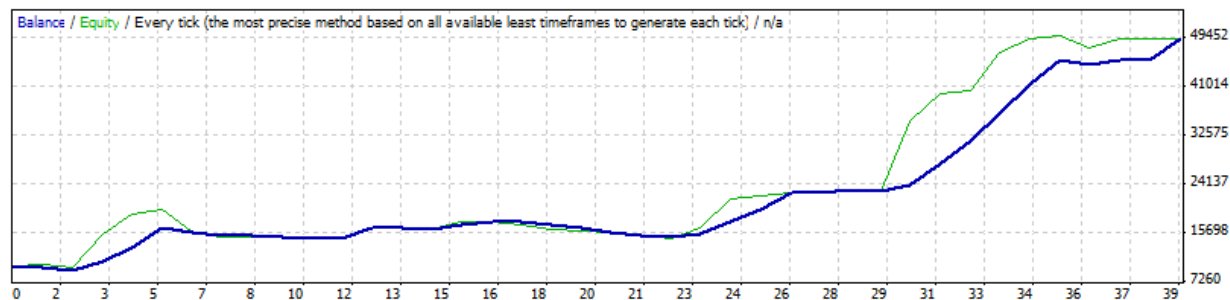
1. A Dow Jones Index teszt során az egyenleg<sup>7</sup> és a tőke<sup>8</sup> változása:



2. A Standard and Poor's Index teszt során az egyenleg és a tőke változása:



3. A Nasdaq 100 index teszt során az egyenleg és a tőke változása:



<sup>7</sup> angolul: balance

<sup>8</sup> angolul: equity

4. A STOXX50 index teszt során az egyenleg és a tőke változása:

