



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

TVORBA OBCHODNÍ STRATEGIE PRO MĚNOVÝ TRH

DEVELOPING OF TRADING STRATEGY FOR CURRENCY MARKET

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Václav Sauer

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Budík, Ph.D.

BRNO 2017

Zadání diplomové práce

Ústav:	Ústav informatiky
Student:	Bc. Václav Sauer
Studijní program:	Systemové inženýrství a informatika
Studijní obor:	Informační management
Vedoucí práce:	Ing. Jan Budík, Ph.D.
Akademický rok:	2016/17

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Tvorba obchodní strategie pro měnový trh

Charakteristika problematiky úkolu:

Úvod
Cíle práce, metody a postupy
Teoretická východiska práce
Analýza problému
Vlastní návrhy řešení
Závěr
Seznam použité literatury

Cíle, kterých má být dosaženo:

Cílem diplomové práce je navrhnout, implementovat a optimalizovat automatický obchodní systém pro pro obchodování na měnovém trhu s důrazem na minimalizaci rizika, které je s obchodováním spojeno pomocí důkladného risk a money managementu a to tak, aby výsledný návrh byl schopen dosahovat dlouhodobě udržitelného kladného zhodnocení investovaných prostředků.

Základní literární prameny:

DOSTÁL, P. Pokročilé metody analýz a modelování v podnikatelství a veřejné správě. 1. vyd. Brno: CERM, 2008. 432 s. ISBN 978-80-7204-605-8.

GOLDBERG, D. Genetic Algorithms in Search, Optimization, and Machine Learning. 1. vyd. USA: Addison-Wesley, 1989. 412 p. ISBN 978-0-201157-67-3.

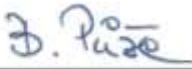
GRAHAM, B. Inteligentní investor. 1. vyd. Praha: GRADA, 2007. 504 s. ISBN 978-80-247-1792-0.

REJNUŠ, O. Finanční trhy. 1. vyd. Ostrava: KEY Publishing, 2008. 548 p. ISBN 978-80-87-8.



WILLIAMS, L. How I Made One Million Dollars Last Year Trading Commodities. 1. vyd. USA: Windsor Books, 1979. 130 p. ISBN 978-09-30233-10-5.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2016/17.

V Brně, dne 28. 2. 2017



doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Diplomová práce se věnuje návrhu, implementaci a optimalizaci automatického obchodního systému pro měnový trh. V práci jsou rozebrány teoretické předpoklady pro tvorbu systému, a to od představení měnového trhu přes typy analýzy trhu, money management a risk management až po technické indikátory. Dále práce popisuje, co vše je k vývoji takového systému potřeba a jaké nutné části musí systém obsahovat. Nakonec práce rozebírá, jak je možné vytvořený systém s využitím historických obchodních dat otestovat a zoptimalizovat a jaké jsou jeho přínosy.

Abstract

This thesis deals with the design, implementation and optimization of the automated trading system for the foreign exchange market. Thesis analyses theoretical aspects for the system implementation, including introduction of foreign exchange market, types of market analysis, money management, risk management and technical indicators. The thesis further describes, what is required for development of such system and what important parts the system must contain. The work also describes how the system can be tested and optimised based on historical data.

Klíčová slova

obchodní systém, měnový trh, Forex, MetaTrader, optimalizace, automatické obchodování, investice

Key words

trading system, currency market, Forex, MetaTrader, optimization, automatic trading, investment

Bibliografická citace

SAUER, V. *Tvorba obchodní strategie pro měnový trh*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2017. 67 s. Vedoucí diplomové práce Ing. Jan Budík, Ph.D.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 26. května 2017

.....
Bc. Václav Sauer

Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Ing. Janu Budíkovi, Ph.D. za inspiraci a za pomoc při tvorbě práce.

Obsah

1. Úvod	10
2. Cíle práce, metody, postupy	11
3. Teoretická východiska práce	12
3.1. Kapitálové trhy	12
3.1.1. Účastníci trhu	12
3.1.2. Komoditní trh	12
3.1.3. Akciový trh	13
3.1.4. Měnový trh	13
3.2. Cena instrumentu	15
3.2.1. Pip, lot	15
3.2.2. Finanční páka	15
3.2.3. Vícedenní obchodování	16
3.3. Risk a money management	16
3.3.1. Skladba portfolia	17
3.3.2. Volba velikostí pozic	17
3.3.3. Výstup z obchodu	17
3.3.4. Trend a velikost finanční páky	18
3.4. Způsoby analýzy trhu	18
3.4.1. Fundamentální analýza	18
3.4.2. Technická analýza	19
3.5. Cenový graf	20
3.5.1. Line chart	20
3.5.2. Bar chart	21
3.5.3. Candlestick chart	22
3.5.4. Další typy grafů	23
3.6. Metody technické analýzy	23
3.6.1. Trendové indikátory	23
3.6.2. Oscilátory	29
3.7. Automatický obchodní systém	31
3.8. Platformy pro vývoj AOS	32
3.8.1. MetaTrader	33
3.8.2. cTrader	34
3.8.3. NinjaTrader	34
4. Analýza problému	36
4.1. Úrovně automatizace obchodování	36
4.2. Získání automatického obchodního systému	36
4.3. Vývoj automatického obchodního systému	38
4.3.1. Požadavky kladené na AOS	38
4.3.2. Programové a hardwarové vybavení pro tvorbu AOS	39
4.4. Testování a optimalizace automatického obchodního systému	40
4.4.1. Testování AOS	40

4.4.2. . Optimalizace AOS	41
4.5. Běh automatického obchodního systému	43
4.5.1. Běh systému u obchodníka	43
4.5.2. Běh systému ve výpočetním centru	44
4.5.3. Běh systému u brokera	44
5. Vlastní návrh řešení	45
5.1. Návrh vlastní strategie	45
5.1.1. Výběr indikátorů	45
5.1.2. Výběr měnového páru a časového rámce	46
5.1.3. Pravidla pro otevření obchodu	46
5.2. Implementace vybrané strategie	47
5.2.1. Inicializace obchodního systému	49
5.2.2. Hlavní smyčka programu.	49
5.2.3. Doplnkové funkce obchodního systému	50
5.3. Testování a optimalizace strategie	50
5.3.1. In sample testování	51
5.4. Optimalizace strategie	51
5.4.1. Optimalizace parametrů indikátoru Parabolic SAR	51
5.4.2. Optimalizace parametrů klouzavých průměrů	52
5.4.3. Optimalizace všech parametrů najednou	53
5.4.4. Out of sample test	55
5.4.5. Optimalizace dalších konfigurovatelných nastavení	55
5.5. Přínos navrženého systému	56
6. Závěr	57
7. Seznam použitých zdrojů	58
8. Seznam obrázků	60
9. Seznam tabulek	61
10. Příloha 1: Zdrojový kód automatického obchodního systému	62

1. Úvod

Většina lidí, kteří vlastní nějaké finanční prostředky, má snahu je nějak dobře zhodnotit. Mnohdy to však činí takovým způsobem, že jim výnosy nepokryjí ani pokles hodnoty peněz způsobený inflací. Existují však i jiné způsoby, jak peníze zhodnotit, než je spořicí účet s ročním výnosem kolem 1 %.

Jednou takovou alternativou je obchodování na forexovém trhu. To s sebou sice nese značná rizika a klade velké nároky na znalosti a vědomosti investora, odměnou mu však můžou být mnohem vyšší výnosy. Výhodou forexového trhu je to, že správnou volbou strategie si každý investor může navolit přesně takový poměr zhodnocení a rizika, jaký mu vyhovuje. Zrovna tak se investor může rozhodnout, zda u obchodování stráví několik hodin denně, a to se tak stane jeho obživou nebo zda pro něj bude obchodování pouze vedlejší činnost, které věnuje sotva pár minut denně.

Obchodování na forexu se díky digitalizaci, rozšíření osobních počítačů a širokopásmovému internetu stalo snadno dostupným i pro širokou masu lidí a na trhu je obrovské množství brokerů, kteří jim nabízejí své služby.

V této práci bych se rád zaměřil na vytvoření automatického obchodního systému, který hlídá forexový trh místo investora a také za něj obchoduje. Vytvořit takový systém je náročné především kvůli nejednoznačnosti finančních trhů, jejichž vývoj se téměř nedá předpovědět. Takový systém proto musí být dobře otestován a musí se u něj dbát na money management a risk management. Bez těchto věcí by systém s velkou pravděpodobností proobchodoval investorův účet do červených čísel.

2. Cíle práce, metody, postupy

Cílem této práce je navrhnout automatický obchodní systém, který by zvládnul automaticky obchodovat na měnovém trhu a tím zhodnotit vložené prostředky.

V teoretické části práce jsou rozebrány komoditní, měnové a akciové trhy a jejich účastníci. Dále jsou v této části rozebírány základní odborné pojmy s obchodováním na Forexu spojené, jejichž znalost je pro pochopení problematiky nutností. Jsou zde popsány přístupy k analýze ceny instrumentu a metody technické analýzy využitelné v automatických obchodních systémech.

Analýza problému vývoje AOS je rozebrána v kapitole 4. Zaměřuji se zde zejména na to, jak může obchodník získat automatický obchodní systém, jaké jsou požadavky pro naprogramování takového systému, co vše obnáší jeho testování a optimalizace a co je potřeba pro to, aby mohl být systém spuštěn a reálně používán.

Vlastní návrh automatického obchodního systému je rozpracován v kapitole 5. Je zde popsáno, jaké jsou v systému použity indikátory technické analýzy a proč byly vybrány právě ty. Také je zde objasněn výběr časového rámce a stanovení pravidel, podle kterých systém obchoduje. V této je také kapitole popsán samotný algoritmus obchodního systému a to, jak byl systém otestován a optimalizován.

V závěru je obchodní strategie zhodnocena, popsán její přínos a případné využití v praxi.

3. Teoretická východiska práce

3.1. Kapitálové trhy

Kapitálové trhy umožňují jejím účastníkům směnu zboží za kapitál, tedy za peníze. Vše na těchto trzích má svou přesně stanovenou cenu. Hlavním cílem většiny subjektů, které na těchto trzích spekulují, je navýšit svůj vlastní kapitál, často na úkor ostatních účastníků.

3.1.1. Účastníci trhu

Účastníky na trzích se dají rozdělit na dvě skupiny, na velké a na malé. Velcí hráči jako jsou nadnárodní investiční společnosti (Goldman-Sachs, Deutsche Bank, JP Morgan atd.) nebo centrální banky jednotlivých států, mají dostatečnou sílu k tomu, aby ovlivnili trh. Objemy jejich obchodů dosahují stovek milionu amerických dolarů, mohou probíhat v řádu několika dnů i týdnů a tím mohou na trhu tvořit trend či ho alespoň výrazně ovlivňovat. Odhaduje se, že tyto velké subjekty tvoří 80-90 % objemu obchodu na trzích (1 str. 19).

Opakem velkých hráčů jsou malé subjekty, jednotliví drobní obchodníci a malé společnosti. Tato skupina účastníků nemá dostatečnou sílu na ovlivnění trhu. Jejich snahou je sledovat a analyzovat trh a chovat se podle něj.

3.1.2. Komoditní trh

Komoditní trhy vznikly a slouží k tomu, aby na nich obchodníci mohli směňovat zboží. K tomu účelu se dají komodity rozdělit do několika tříd. Toto dělení není pevně dané a v různých zdrojích lze najít od několika základních kategorií až po několik desítek kategorií. Základní dělení může vypadat následovně:

- Obiloviny – kukuřice, sójové boby, pšenice apod.
- Maso – hovězí, vepřové, telecí apod.
- Kovy – zlato, měď, olovo apod.
- Měkké – cukr, káva, bavlna apod.
- Energie – ropa, plyn, topný olej apod.

Kontrakty na komoditních trzích se nejčastěji vypisují na konkrétní data, po jejichž vypršení dochází k fyzickému dodání dané komodity. Prodejce se tedy zavazuje k tomu, že dodá danou komoditu za danou cenu a kupující k tomu, že ji zaplatí a převezme. Pokud

obchodník na tomto trhu spekuluje a nemá v plánu obchod skutečně provést a nakoupenou věc převzít, tak musí uzavřít ty kontrakty, kterým se blíží datum vypršení, a přesunout své pozice na kontrakty končící později v budoucnu, jedná se o tzv. rollover.

Druhou možností pro obchodování komodit je spotové obchodování, kde po objednání zboží je obchod ihned uzavřen a může se přejít k dodávce zboží.

3.1.3. Akciový trh

Na akciových trzích je možné kupovat a prodávat finanční deriváty. Mezi ty se dají zařadit akcie jednotlivých firem, dluhopisy a jejich deriváty. Jednotlivé společnosti zde právě pomocí akcií a dluhopisů získávají finanční prostředky od investorů, kteří za zapůjčené prostředky na oplátku očekávají výnos ve formě úroku či dividend. Mezi největší akciové burzy patří New York Stock Exchange a NASDAQ, obě sídlící v New Yorku, či například londýnská London Stock Exchange. V České Republice se nachází Burza cenných papírů v Praze. Jedná se však o menší burzu s malou likviditou.

U akcií navíc existuje tzv. akciové indexy, které pod sebou sdružují skupinu několika akcií, ze kterých se následně počítá hodnota výsledného indexu. Mezi nejznámější indexy patří S&P 500 a Dew Jones, sdružující 500, respektive 30 významných amerických společností a také například index FTSE 100, který se počítá z ceny akcií sta významných britských společností. Ukázka některých indexů s jejich obchodním symbolem a cenou je znázorněna na obrázku níže.

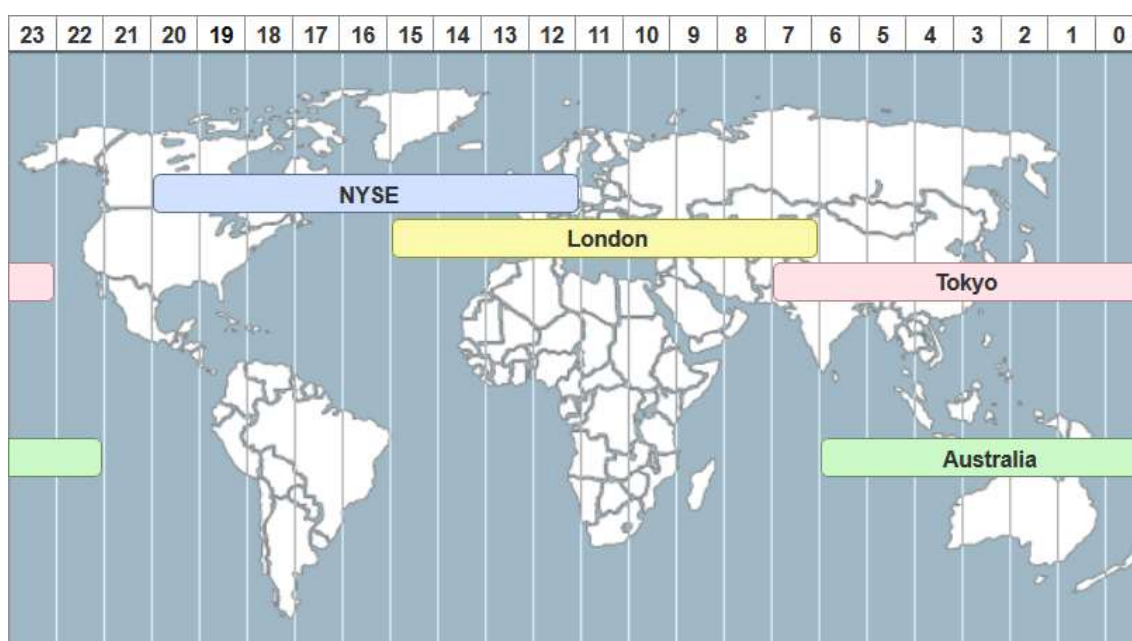
Symbol	Company	Last Price	Change	% Change	Day Chart
^GSPC	S&P 500	2,398.42	4.40	0.18%	
^DJI	Dow Jones Industrial Average	20,937.91	43.08	0.21%	
^IXIC	NASDAQ Composite	6,138.71	5.09	0.08%	
^VIX	Vix	10.46	-0.47	-4.30%	
^FTSE	FTSE 100	7,501.47	16.18	0.22%	

Obrázek 1: Cena některých významných indexů a její vývoj. Zdroj: <https://finance.yahoo.com/world-indices>

3.1.4. Měnový trh

Měnový trh neboli také Forex, od anglického Foreign Exchange, je devizový trh, na kterém se obchodují hlavní a vedlejší měnové páry. Jedná se o nejlikvidnější a největší

světový trh. Forex nemá žádnou centrální burzu. Obchoduje se ve hlavních světových obchodních centrech, a to především v Londýně, Frankfurtu, Paříži, New Yorku, Tokiu, Hong-Kongu a v Sydney. Díky tomu má tento trh tu unikátní vlastnost, že je otevřen 24 hodin denně. Naprostá většina obchodů se však uskutečňuje pouze v pracovní dny a přes víkend trh hibernuje. To je dáno tím, že velcí hráči, jako například německá Deutsche bank, o víkendu také neobchodují a většina brokerů o víkendech vůbec neumožňuje obchodovat a neumožňují na trh v této době zasílat obchodní příkazy. U brokerů, kteří to umožňují, to bývá za cenu velmi vysokého spreadu, a tedy i nákladů pro obchodníka. Časy otevření a zavření jednotlivých významných burz jsou znázorněny na obrázku níže.



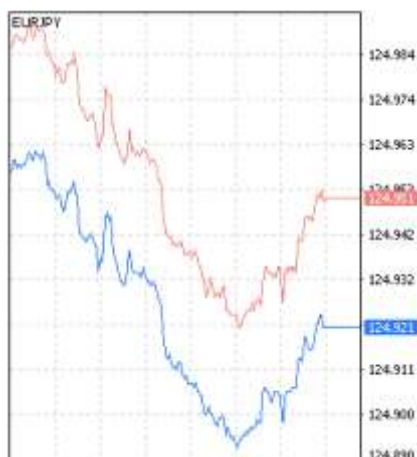
Obrázek 2: Forex obchodní hodiny. Zdroj: <https://www.investing.com/tools/market-hours>

Za výhody měnového trhu lze považovat jeho velmi vysokou likviditu, kdy zadané příkazy jsou vykonané v řádu maximálně několika sekund. Dále pak možnost dosáhnout zisku jak při nákupu, tak při prodeji instrumentů a možnost využít pákového efektu, kdy obchodníkovi k vytvoření kontraktu o určité velikosti stačí mít u brokera pouze složenou jistinu například o velikosti pouze 1/100 hodnoty kontraktu. Další velkou výhodou je dostupnost obchodování na tomto trhu, kdy mnoho brokerů nabízí bezplatné testovací demoverze účtů a zároveň je k tomuto trhu možné dohledat nepřehledné množství analýz, zpráv, grafů a dalších informací. Pak již je jen na zkušenostech obchodníka, jak toto vše zvládne využít.

3.2. Cena instrumentu

Instrument obchodovaný na burze má několik cen, kde každá určuje něco jiného. Nejdůležitější pro obchodníka je cena, za kterou obchodník prodává (také *bid* či *sell* – obchodník je v krátké pozici) a pak cena za kterou nakoupí (*ask*, *buy* – dlouhá pozice) (2). Cena prodeje je nižší než cena nákupu a rozdíl mezi těmito dvěma cenami se říká *spread*. Když tedy obchodník provede obchod, tak musí nejdříve překročit hranici spreadu a až teprve poté je jeho obchod ziskový. Tato hranice se nazývá *break even point*. To, jak je *spread* velký, záleží především na obchodovaném instrumentu a na brokerovi, který na *spreadu* vydělává. Hlavní měnové páry, které jsou obchodované ve velkém množství a které jsou více likvidní (např. EURUSD) mají *spread* nižší než méně likvidní vedlejší měnové páry.

Na obrázku níže je na měnovém páru EURJPY ukázka rozdílu mezi *sell* cenou (červeně) a *buy* cenou (modře) a *spreadu*, který je mezi nimi vytvořen.



Obrázek 3: Rozdíl mezi *sell* a *buy*. Zdroj: Program MetaTrader 4, vlastní tvorba

3.2.1. Pip, lot

Cena měnového páru na trhu se stanovuje systémem tzv. pipů. Jeden pip představuje nejmenší hodnotu, o který se může pohnout cena u měnového páru. Jedná se tedy o nejmenší desetinné číslo v ceně měnového páru.

Velikost kontraktu je poté dána množstvím zakoupených lotů. Jeden lot je nejčastěji 100 000 základních jednotek měny.

3.2.2. Finanční páka

Aby obchodník nemusel mít na svém účtu pro nákup jednoho lotu měnového páru vklad o celé jeho výši, tak umožňují brokeri využít systému finanční páky. Finanční páka

udává poměr prostředků, které musí obchodník mít na účtu vůči velikosti kontraktu, který může obchodovat. Brokeři většinou nabízí páky od 1:2, přes 1:20, 1:50, 1:100 až například po 1:1000. Obchodník poté například může na účtu, na kterém má páku 1:50, otevřít při vložených prostředcích 1000 USD obchod 50krát větší, tedy až o velikosti 50 000 USD. Není to však tak, že by obchodník poskytl na obchod 1000 USD z vlastních prostředků a 49 000 USD mu poskytl broker, ale broker poskytne obchodníkovi celou částku 50 000 USD a částka 1000 USD slouží brokerovi pouze jako záruka, že obchodník bude moci pokrýt případné vzniklé ztráty. Důležité je, že celá ztráta (ale i zisk) z obchodu náleží obchodníkovi a v případě, že by hrozila větší ztráta než obchodníkem vložené prostředky, tak broker buď obchod automaticky uzavře a tím ochrání obchodníka před vyšší ztrátou anebo vyzve obchodníka k vložení dalších prostředků na svůj účet. V takovém případě se jedná o tzv. margin call.

Páka tedy umožňuje obchodníkovi vytvářet mnohem větší obchody a tím dosáhnout násobně větších zisků, zároveň však má obchodník při prodělečných obchodech mnohonásobně větší ztráty a mnohem rychleji může přijít o prostředky na svém účtu. Z toho důvodu nejsou velmi vysoké páky příliš běžné.

3.2.3. Vícedenní obchodování

Pro vícedenní obchodování je velice důležitý rollover. Ten nastává v situaci, kdy obchodník drží obchod do následujícího dne. Přejít na následující den je na měnovém trhu pevně stanoven na 22:00 GMT. V tuto chvíli dochází k resetování všech pozic a započítání úrokových sazeb jednotlivých měn. Na tom může obchodník vydělat, ale i prodělat. Například při nákupu páru EUR/USD, obchodník platí za dolary, které si k uskutečnění obchodu zapůjčil a dostává úrok za eura, která po dobu otevření obchodu vlastní. Toho se dá využít i k samotnému obchodování, kdy obchodník nakoupí měnu s vyšší sazbou a vydělává na obdrženém úroku. Tato technika se nazývá Carry Trade. Riziko u ní spočívá v tom, že změna kurzu může vymazat zisk z připsaného úroku.

3.3. Risk a money management

Risk a money management jsou spolu s dobrou strategií jedny z nejdůležitějších věcí pro to, aby obchodník byl na trhu úspěšný. I přesto se však stále najdou obchodníci, kteří je berou na lehkou váhu, případně se jimi nezabývají vůbec. Na problematiku risk a money managementu má vliv skladba portfolia, volba velikosti pozic, řízení výstupu z obchodů a některé další věci.

3.3.1. Skladba portfolia

Dobrou skladbou portfolia může obchodník omezit riziko toho, že se trh bude pohybovat proti jeho systému. I přesto, že obchodník může mít sebelépe navržený a otestovaný systém, tak nikdy dopředu neví, jak se bude trh dále chovat. Tím pádem i dobře otestovaný systém, či systém, který v minulosti zaručeně fungoval, najednou fungovat nemusí. Pokud si však obchodník najde více trhů, které jeho systému vyhovují a na které si rozloží své obchody, tak pokud některý z nich přestane být profitabilní, ztráty z něj vykryjí zisky na ostatních trzích.

3.3.2. Volba velikostí pozic

Určení správné velikosti pozice je velmi důležité. Pokud budou otevírané pozice příliš vysoké, tak sice obchodník má možnost dosáhnout vyšších zisků, ale také ho i krátká série neúspěšných obchodů může velmi rychle připravit o prostředky na účtu. Pokud jsou zase pozice příliš malé, tak se obchodník zbytečně připravuje o potenciální zisk. Menší obchodníci většinou pro jeden obchod nepoužívají více jak 2-5 % dostupných peněz na účtu.

Další věc, kterou si musí obchodník určit je to, zda bude po celou dobu obchodování volit stejně velkou pozici, anebo zda ji bude upravovat podle právě dostupných prostředků. Při upravování velikosti pozice se zvyšuje ziskovost ziskového systému, protože s narůstajícím kapitálem jsou otevírány větší pozice a snižují ztráty neziskového systému, protože otevírané pozice jsou menší tím, jak klesá stav obchodníka účtu. Na druhou stranu se však systém po ztrátové sérii díky menším obchodům déle zotavuje, a pokud po úspěšné sérii přijde neúspěšný obchod, tak kvůli větším pozicím jsou u něj vyšší ztráty.

3.3.3. Výstup z obchodu

Ukončení, a tedy výstup z obchodů se dá provést dvěma způsoby, a to buď nastavením profit targetu, nebo nastavením stop lossu. Nastavením profit targetu stanovujeme cenu, při jejímž překročení se obchod ukončí a my bereme výsledný zisk. Nastavením stop lossu zase stanovujeme hranici, při které ukončíme obchod v případě, že jde trh proti našemu obchodu a chráníme se tak proti přílišným ztrátám. Stop loss se dá použít i stylem tzv. trailing stop lossu. V takovém případě, když začne být obchod profitabilní, tak posouváme hranici stop lossu takovým způsobem, abychom si uchránili již nabytý zisk. Toto posunutí je ale potřeba dělat s citem, aby nedošlo k předčasnému výstupu z obchodu.

Někteří obchodníci také pro své obchodování používají systém tzv. vícenásobných výstupů, kdy při vstupu do obchodu otevřou dvě pozice. První z nich uzavřou v momentě,

kdy se jim pokryjí náklady za vstup do obchodu. Druhou pozici pak nechávají otevřenou pro vytvoření jimi požadovaného zisku.

3.3.4. Trend a velikost finanční páky

Pro omezení podstupovaného rizika je dobré, když obchodník obchoduje ve směru dlouhodobějšího trendu (3). Při takových obchodech je zpravidla dosahováno menších ztrát a větších zisků, než když obchodník obchoduje proti trendu. Obchodování proti trendu by především začínající obchodníci měli přenechat těm zkušenějším.

Dalším rizikem, které může především začínající obchodníky rychle připravit o peníze je využití příliš velké páky. Vidina větších zisků láká k volbě co největší finanční páky. Tím, jak roste finanční páka, ale kromě zisků, rostou i případné ztráty. Pokud obchodník nemá dostatek zkušeností, aby přešel sérii ztrátových obchodů, tak se může rychle připravit o finance, které chtěl původně zhodnotit.

3.4. Způsoby analýzy trhu

Aby mohl obchodník na trhu být úspěšný, musí provést analýzu, ze které stanoví chování trhu v daný okamžik. Obchodník se tím pokouší odhadnout, jak by se trh mohl chovat do budoucna a tedy vyvodit, jaké obchody může uskutečnit. V následující kapitole jsou popsány hlavní přístupy k analýze trhu. Jedná se především o fundamentální a technickou analýzu. Někdy se lze setkat i s psychologickou analýzou (1 str. 95), jedná se však o velmi okrajovou záležitost, a proto není tato analýza v práci rozebírána.

3.4.1. Fundamentální analýza

Fundamentální analýza spočívá v analytické činnosti, kdy se obchodník na základě ekonomických, politických, statistických a dalších ukazatelů, tedy na základě známých informací a s využitím svých zkušeností snaží odhadnout chování trhu.

Fundamentální analýza se dá podle rozdělit podle typu fundamentů, který sledujeme na globální fundamentální analýzu, odvětvovou fundamentální analýzu a firemní fundamentální analýzu. (1 str. 101)

Globální fundamentální analýza se zabývá vlivem ekonomiky jako celku na pohyb kurzu. Sledují se zde veličiny jako HDP, státní rozpočty, úrokové míry daných států. Peněžní nabídka, inflace, ekonomické a politické šoky, pohyb zahraničního kapitálu a další ekonomické faktory, které mohou mít vliv na pohyb daného instrumentu.

Odvětvová fundamentální analýza se na druhou stranu již zabývá specifickým odvětvím. Snaží se specifikovat jeho rysy, hledá v daném odvětví vztahy a vazby a další specifika a snaží se stanovit, jaký mohou mít vliv na pohyb kurzu. Významnou roli zde hraje struktura daného odvětví (monopolní, konkurenční, ...), zda je odvětví nějak regulováno, jaký je aktuální hospodářský cyklus či v jaké fázi životního cyklu se odvětví právě nachází.

Firemní fundamentální analýza se přímo zabývá konkrétní společností či jedním daným instrumentem. Snaží se tedy odhadnout vnitřní hodnotu dané akcie nebo instrumentu. Po určení této vnitřní hodnoty se dá stanovit, zda je hodnota, za kterou je instrument obchodován nadhodnocená či podhodnocená. Dle toho následně může obchodník uvažovat o zahrnutí instrumentu do svého investičního portfolia.

3.4.2. Technická analýza

Technická analýza se soustředí především na sledování pohybu kurzu v čase, objemu obchodů hledání minim a maxim v ceně, sledování otevíracích a uzavíracích cen, a to vše s cílem, aby investor nakupoval právě v momentě, když se cena pohybuje kolem tržního minima a prodával, když se cena pohybuje kolem maxima. Cílem tedy není zkoumání vnitřní hodnoty instrumentu, ale sledování vývoje kurzu, hledání trendu a určení jeho síly a snaha předpovědět změnu nastoupeného trendu. Obchodník se pomocí technické analýzy snaží zjistit kdy a co se na trhu stane. Na rozdíl od fundamentální analýzy už však obchodník nesleduje, proč daná událost nastala. Technická analýza se dá shrnout do tří tezí (4 str. 13):

- Trh absorbuje veškeré informace o daném instrumentu

Všechny informace, které ovlivňují cenu instrumentu, se promítnou do jeho ceny. Díky tomu, že různí investoři se k těmto informacím dostanou různě rychle a také potřebují různý čas na jejich zpracování, tak kurz na tyto informace reaguje postupně, vzniká tak určitý trend a cílem obchodníka je ho najít, určit jeho směr a sílu a toho následně využít pro své obchody.

- Ve vývoji kurzů existují vzory

Analýzou historických dat a historického vývoje kurzů lze v tomto vývoji najít určité vzory, obrazce či formace. Tyto vzory je možné najít jak ve dlouhodobém, tak ve střednědobém a krátkodobém časovém horizontu. Při dobré znalosti těchto formací, a hlavně při jejich úspěšném a včasném identifikování lze následně na trhu s jistou mírou pravděpodobnosti odhadnout, jaký vývoj bude dále následovat.

Technická analýza také vychází z předpokladu, že tyto formace mají charakter fraktálu, a tedy že je možné je najít na grafu nezávisle na použitém časovém rámci.

- Historie se opakuje

Formace, které se ve vývoji kurzu objevují, se s postupem času nemění a lze najít stále ty stejné opakující se formace dokola. Toto lze vysvětlit tím, že vývoj kurzu je ovlivněn především myšlením a psychikou obchodníků, kteří daný instrument obchodují, a to se v čase příliš nemění. Obchodníci totiž na podobné události a zprávy, které mají na instrument vliv, reagují podobným způsobem.

Technická analýza bohužel nikdy není dokonale přesná. Pracuje se zde především s pravděpodobnostmi. Problémem je to, že v grafech sice lze najít opakující se formace, podobnosti a trendy, ty se však nikdy neopakují dokonale stejně. Trhy na podobnou událost mohou v jednu chvíli reagovat někdy rychleji, jindy pomaleji, s větší silou i se silou menší, kdy se kurz pohne jen velmi málo. Dalším problémem je, že metody technické analýzy se mohou zpoždovat za reálným kurzem, generovat falešné signály, mohou nabízet více způsobů interpretace nebo vzniklé signály nemusí být užitečné vůbec. Pomocí technické analýzy nelze předpovědět budoucnost, a tak k úspěšnému obchodu záleží především na zkušenostech a schopnostech daného obchodníka a na tom, jaké z výstupů technické analýzy učiní závěry.

3.5. Cenový graf

Graf v obchodování standardně znázorňuje vývoj ceny daného instrumentu v čase. Kromě toho mohou některé typy grafů navíc například znázorňovat i objem provedených obchodů v daném časovém období a další pro obchodníka důležité údaje. V grafech bývá nejčastěji na ose X znázorněn čas a na ose Y cena instrumentu.

3.5.1. Line chart

Line chart je nejjednodušší graf v technické analýze. Údaje o ceně (nejčastěji závírací cena pro dané časové období) jsou vynášeny do grafu a spojeny čarou. Výhodou tohoto grafu je jeho přehlednost a jednoduchost. Informace v něm obsažené jsou velmi čitelné i pro začátečníka. Nevýhodou tohoto grafu je především pro zkušenějšího obchodníka menší vypovídací hodnota, protože graf neobsahuje informace o minimech, maximech a otevíracích cenách pro jednotlivá časová období.



Obrázek 4: Hodinový sloupkový graf EURUSD s ukazatelem aktuální ceny.
Zdroj: vlastní tvorba

3.5.2. Bar chart

Bar chart, česky sloupkový graf, je pokročilejší typ grafu, který kromě zavírací ceny obsahuje také informace o otevírací ceně, cenovém maximu a cenovém minimu. Každý časový rámec je znázorněn sloupcem se dvěma zobáčky. Výška sloupce je dána minimem a maximem daného časového rámce a zobáček nalevo od sloupce zobrazuje otevírací cenu a zobáček napravo cenu zavírací. V tomto grafu mohou vznikat mezery, a to buďto pokud pro daný časový rámec není dostatečné množství obchodů, anebo pokud ceny předchozího rámce končí jinde, než začíná rámec nový (například z důvodu změny ceny v době zavřeného trhu).



Obrázek 5: Hodinový sloupkový graf EURNZD.
Zdroj: vlastní tvorba

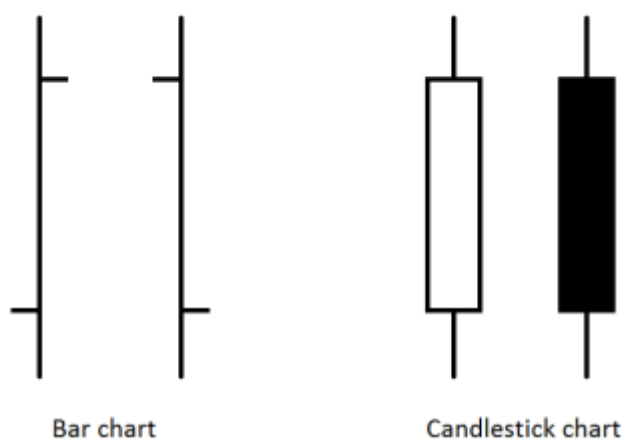
3.5.3. Candlestick chart

Candlestick chart, někdy také *japanese candlestick chart* a česky *svíčkový graf*, je graf velmi podobný sloupkovému grafu. Také zobrazuje minimum, maximum, otevírací a zavírací cenu, ale na rozdíl od sloupkového grafu jsou ceny vyneseny na tzv. svíci. Ta je tvořena tělem, jehož výška je dána právě otevírací a zavírací cenou a dvěma „knoty“, jejichž délku určuje maximum (nad tělem svíce) a minimum (pod svíci). Svíčkový graf je velmi přehledný a spolu se sloupkovým grafem patří mezi nejpoužívanější typy grafů. Na tom, jaký mají jednotlivé svíčky tvar a jak na sebe navazují, lze provádět další případnou analýzu chování trhu.



Obrázek 6: Denní svíčkový graf na měnovém páru GBPJPY
Zdroj: vlastní tvorba

Na obrázku níže je znázorněn rozdíl mezi vzestupným a sestupným sloupcem u sloupkového a u svíčkového grafu.

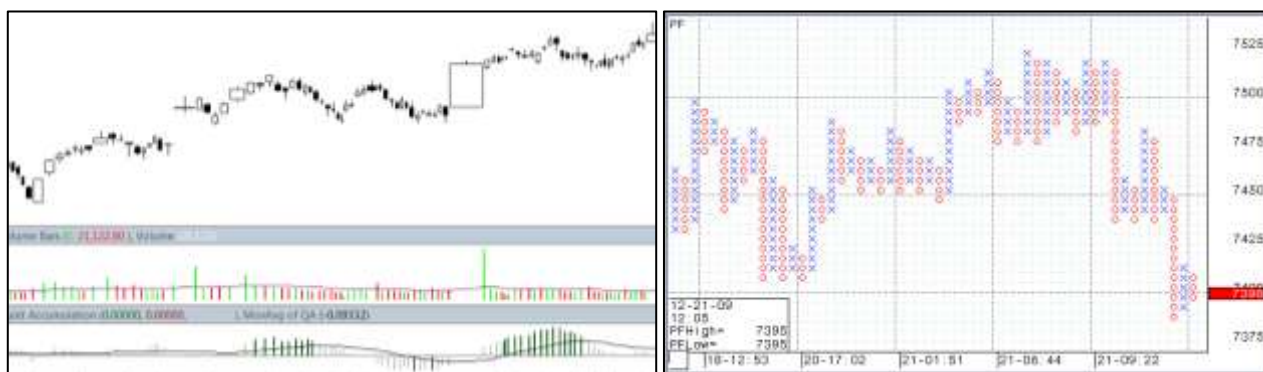


Obrázek 7: Rozdíl sloupců u sloupkového a svíčkového grafu. Zdroj: vlastní tvorba

3.5.4. Další typy grafů

Kromě tří výše uvedených a nejrozšířenějších typů grafů existují i další, méně známé. Příkladem může být point and figure chart, který vývoj ceny zakresluje pomocí symbolů O pro pokles ceny a X pro její vzestup. Ten je vhodný ke sledování trendů na trhu, je u něj však nutné dobře nastavit jeho parametry a jeho nevýhodou je jisté zpoždění zobrazených informací vůči reálnému dění na trhu.

Dalším grafem může být candlevolume chart, který vychází ze svíčkového grafu, ale navíc v něm šířka jednotlivých svící závisí na objemu provedených obchodů v časovém rámci. Výhodou tedy u něj je obsažená informace o objemu obchodů, na druhou stranu je zde však nevýhoda v chybějícím časovém rámci, který je zkrácen právě různou šíří svíček. Dále pak například grafy kagi chart, ranko chart, three line break chart a další.



Obrázek 8: Candlevolume chart a Point and figure chart.
Zdroj: <http://www.learncharts.com>

3.6. Metody technické analýzy

Pro praktickou aplikaci technické analýzy na některém z trhů existuje nepřehledné množství různých technických indikátorů. Tyto indikátory vycházejí z různých idejí a pohledů na analýzu trhu. Jejich konstrukce může zohledňovat cenu, objem obchodů, rychlost změny ceny a další faktory. Indikátory se liší vypovídací schopností, složitostí, spolehlivostí a samozřejmě i oblíbeností mezi obchodníky. Pro lepší přehlednost jsou představené indikátory rozděleny do několika kategorií. Toto členění není pevně dané. Některé indikátory mohou mít vlastnosti z více kategorií, a tak velmi záleží, jak daný indikátor obchodník použije.

3.6.1. Trendové indikátory

Trendové indikátory jsou indikátory, které se snaží následovat právě probíhající trend na trhu. Z těchto indikátorů je velmi snadné vyčíst, zda je na trhu právě nastoupen býččí či medvědí trend a případná síla tohoto trendu.

Kvůli závislosti těchto indikátorů na vývoji ceny v čase, mají tyto indikátory oproti reálné situaci na trhu jisté zpoždění. To je dáno právě tím, jak velké časové období zpětně sledují pro vyhodnocení aktuální situace. Kvůli tomu i signály generované těmito indikátory přicházejí se zpožděním a mohou tak přijít v okamžiku, kdy značná část trendu již proběhla nebo dokonce když už je trend u konce či došlo k jeho obrácení. Na druhou stranu však tyto indikátory mají výhodu v tom, že vyhlazují aktuální průběh ceny. Patří sem všechny typy klouzavých průměrů a další indikátory na nich založené.

Klouzavé průměry

Nejjednodušší technický trendový ukazatel jsou klouzavé průměry (5). Základním klouzavým průměrem je jednoduchý klouzavý průměr, který pro stanovení aktuální hodnoty spočítá průměr z cen za N posledních období, tzv. periodu. Pro výpočet se může použít jak open, tak high, low či close cena daného časového období. Velikost vzorku N, tedy počet časových období zahrnutých do výpočtu má vliv na vyhlazení ceny. Čím je časové období delší, tím je cena více vyhlazena, ale tím také vzrůstá zpoždění tohoto indikátoru za aktuálním stavem na trhu. Naopak s kratší periodou je zvýšena citlivost tohoto indikátoru. Jednoduchý klouzavý průměr se počítá pomocí následujícího vzorce:

$$SMA_N = \frac{P_N + P_{N-1} + P_{N-2} + \dots + P_0}{N}$$

kde N je počet zvolených časových období, P je cena daného časového období a index u ceny P určuje pořadí období ve zvoleném časové periodě.

Na následujícím obrázku jsou znázorněny tři klouzavé průměry s různou délkou periody. Nejrychlejší průměr s periodou deset nejlépe kopíruje cenu instrumentu, ale hůře se z něj odhaduje dlouhodobější vývoj trendu. Na pomalejším, zeleně znázorněném průměru, s periodou 50 již je vývoj trendu lépe zřetelný a na nejpomalejším klouzavém průměru, znázorněném modře, je jasně vidět dlouhodobější trend, ale je na něm také vidět i největší zpoždění, kdy trend po poklesu začíná růst o dost později, než kdy začala růst cena.



Obrázek 9: Ukázka klouzavých průměrů. Zdroj: vlastní tvorba

Pokročilejším časovým průměrem je vážený klouzavý průměr, který jednotlivým cenám navíc přiřazuje lineárně klesající váhu a to tak, že největší váhu má nejaktuálnější cena. Přiřazením vyšší váhy novějším cenám je dosaženo vyšší citlivosti tohoto indikátoru na aktuální situaci na trhu. Výpočet váženého klouzavého průměru je následující:

$$WMA_N = \frac{NP_N + (N - 1)P_{N-1} + (N - 2)P_{N-2} + \dots + P_0}{N + (N - 1) + (N - 2) + \dots + 1}$$

Kde N je počet zvolených časových období, P je cena daného časového období a index u ceny P určuje pořadí období ve zvoleném časové periodě.

Ještě citlivější ale zato složitější je exponenciální klouzavý průměr, který jednotlivým cenám přiřazuje exponenciálně rostoucí váhy. Váha pro nejaktuálnější cenu je stanovena pomocí tzv. exponenciálního procenta či vyhlazovacího faktoru. To si buď může analytik sám stanovit a tím bude dána časová perioda pro výpočet tohoto průměru, nebo si stanoví časovou periodu a vyhlazovacího faktor se vypočítá z ní. Výpočet probíhá podle vzorce $ep = \frac{2}{M+1}$ pokud máme stanovenou časovou periodu, případně podle vzorce $M = \frac{2}{ep} - 1$, pokud víme vyhlazovacího faktor. Citlivost tohoto průměru je dána právě vyhlazovacím faktorem. Samotný exponenciální průměr se počítá vzorcem:

$$EMA_0 = ((1 - ep)EMA_{-1}) + (P_0ep)$$

Kde ep je vyhlazovací faktor, P je aktuální cena, EMA je exponenciální klouzavý průměr a indexy 0 a -1 značí aktuální a předchozí časové období.

Z dalších klouzavých průměrů lze jmenovat například proměnlivý klouzavý průměr, který automaticky přizpůsobuje váhu aktuálního kurzu podle variability dat. Při větších výkyvech ceny citlivost průměru roste. Klouzavé trendy s pevně stanovenou vahou většinou nelze správně nastavit tak, aby fungovaly na trzích se silným i se slabým trendem, s čímž se snaží tento průměr vypořádat. Vychází z exponenciálního průměru, a tak je jeho další vlastností i větší citlivost.

Objem provedených obchodů do svého výpočtu zahrnuje klouzavý průměr přizpůsobený objemu (volume adjusted moving average). Ten dává vyšší váhu těm časovým úsekům, kdy byl větší objem uskutečněných obchodů.

MACD

Indikátor MACD (moving average convergence divergence) je indikátor založený na klouzavých průměrech. Má však také i některé charakteristiky typické pro oscilátory. Z podstaty trend following indikátorů následuje trh, a tedy poskytuje signály až s určitým zpožděním. Umí však poměrně dobře sledovat hlavní trend a odfiltrout zbytečné signály (6).

MACD se konstruuje pomocí dvou klouzavých průměrů s různou časovou periodou, mezi kterými se vypočítá rozdíl (případně podíl, dle konkrétní implementace). Hodnota tohoto indikátoru následně osciluje kolem nulové hodnoty. Ta odpovídá dlouhodobému, pomalejšímu z klouzavých průměrů. Pokud se trh nachází v rostoucím trendu, tak indikátor nabývá kladných hodnot, a naopak při klesajícím trendu nabývá hodnot záporných.

Indikátor MACD může být doplněn ještě o spouštěcí linii, tzv. trigger line. Ta bývá počítána jako klouzavý průměr s ještě kratší periodou, než mají průměry použité pro výpočet MACD. Za signály generované tímto indikátorem se dají považovat průsečíky MACD s oscilační linií, průsečíky MACD a trigger line a vychýlení indikátoru MACD do extrémně kladných či záporných hodnot, indikující přeprodanost či překoupenost trhu.

Na obrázku níže je pod cenovým grafem černými svislými čarami vykreslen indikátor MACD, doplněn o červenou spouštěcí linii.

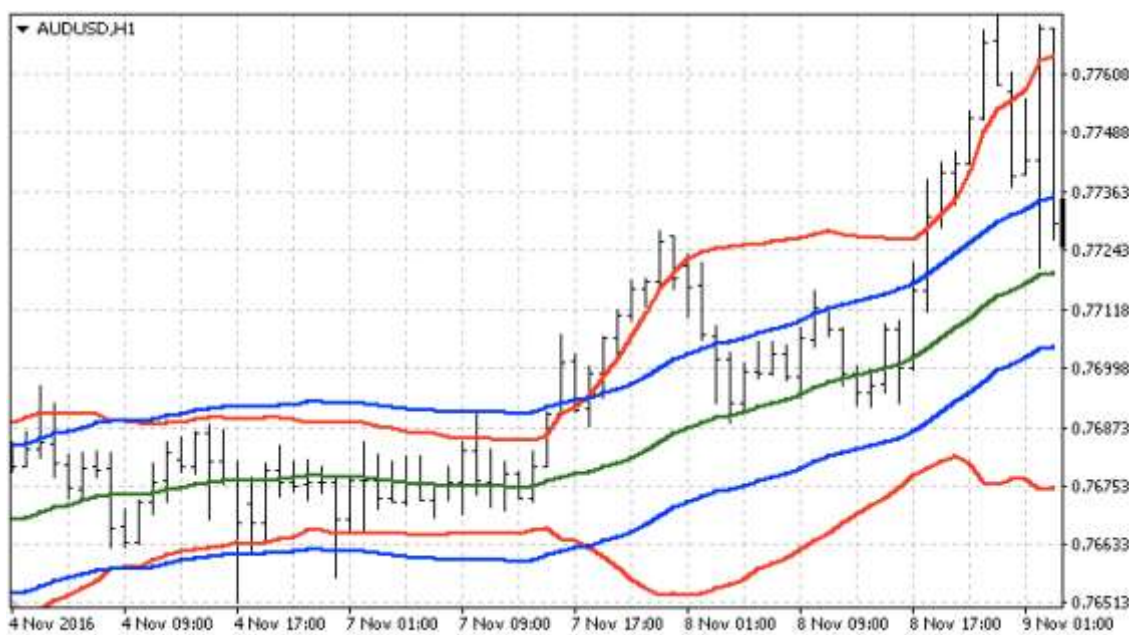


Obrázek 10: Indikátor MACD. Zdroj: vlastní tvorba

Bollingerovy pásy

Bollingerovy pásy také vycházejí z klouzavého průměru. Zakreslují se do cenového grafu, na kterém vytvoří pásmo kolem ceny instrumentu, jehož překročení či alespoň přiblížení se ke hraně pásma, může pro obchodníka znamenat signál k provedení obchodu. Bollingerovy pásy vycházejí z indikátoru obálek (envelope), které představují pouhý klouzavý průměr posunutý o určitý kus nahoru a dolů. Na rozdíl od obálek, však toto posunutí u tohoto indikátoru není pokaždé stejné, ale míra posunutí se počítá podle směrodatné odchylky vypočítané k aktuální ceně. Díky zahrnutí směrodatné odchylky se Bollingerovy pásy zužují v momentě, kdy má cena menší volatilitu, a naopak rozšiřují při větší volatilitě, a tedy velkých pohybech ceny.

Ukázka bollingerových pásů je na spodním obrázku. Bollingerovy pásy jsou vyobrazeny červeně a indikátor obálek je vyobrazen modře. Je zde zřetelně vidět rozšíření bollingerových pásů poté, co se zvýšila volatilita ceny. Obálky oproti tomu zůstávají stále stejně vzdálené od klouzavého průměru znázorněného zelenou barvou.



Obrázek 11: Ukázka bollingerových pásem. Zdroj: vlastní tvorba

Parabolic SAR

Tento indikátor patří mezi další zástupce trend following indikátorů. Nejčastěji se tento indikátor používá ke stanovení hranic stop-loss, případně ke stanovení obrátů v trendu (7). Z toho důvodu je vhodný pro strategie, které se snaží zohlednit risk management a ochranu zisků. Tento indikátor popsal J. W. Wilder v knize *New Concepts in Technical Trading Systems*.

Samotný indikátor je tvořen sérií bodů, které vyhlazeně kopírují trend. Při stoupajícím trendu se nachází pod aktuální cenou a při klesajícím trendu nad aktuální cenou. Indikátor se snaží využít skutečnosti, že nastoupené trendy mají ze začátku často výraznější pohyb, který postupně slábne podle toho, jak dochází k vysílení trendu. Proto indikátor ve svém výpočtu využívá tzv. akcelerační koeficient, který postupně roste a indikátor se tak přibližuje více a více k aktuální ceně. V momentě kdy je již trend vyčerpán a cena překročí hodnotu indikátoru, je dán signál pro obchodníka, aby uzavřel svůj obchod.

Jako u každého jiného indikátoru využívajícího trendy je i pro tento indikátor potřebné, aby daný trh byl trendový. V opačném případě mohou generované signály vést ke značně úbytkům z obchodníkovu účtu.

Praktická ukázka indikátoru je na obrázku číslo 12. Zde je vidět reakce indikátoru, znázorněného posloupností červených bodů, na pohyb ceny. Následně jsou zde znázorněny potenciální obchodní příležitosti pro nákup (modře) a pro prodej (zeleně).



Obrázek 12: Ukázka indikátoru Parabolic SAR. Zdroj: vlastní tvorba

3.6.2. Oscilátory

Oscilátory jsou indikátory, jejichž hodnota osciluje od jednoho extrému do druhého kolem středové linie v závislosti na situaci na trhu. Jejich hodnota se buďto může pohybovat mezi mezními hodnotami (např. ± 100), či jejich pohyb nemusí být omezený vůbec.

Index relativní síly

Index relativní síly (relative strength index, dále jen RSI) je starší indikátor původně vytvořený pro sledování trhů s komoditami. Jedná se o pomalejší indikátor, který se snaží změřit vnitřní relativní sílu daného instrumentu.

Pro výpočet indikátoru RSI je potřeba spočítat průměr kladných a záporných relativních změn za danou časovou periodu a z ní následně vypočítat hodnotu indikátoru pomocí vzorce:

$$RSI = 100 - \frac{100}{1 + \frac{U}{D}}$$

kde U je průměr kladných a D průměr záporných kurzových změn za danou časovou periodu.

Jako časová perioda nejčastěji bývají zvoleny úseky 9, 14 nebo 25 dní. Záleží však na konkrétním obchodníkovi, jak chce indikátor na daný trh použít a jaká perioda mu nejvíce vyhovuje. Hodnoty indikátoru se pohybují v rozmezí 0-100, kdy vysoké hodnoty značí, že trh může být překoupený a nízké hodnoty zase to, že trh může být přeprodáný.

Ukázka indikátoru je na obrázku níže, kde je na začátku sestupného trendu vidět překoupenost trhu a indikátor zde dosahuje velmi vysokých hodnot. Naopak na konci trendu, kde cena dosahuje lokálního minima, má indikátor velmi malou hodnotu, předpovídající přeprodanost a tedy možný konec sestupného trendu.



Obrázek 13: Ukázka indikátoru RSI. Zdroj: vlastní tvorba

Stochastic

Indikátor stochastic patří mezi oscilátory s pevně vymezeným pásmem oscilace. Indikátor porovnává zavírací cenu s celkovým cenovým rozpětím za dané časové období. Vychází z předpokladu, že instrument s rostoucí cenou bude mít zavírací cenu poblíž horních maxim, a naopak instrument s klesající cenou poblíž minimálních cen za předchozí období. Indikátor se počítá pomocí vzorce:

$$S = \frac{\text{uzavírací cena} - P_{MAX}}{P_{MAX} - P_{MIN}}$$

kde S je výsledná hodnota indikátoru, P_{MAX} je maximální cena za daný počet období a P_{MIN} je minimální cena za daný počet období.

Samotný stochastic je sice poměrně rychlý indikátor, protože se většinou počítá s časovým úsekem kolem 5 období, bývá ale kvůli tomu také poměrně volatilní a generuje

tak velké množství falešných signálů. Kvůli tomu se většinou také zakresluje k samotnému stochasticu i jeho průměr za posledních několik časových období. V takovém případě se samotný indikátor značí jako %K a jeho průměr za %D.

Pro obchodníka je pak důležité, když nastane některá z následujících událostí:

- Protnutí křivek %K a %D
- Hodnota indikátoru spadne pod hranici 20 %, nebo vystoupá nad hranici 80 %

Na ukázce indikátoru na obrázku 14 je dobře patrný moment, kdy indikátor dosahuje nízkých hodnot, symbolizujících možné obrácení trendu. K obrácení trendu nakonec došlo a na indikátoru je vidět, jak přechází do vysokých hodnot v oblasti cenového maxima.



Obrázek 14: Indikátor stochastic. Zdroj: vlastní tvorba

3.7. Automatický obchodní systém

Automatický obchodní systém (dále AOS) je systém, který provádí obchodování automaticky, bez zásahu obchodníka. Aby mohl systém obchodovat automaticky, potřebuje mít sadu pravidel, podle kterých určí okamžiky pro vstup do obchodu a výstup z obchodu. Špatné sestavení těchto pravidel vede k vytváření ztrátových obchodů a systém poté není profitabilní. Systém také může dynamicky upravovat velikost otevíraných pozic (např. dle velikosti účtu obchodníka).

V dnešní době je na trhu velké množství těchto systémů. U velkých účastníků trhu jsou takové systémy nutností, protože jejich počty se pohybují v řádu desítek i stovek obchodů za sekundu na desítkách různých trhů. Pro malého obchodníka je k provozování

automatického obchodního systému nutné mít program, který toto podporuje a ve kterém si vytvoří svůj AOS a poté připojení k internetu, přes který se obchody zasílají k brokerovi.

Výhodou automatického obchodního systému je to, že nepodléhá žádným emocím, provádí příkazy strojově přesně. AOS také může obchodovat v době, kdy normální obchodník nemůže, například když spí. Zároveň také takovýto systém nemá problém s koncentrací. Pokud chce obchodník obchodovat intradenně, a případně ještě na několika trzích, tak je samotné obchodování velmi vyčerpávající. Obchodník se zvládne plně koncentrovat na trh a vyhodnocovat jeho stav po dobu maximálně několika hodin. AOS však takový problém nemá a zvládne se koncentrovat 24 hodin denně. Nevýhodou AOS však je, že tento systém nikdy nebude mít o trhu tak komplexní přehled jako obchodník a ani se nezvládne poučit ze špatných obchodů či adaptovat na nové podmínky, které mohou nastat. Je tak na obchodníkovi, aby systém správně definoval a případně upravoval podle svých potřeb a nově vzniklých požadavků.

3.8. Platformy pro vývoj AOS

V dnešní době je na výběr velké množství platforem, na kterých se dají tyto systémy tvořit. Liší se funkcionalitou, kterou v základu nabízejí (např. již mohou obsahovat implementace základních indikátorů), velikostí komunity kolem platformy, dostupností studijních materiálů i programovacím jazykem, který se v nich používá. Mezi nejčastěji používané platformy patří:

- MetaTrader 4
- MetaTrader 5
- cTrader
- Ninja Trader
- a další

Za zmínku také jistě stojí to, že obchodník může využít API (application programming interface) některého z brokerů. API je rozhraní pro komunikaci mezi programy, v tomto případě mezi serverem brokera a programem obchodníka. Obchodník tak toto rozhraní může využít při tvorbě úplně nezávislého obchodního systému, který nebude vytvořen v žádné z výše uvedených platforem. Obchodník si poté může naprosto svobodně zvolit

programovací jazyk, ve kterém chce svůj systém vytvořit. Nejčastěji se jedná o programovací jazyk Java, C#, Python nebo C++. Pro analýzu historických dat je poté velmi vhodný nástroj R.

Brokeři, kteří nabízejí obchodníkům přístup přes API, často také nabízejí ukázkové aplikace, ze kterých může obchodník vycházet, či knihovny, které může využít. I přesto je však náročnost vývoje vlastního systému využívajícího API velmi složitá záležitost, a tak se pro jednotlivé obchodníky jedná spíše o okrajovou záležitost. Pro menší společnosti to však již může být zajímavá alternativa, nabízející podstatně širší možnosti.

3.8.1. MetaTrader

Platforma MetaTrader 4 od společnosti MetaQuotes patří mezi jednu z nejoblíbenějších a nejrozšířenějších platforem pro obchodování na trzích. Mezi její hlavní výhody (8) patří:

- Možnost sledovat několik grafů najednou
- Obchodování pomocí jednoho kliknutí
- Velké množství vestavěných indikátorů
- Možnost tvorby vlastního indikátoru
- Možnost tvorby a následného ladění AOS

Ač je MetaTrader 4 již starší platformou, tak i přesto se díky velké komunitě stále těší obrovské oblibě. Díky této komunitě je možné nalézt mnoho zdrojů s dalšími informacemi o indikátorech a jejich hodnocení, o automatických obchodních systémech, jejich tvorbě a optimalizaci a další věci, které by mohl obchodník potřebovat.

O něco novější je platforma MetaTrader 5. Ta nabízí větší množství nabízených indikátorů, nový programovací jazyk pro tvorbu indikátorů a AOS, forexový kalendář s důležitými událostmi pro fundamentální analýzu a další drobná vylepšení. I přesto však většina obchodníků a brokerů stále funguje se starší platformou a na novější nepřechází.

Společnost MetaQuotes naštěstí stále podporuje a aktualizuje obě platformy, a tak obchodníci nemají potřebu přecházet na novější.

Na následujícím obrázku je ukázka rozhraní programu MetaTrader 4, s přehledem aktuálních cen významných měnových párů, grafem pro měnový pár EURUSD, seznamem otevřených obchodů ve střední části okna programu a rozhraním pro testování strategií v dolní části okna.



Obrázek 15: Rozhraní programu MetaTrader 4. Zdroj: vlastní tvorba

3.8.2. cTrader

Program cTrader od společnosti Spotware má také mnoho užitečných vlastností, z nichž podle webu společnosti (9) stojí za zmínku například:

- Rychlé provádění příkazů využitím STP sítě
- Velké množství grafů a nástrojů pro práci s nimi
- Lokalizace do několika světových jazyků
- Vlastní indikátory
- Backtesting

cTrader pro tvorbu AOS nabízí platformu cAlgo, která je v cTraderu dobře integrovaná a spolupracuje s ním. Velikou výhodou této platformy je využití již existujícího programovacího jazyka C#, jehož případná znalost může podstatně urychlit počáteční vývoj AOS.

3.8.3. NinjaTrader

Platforma NinjaTrader (10) nabízí pokročilé možnosti práce s cenovými grafy, přes 80 různých ukazatelů a vysokou přizpůsobitelnost aplikace pro uživatele. Dále nabízí testování vytvořených strategií, podporu uživatelsky vytvořených aplikací a addonů, vytváření upozornění, velmi rychlou exekuci příkazů. Skripty se na této platformě tvoří pomocí jejich vlastního programovacího jazyka NinjaScript, který vychází z jazyka C#.

Nevýhodou NinjaTraderu je mnohem nižší podpora ze strany brokerů. Ti většinou kromě vlastních řešení nabízejí pouze přístup přes platformu MetaTrader. Oproti předchozím dvěma uvedeným platformám je NinjaTrader považován za platformu pokročilejší a s mnohem větší nabídkou funkcí.

4. Analýza problému

V této kapitole jsou rozebrány stupně automatizace obchodování na Forexu, poté zde jsou popsány postupy, jak může obchodník získat automatický obchodní systém a jaké jsou na vývoj AOS kladeny hardwarové a softwarové požadavky. Dále jsou zde nastíněny možné přístupy k testování a optimalizaci systému a na závěr je zde popsáno co vše je potřeba k tomu, aby obchodník již hotový systém mohl spustit a nechat samostatně obchodovat

4.1. Úrovně automatizace obchodování

Pro obchodování na forexovém trhu má obchodník několik možností, jak bude obchodovat. Nejnáročnější metodou je manuální obchodování, kdy obchodník sleduje konkrétní trh, do grafů si zakresluje vybrané indikátory a vede si obchodní deník s jednotlivými obchody. Obchodník si však může ulehčit svou práci a čas a zautomatizoval některé své analytické a rozhodovací činnosti. Mezi nejzákladnější usnadnění patří využití softwaru pro výpočet technických indikátorů. Pro dnešního obchodníka je toto již samozřejmostí a většina obchodních platform, které brokeři nabízí, obsahuje velké množství indikátorů, které mohou obchodníci využít.

Vyšším stupněm automatizace je využití indikátorů pro generování vstupních a výstupních signálů, které poté obchodník po další analýze využije ke svému obchodování. Nevýhodou je, že po vytvoření signálu obchodníkovi nějakou dobu trvá, než ho vyhodnotí. Tento přístup je tedy vhodné spíše pro poziční obchodování, kdy má obchodník čas signál dobře zvážit. Při intradenním obchodování může obchodní příležitost pominout dříve, než obchodník signál vyhodnotí.

Poslední možností je využití automatického obchodního systému. Ten vyhodnocuje signály místo obchodníka a reaguje na ně ve zlomku vteřiny. Výhodou takového systému je, že signály vyhodnocuje vždy stejně, nemají na něj vliv emoce či únava a jeho úsudek je stále stejný. Nevýhodou však je, že OAS nikdy nebude mít tak komplexní a ucelený přehled o dění na trhu jako zkušený obchodník.

4.2. Získání automatického obchodního systému

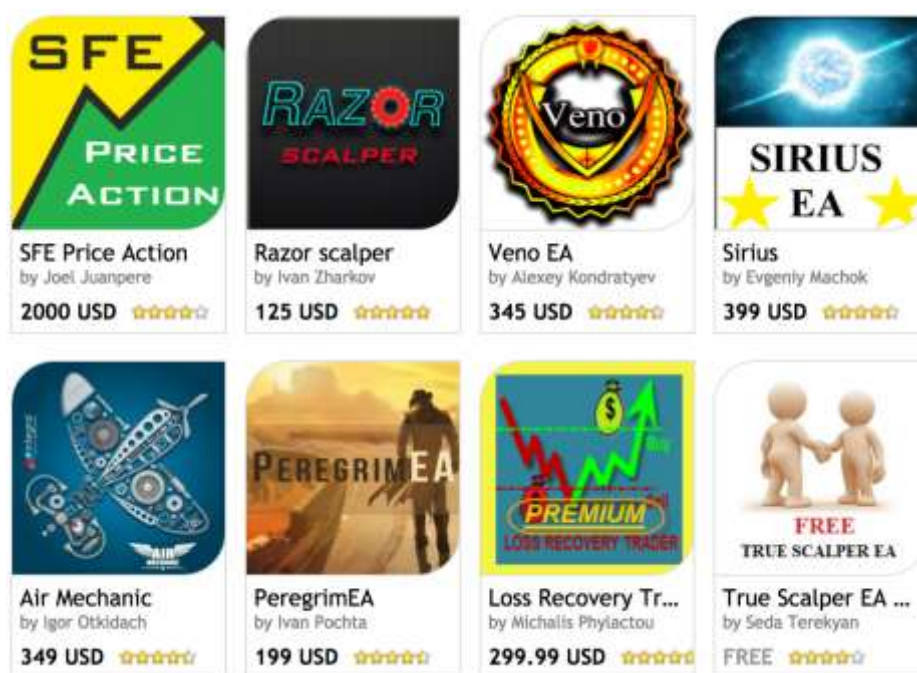
Obchodník může získat AOS několika způsoby:

- Využitím některého z volně dostupných systémů
- Zakoupením již vytvořeného systému
- Objednáním systému na zakázku

- Vytvořením vlastního systému

Nejjednodušší metodou, jak získat automatický obchodní systém, je sáhnout po některém z obchodních systémů volně dostupných na internetu. Tyto systémy však pro obchodníka znamenají častěji spíše ztrátu než zisk. Ve většině případů se jedná o velmi jednoduché systémy, které mohou sloužit buď jen pro ukázkou, nebo pro představení nějakého principu. Často vůbec neřeší věci jako je money management, který je však stejně důležitý jako samotná strategie.

Druhou možností je zakoupit někým vytvořený systém. Brokeři, ale i různé společnosti či jednotlivci nabízejí k prodeji pokročilejší obchodní systémy. Nevýhodou u tohoto způsobu je, že si obchodník zpravidla nemůže ověřit skutečnou funkčnost takového systému, dokud ho nezakoupí. Je logické, že si prodejci těchto systémů chrání své duševní vlastnictví a nenabízejí nahlédnutí například do zdrojových kódů. Bohužel, existují však prodejci, kteří tohoto zneužívají a nabízejí systémy, které na reálném trhu příliš nefungují. I přesto jsou však nabízeny jako vysoce ziskové a spolehlivé.



Obrázek 16: Ukázka nabídky obchodu s automatickými obchodními systémy. Zdroj: <https://www.mql5.com/en/market/mt4/expert>

Vyhnut se koupi špatného systému se obchodník může navrhnutím vlastního systému a jeho následným vytvořením třetí osobou na zakázku, dle jeho specifikací. Vytvoření takového systému musí obchodník zaplatit programátorovi, ale pokud má ověřenou a fungující strategii, jedná se rozhodně o lepší volbu než kupovat cizí, neověřené systémy

nabízené ostatními. Nevýhodou této možnosti je to, že obchodník potřebuje asistenci programátora na jakoukoliv úpravu systému, která přesahuje možnosti konfigurovatelného nastavení systému.

Poslední možností pro obchodníka je vytvořit si automatický obchodní systém sám. Obchodník k tomu musí mít základní logické a analytické myšlení a musí mít alespoň základní znalosti programování. Pokud je však obchodník natolik zručný, že již má vlastní funkční strategii, tak naučit se základy programování a strategii zprovoznit v automatickém obchodním systému už není tak složité.

4.3. Vývoj automatického obchodního systému

4.3.1. Požadavky kladené na AOS

K vývoji automatického obchodního systému musí nejprve obchodník mít navrženou strategii, kterou chce implementovat. Každá obchodní strategie má dva základní úkoly:

- Určit okamžik vstupu do obchodu
- Určit podmínku, při které bude obchod ukončen

Při pozičním obchodování se obchodník často snaží využít trendu, a tedy nějakého indikátoru, který měří jeho směr a sílu. Při nalezení dostatečně silného trendu poté potřebuje obchodník indikátor, který by ve správný okamžik generoval signály pro vstup do trhu. K tomuto účelu se dá použít některý z oscilátorů. Signály generované jakýmkoliv indikátorem však často nejsou jasné na 100 %. Na signály, které indikují správný moment vstupu do obchodu, připadá velké množství falešných signálů. Tyto falešné signály musí umět AOS odfiltrovat, protože jejich obchodování vede ke ztrátám. Na úspěšnosti filtrování dobrých signálů od špatných velkou měrou závisí úspěšnost celého obchodního systému.

Ještě před samotným otevřením obchodu je potřeba na základě stanoveného rizika spočítat, jak velký objem bude obchodován a jak bude nastaven stop loss. Tyto dvě veličiny jsou s případnou ztrátou u neúspěšného obchodu provázané, protože při ztrátovém obchodu je ztráta dána právě velikostí obchodu a pak tím, jak daleko od vstupní ceny byl nastaven stop loss.

Za předpokladu že je obchod profitabilní, tak musí obchodník stanovit, jak systém obchodu uzavře. Jednou z možností je nastavení profit targetu, při jehož dosažení bude pozice uzavřena a inkasován zisk. Výhodou nastaveného profit targetu je to, že se profitabilní obchod při jeho dosažení uzavře i v momentě, kdy AOS z nějakého důvodu neběží

(např. u důvodu výpadku). Nastavení profit targetu tedy umožňuje chránit potenciální zisk.

Druhou možností je vystoupení pomocí stop lossu. Aby z obchodu mohlo být vystoupeno pomocí stop lossu a obchod byl i přesto ziskový, tak je potřeba stop loss posouvat s tím, jak se hýbe kurz instrumentu ve prospěch obchodu a obchodníkovi narůstá zisk. To, jak rychle stop loss sleduje kurz, musí systém umět správně spočítat, protože při příliš rychlém přibližování dojde vlivem kolísání kurzu k ukončení obchodu dříve, než je dosaženo celého zisku z trendu, který obchod spustil. Pokud se stop loss přibližuje příliš pomalu, tak se zase zisk z obchodu může snížit s tím, jak trend skončí a kurz se začne vracet zpátky.

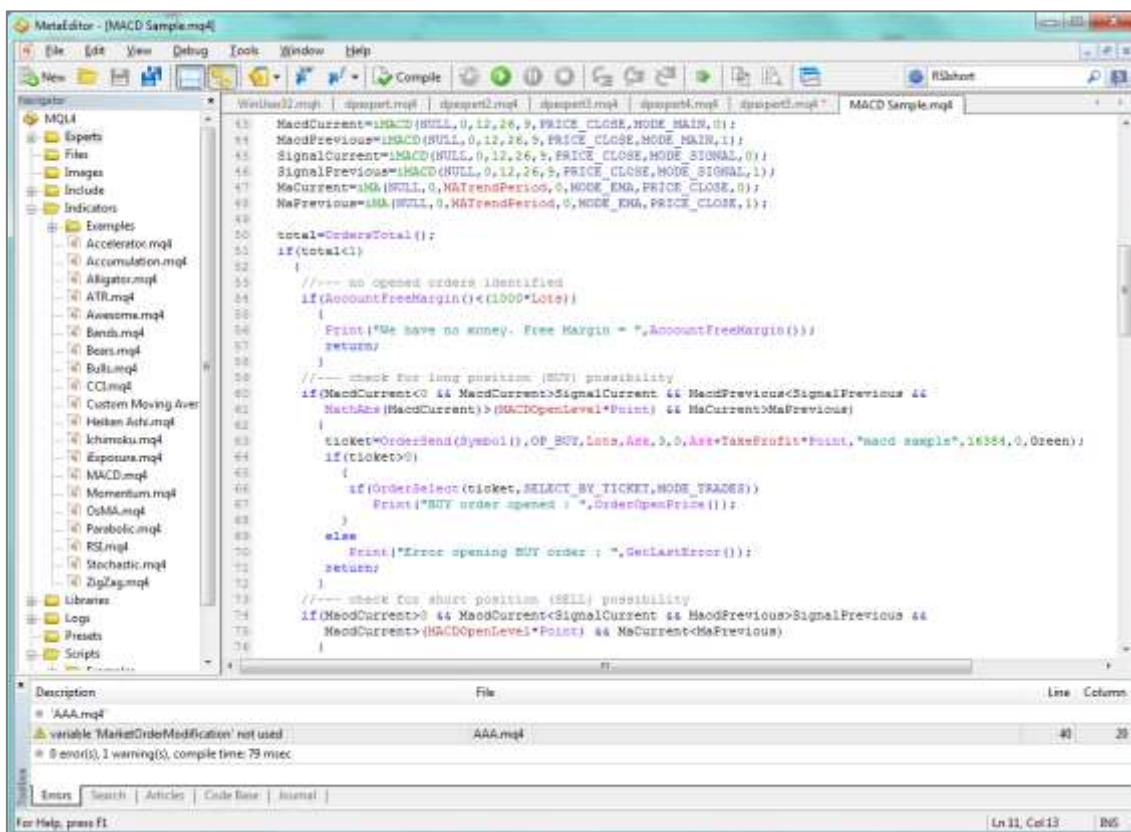
Ideálním případem je tedy systém, který opatrně přibližuje hranici stop loss a zároveň odsouvá hranici profit targetu těsně pod očekávanou hranici zisku do té doby, než je trend vyčerpán. V ten moment systém obchod ukončí. Pokud je toto splněno a správně v obchodním systému naimplementováno, tak jsou podstatně více chráněny vložené prostředky a je zvýšena šance na zisk obchodníka.

4.3.2. Programové a hardwarové vybavení pro tvorbu AOS

Z programového vybavení obchodníkovi pro tvorbu systému stačí téměř jakýkoliv počítač. Většinu času totiž obchodník stráví samotnou implementací, tedy psaním textu. To může probíhat v jednoduchém textovém editoru, v editoru nabízeném s platformou, kterou obchodník využívá či v nějakém pokročilém samostatném vývojovém prostředí.

V poslední době se také objevují platformy, které fungují na bázi software as service a celé vývojové prostředí a také prostředí pro testování, optimalizaci a běh obchodního systému nabízí online, odstupné přes internetový prohlížeč. Za zmínku takového poskytovatele stojí například služba Quantopian (11).

Na následujícím obrázku je ukázáno vývojové prostředí platformy MetaTrader. Většinu okna zabírá blok pro psaní zdrojového kódu. V seznamu vlevo má uživatel přehled o dostupných skriptech, indikátorech, knihovnách a dalších součástech, potřebných pro vývoj a ve spodní části okna jsou zobrazeny chyby ve skriptu a případný log z běhu skriptu.



Obrázek 17: Ukázka programovacího prostředí platformy MetaTrader 4. Zdroj: vlastní tvorba

4.4. Testování a optimalizace automatického obchodního systému

4.4.1. Testování AOS

Testování automatického systému je proces, kdy je na základě historických dat simulován běh systému a výsledek simulace je poté předložen obchodníkovi, který ho vyhodnotí a případně na jeho základě provede úpravy systému. Pro testování obchodních systémů mají platformy integrované testery strategií. Pro jejich dobrou výpovědní schopnost je důležité, aby obsahovala kvalitní historická data, která nemají slepá místa.

Každý systém by měl být vždy pořádně otestován předtím, než je spuštěn na účtu pro reálné obchodování. Důvodem pro testování je ověření správné implementace systému, aby například nedošlo k situaci, kdy obchodník při tvorbě AOS omylem prohodí nákupní a prodejní příkazy apod.

Druhý důvod pro testování je ten, že každý trh má odlišné a unikátní charakteristiky jako je jeho trendovost, likvidita, volatilita, stabilita atd. Tyto charakteristiky se navíc i mění v čase, a tak musí být systém odladěn tak, aby s nimi správně fungoval. I když má obchodník při návrhu často již nastavené hodnoty jednotlivých vstupních proměnných

obchodního systému (jako je např. délka klouzavých průměrů), tak pro další doladění je vhodné tyto parametry dále optimalizovat.

4.4.2. . Optimalizace AOS

Optimalizace AOS je proces, kdy je systém spouštěn stále dokola na stejných historických datech, ale s různými vstupními parametry. Pro každý jednotlivý běh jsou pak zaznamenány statistiky jako:

- Celkový zisk
- Celková ztráta
- Čistý zisk
- Počet ziskových obchodů
- Počet ztrátových obchodů
- Největší pokles vloženého kapitálu (tzv. drawdown)
- Nejdelší série ziskových a ztrátových obchodů
- Risk-Reward ratio – poměr průměrných zisků na obchod k průměrným ztrátám na obchod
- Profit factor – podíl celkových zisků a celkových ztrát všech obchodů

Podle výsledků těchto testů lze následně vybrat nejvhodnější vstupní proměnné, dle požadavků obchodníka. Někteří obchodníci preferují výsledky s vyšším ziskem, ale s hrozbou potenciálně vyššího drawdownu či delších sérií ztrátových obchodů. Jiní zase preferují sice pomalejší, ale zato stabilní nárůst kapitálu, bez výrazných poklesů.

Optimalizace AOS je náročná na výpočetní výkon, a tak je pro její zkrácení vhodné vlastnit výkonnější počítač. Pokud obchodníkovi nevádí, že optimalizace zabere hodně času (řádově to může být i několik dní), tak výkonnější hardware nepotřebuje. Hlavní důvod, proč je optimalizace tolik náročná je to, že počet možných scénářů, které lze otestovat roste exponenciálně s každou další konfigurovatelnou proměnnou systému.

Druhou možností je pronajmout si výpočetní výkon a nechat optimalizaci běžet na serveru u některého z poskytovatelů těchto služeb.

In sample testování

In sample testování je testování systému na stejných datech, na kterých je systém optimalizován. Toto testování má nejmenší vypovídací schopnost, protože provedením optimalizace na stejných datech máme systém přesně přizpůsoben na tato data. Na jiných datech poté systém nemusí vůbec fungovat tak, jak je od něj očekáváno.

Hrozbou je zde také tzv. curvefitting. Jedná se o stav, kdy máme systém optimalizovaný přesně na vzorek dat, která jsme pro optimalizaci využili. Na tomto konkrétním vzorku dat systém vykazuje velmi dobrých výsledků, ale na jiných datech již tak ziskový není, nebo je dokonce ztrátový.

Out of sample testování

Jedná se o pokročilejší testování, kdy si testovaná data rozdělíme na dvě části. První část si necháme pro optimalizaci a druhou část poté pro samotné otestování výsledků optimalizace. Díky tomu se dá omezit curvefitting a u systému je mnohem vyšší pravděpodobnost, že bude spolehlivě fungovat i do budoucna.

Walk forward testování

Velmi dobrým způsobem optimalizace je walk forward optimalizace. Ta rozdělí historická data na více segmentů určených pro optimalizaci (in-sample data) a pro backtesting (out-of-sample data).

Mezi hlavní přednosti této metody patří to, že máme více po sobě následujících výsledků testování pro každé nastavení systému a tím získáváme mnohem ucelenější přehled o jeho spolehlivosti a stabilitě. Pokud systém dosahuje dobrých výsledků při walk forward optimalizaci, tak je i vysoká pravděpodobnost, že bude dosahovat dobrých výsledků i při ostrém běhu.

Walk forward testování se dá rozdělit na plovoucí, kdy se testované úseky postupně v čase posouvají a ukotvené, kdy se posouvají konce testovaných úseků, ale úseky mají stejný začátek (12). Rozdělení dat podle druhu walk forward testování je znázorněno v následující tabulce.

Tabulka 1: Rozdíl mezi plovoucím a ukotveným testováním. Zdroj: vlastní tvorba

Data	
Vzorek	Plovoucí testování
1	in-sample OOS
2	in-sample OOS
3	in-sample OOS
4	in-sample OOS
Vzorek	Ukotvené testování
1	in-sample OOS
2	in-sample OOS
3	in-sample OOS
4	in-sample OOS

Reoptimalizace

Reoptimalizace (4 str. 226) je proces, při kterém systém za jeho běhu průběžně znovu testujeme a optimalizujeme, aby se nesnižovala jeho výkonnost. Toto je potřeba proto, že charakter trhů se neustále vyvíjí a mění, a proto systém spolehlivě může fungovat většínou pouze po omezenou dobu a poté jeho výkonnost klesá.

Reoptimalizaci je také vhodné provést po větší změně na trhu, kdy například skončí nějaký dlouhodobý trend, či na trhu nastane výrazná změna volatility.

4.5. Běh automatického obchodního systému

Pro správnou funkčnost obchodního systému je důležité, aby byl neustále spuštěný a mohl tak neustále vyhodnocovat pohyby trhu a reagovat na ně. Obchodníkovi se tak nabízejí tři základní možnosti, kde může svůj systém nechat spuštěný:

- Běh systému u obchodníka
- Běh systému ve výpočetním centru
- Běh systému u brokera

4.5.1. Běh systému u obchodníka

Běh systému u obchodníka je pro obchodníka nejvíce rizikový, ale zároveň také finančně velmi málo nákladný. Je vhodný především pro začínající obchodníky, kteří mají malé účty a provádí menší a méně časté obchody. Z důvodu snížení rizikovosti není vhodné využívat pro tyto účely běžný domácí počítač. Pokud by obchodník nechal systém běžet na takovém počítači, tak zde je riziko toho, že systém spadne vinou nějakého uživatelského programu, nebo může dokonce být omylem ukončen, restartován, odpojen ze zásuvky apod.

Dobře navržená strategie sice chrání každý obchod pomocí stop lossu, i přesto však kvůli pádu systému může dojít k tomu, že stop loss nebude posunut a potenciálně ziskový obchod skončí se ztrátou. Dalším problémem je ušlý zisk za dobu, kdy systém neběží a nemůže zadávat obchody.

Z toho důvodu je vhodné, aby obchodník nechal automatický obchodní systém běžet na dedikovaném stroji, na kterém nic jiného není spuštěné. Dalším prvkem snižujícím riziko je, pokud obchodník počítač připojí do záložního zdroje UPS a tím se částečně chrání před výpadky elektřiny. I přesto se však neochrání před výpadky internetu, které mohou nastat buď při výpadku elektřiny, nebo z jiných důvodů.

4.5.2. Běh systému ve výpočetním centru

Nejlepším řešením tedy pro obchodníka je, pokud si u některého z poskytovatelů serverových řešení pronajme server. Tyto servery bývají umístěny v klimatizovaných serverovnách, často přímo na páteřních internetových linkách. To jim zajišťuje jak stabilní a zálohované internetové připojení, tak zálohované dodávky elektrické energie a dodavatelé takovýchto řešení jsou schopni zaručit velmi vysoký uptime.

Obchodník tak je chráněn před rizikem výpadku elektřiny, výpadku internetu a je zde i snížena pravděpodobnost nechtěného přerušení běhu AOS.

Na tento vzdálený server se obchodník připojí pomocí internetového připojení ze svého domácího počítače. Pokud tedy u obchodníka vypadne elektřina či internetové připojení, tak se obchodník sice nepřipojí ke svému obchodnímu systému, má ale zajištěno, že systém nadále bezpečně funguje.

Finančně je tato možnost o něco nákladnější, než když systém běží na lokálním počítači. Když se však započítají náklady na elektřinu, tak spolu s klesajícími cenami za pronájem serveru se jedná o rozumnou alternativu, především pokud obchodník obchoduje větší obchody a s větším účtem.

4.5.3. Běh systému u brokera

Tato možnost je podobně bezpečná jako běh ve výpočetním centru. Rozdíl je v tom, že místo toho, aby si obchodník nakonfiguroval vlastní server a na něm platformu, kterou využívá, tak AOS nahraje na server brokera a systém běží tam. Výhodou takového přístupu je lepší dostupnost takového systému, protože brokeři ke kontrole a ovládní systému často nabízí webové rozhraní či mobilní aplikace.

5. Vlastní návrh řešení

Tato kapitola popisuje postup při návrhu obchodní strategie, proces její implementace, s vybranými důležitými částmi zdrojového kódu, proces testování strategie a její následné optimalizace.

5.1. Návrh vlastní strategie

Navrhovaná strategie má několik základních požadavků. Jedná se o to, aby AOS obchodoval pozičně a obchody vydržely otevřené delší dobu. To především proto, že cílem navrhovaného systému je dlouhodobé zhodnocení vložených prostředků tak, aby uživatel této strategie nemusel obchodování věnovat příliš času. Pokud tedy AOS vygeneruje jen pár obchodů měsíčně, tak může stále dosáhnout poměrně zajímavého zhodnocení a kontrola uskutečněných obchodů zabere oproti systémům s desítkami obchodů denně velmi málo času.

5.1.1. Výběr indikátorů

Pro implementaci automatického obchodního systému bylo zvaženo využití indikátorů zmíněných v teoretické části této práce. Každý indikátor byl zkoumán z následujících hledisek:

- Jak indikátor reaguje na změny kurzu
- Jaké signály indikátor generuje
- Jak se indikátor doplňuje s ostatními indikátory.

Podle výsledků tohoto výzkumu byly pro implementaci ve výsledné strategii vybrány indikátory exponenciální klouzavý průměr, vážený klouzavý průměr a Parabolic SAR.

Indikátor vážený klouzavý průměr byl vybrán pro určení dlouhodobého trendu. Ten je v této strategii využit k tomu, aby obchody byly prováděny vždy ve směru tohoto dlouhodobého trendu a ne proti. To totiž zvyšuje pravděpodobnost, že trh půjde požadovaným směrem a obchody skončí se ziskem. Vážený klouzavý průměr oproti jiným klouzavým průměrům byl vybrán proto, že na situaci na trhu reaguje rychleji než obyčejný klouzavý průměr, zároveň je ale dostatečně stabilní.

Indikátor Parabolic SAR je využit ke generování signálů pro vstup do trhu a pro posun hranice stop loss. Signál vstupu do trhu nastává v momentě, kdy se hodnota indikátoru

Parabolic SAR překloupí z jedné strany na druhou. Podle hodnoty indikátoru je pak nastaven stop loss vytvořeného obchodu.

Protože indikátor Parabolic SAR generuje signály stále, a tedy i v době, kdy na trhu není významný trend, tak je potřeba tyto signály nějak filtrovat, jinak by mohlo vzniknout velké množství ztrátových obchodů. Toho je dosaženo využitím dalšího exponenciálního klouzavého průměru. Ten má kratší periodu a reaguje tak rychleji než vážený klouzavý průměr. Signály pro vstup do trhu vzniklé indikátorem Parabolic SAR jsou tedy považovány za validní pouze v případě, že rychlejší klouzavý průměr předběhl ve směru trendu klouzavý průměr pomalejší. Při vzestupném trendu tedy musí být exponenciální průměr nad lineárním, a při sestupném obráceně.

5.1.2. Výběr měnového páru a časového rámce

Pro běh obchodního systému byl zvolen měnový pár EURUSD. Důvodem pro to je dobrá likvidita a nízké spready.

Obchodní systém má obchodovat pouze několikrát týdně, proto něj stačí některý z vyšších časových rámců. Z toho důvodu byl vybrán časový rámec 15 minut. Ten se může zdát pro vícedenní obchodování zbytečně jemný, výhodou ale je, že nabízí podrobnější rozlišení pohybu ceny.

5.1.3. Pravidla pro otevření obchodu

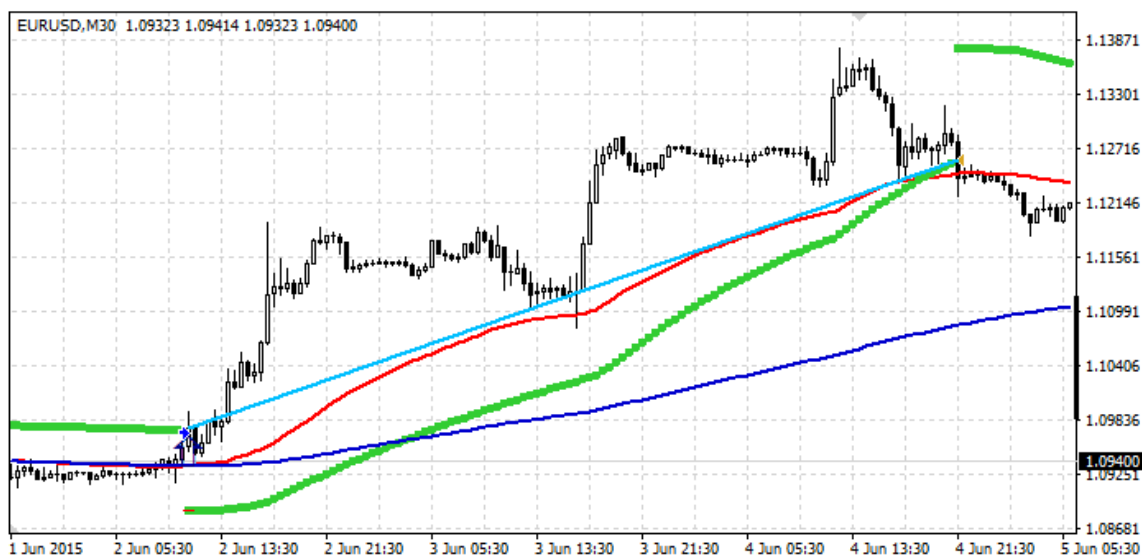
Pravidla nutná pro vstup do pozice jsou shrnuta v následující tabulce. *Long MA* značí pomalý klouzavý průměr, *Short MA* značí rychlý klouzavý průměr:

Tabulka 2: Pravidla pro vstup do obchodu. Zdroj: vlastní tvorba

Pravidla pro vstup do pozice	
Dlouhá pozice (nákup)	Krátká pozice (prodej)
<p>Long MA roste</p> <p>hodnota Long MA < hodnota Short MA</p> <p>Došlo k překlopení Parabolic SAR</p> <p>Parabolic SAR > aktuální cena</p>	<p>Long MA klesá</p> <p>hodnota Long MA > hodnota Short MA</p> <p>Došlo k překlopení Parabolic SAR</p> <p>Parabolic SAR < aktuální cena</p>

Jak spolu jednotlivé indikátory fungují a doplňují se je vidět na následujícím obrázku. Indikátor Parabolic SAR je na obrázku vyznačen splývajícími zelenými tečkami, pomalý klouzavý průměr je vyznačen tmavší modrou barvou, rychlý klouzavý průměr červenou barvou a uskutečněný obchod je vyznačen světle modrou barvou.

Je zde názorně vidět, že na začátku se trh pohyboval pouze bočním směrem. Poté nabral vzestupný trend, rychlejší klouzavý průměr začal předbíhat klouzavý průměr pomalejší a došlo k překlopení indikátoru Parabolic SAR. V ten moment byl zadán příkaz pro nákup, který trval po celou dobu vzestupného trendu, než byl obchod o dva dny později ukončen protnutím hranice stop loss. V tu chvíli již byl trend vyčerpán.



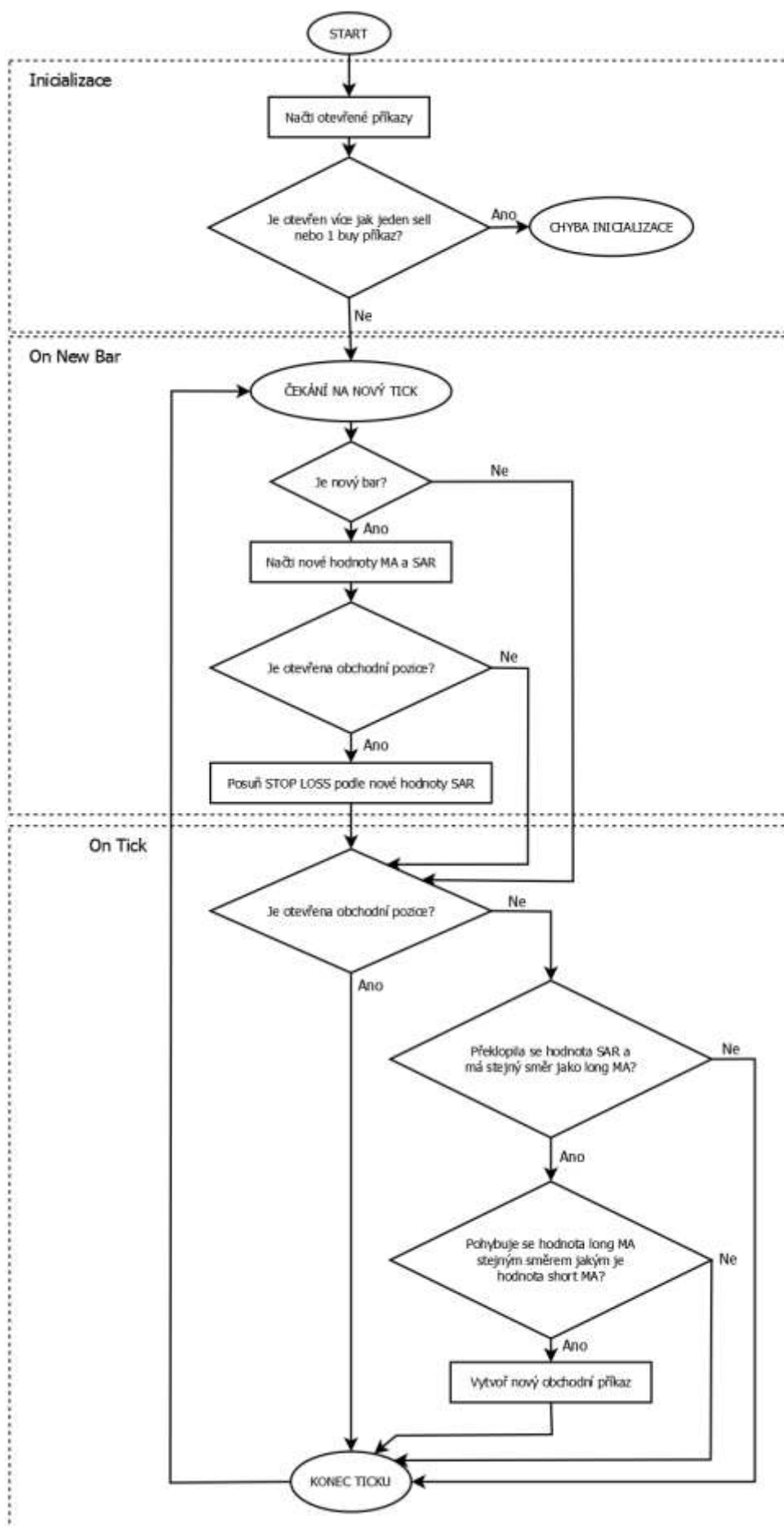
Obrázek 18: Ukázka obchodu uskutečněného obchodu. Zdroj: vlastní tvorba

5.2. Implementace vybrané strategie

Pro implementaci bylo zvoleno prostředí MetaTrader 4. Důvody pro to byly především ty, že je k němu velké množství dostupných studijních materiálů, je snadné otevřít si u nějakého brokera demo účet pro obchodování a platforma je velmi rozšířená a s velkou uživatelskou základnou.

Navrhnutá strategie se pro implementaci dá rozdělit na několik částí. Na začátku se obchodní systém musí inicializovat. Poté běží hlavní smyčka programu provádějící výpočty a obchody. Ta pak ke své funkčnosti využívá některé doplňující funkce. Hlavní smyčka programu má dvě části, a to metodu *OnTick*, která je volána při každém obchodním ticku a funkci *OnNewBar*, která je volána při přehoupnutí obchodu do nového časového rámce.

Algoritmus navržené strategie je znázorněn na vývojovém diagramu na následující stránce. Jsou zde zvýrazněny části inicializace, událostí vykonaných při přechodu do nového časového rámce a událostí vykonaných při každém ticku.



Obrázek 19: Vývojový diagram obchodního systému. Zdroj: vlastní tvorba

5.2.1. Inicializace obchodního systému

Při spuštění se obchodní systém musí inicializovat, tedy musí si nastavit potřebné parametry, se kterými pracuje a ověřit si, že vůbec může správně obchodovat.

Naimplementovaný obchodní systém si při svém spuštění ověří, že jsou zadány správné vstupní parametry. Pokud by byly zadány nesprávně, tak strategie nemusí vůbec fungovat nebo by dokonce mohla způsobit nepředvídatelné ztráty.

Následně se AOS podívá, zda je otevřena maximálně jedna krátká či jedna dlouhá pozice. Toto se děje z toho důvodu, že strategie ke své funkčnosti pracuje vždy jen s jedním obchodem. Trh totiž buď roste, anebo klesá, ale dva obchody otevřené proti sobě si pouze navzájem snižují zisk.

Pokud je nějaký obchod otevřený, tak si ho strategie načte a pracuje s ním. Tato situace může nastat v případě, kdy byl AOS ukončen, ale zrovna měl otevřenou nějakou pozici.

Pokud je toto všechno splněno, tak je systém správně načten a hlavnímu programu vrátí návratovou hodnotu `INIT_SUCCEEDED`.

5.2.2. Hlavní smyčka programu.

Hlavní smyčka programu má na starosti samotné vytváření obchodů, jejich kontrolu a posouvání stop lossu. To vše se provádí ve funkci `OnTick`, kterou platforma MetaTrader volá při každém novém ticku na trhu. Protože některé výpočty není potřeba provádět s každým tickem, tak je na začátku funkce `OnTick` přidána funkce `OnNewBar`. Ta je zavolána, pouze když je dosaženo nového časového rámce a naroste počítadlo časových rámců v grafu.

Ve funkci `OnNewBar` jsou výpočty, jejichž aktualizace stačí udělat jednou za daný časový rámeček. Jedná se o:

- Výpočet nové hodnoty indikátoru Parabolic SAR
- Posunutí hladiny stop lossu podle ukazatele Parabolic SAR
- Uložení historie dlouhého klouzavého průměru, pro určení vzestupného či sestupného trendu
- Výpočet nové hodnoty dlouhého klouzavého průměru

Funkce `OnTick` se má, kromě volání funkce `OnNewBar`, na starosti také vyhodnocení podmínek pro vstup do dlouhé nebo krátké pozice a při splnění podmínek otevírá nové obchody.

5.2.3. Doplnkové funkce obchodního systému

Obchodní systém ke své funkčnosti využívá několik doplňkových funkcí.

První z nich je funkce `GetTradeVolume`. Ta slouží k určení správné velikosti pozice. Pro výpočet používá počáteční a současnou hodnotu finančních prostředků na účtu obchodníka a konstantu určující počáteční velikost obchodovaného lotu. Pokud je obchodník v zisku a jeho kapitál narůstá, tak jsou otevírané větší obchodní pozice. Naopak pokud by systém byl ztrátový, tak otevírá menší obchody ve snaze snížit případné další ztráty. Funkce vypadá následovně:

```
double GetTradeVolume()
{
    double available_margin = AccountFreeMargin();
    double ratio = available_margin / initial_margin;
    double min_lot = MarketInfo(Symbol(), MODE_MINLOT);
    double lot = MathMax(min_lot, initial_lot *ratio);
    return lot;
}
```

Další pomocnou funkcí je funkce `CreateObject` pro zakreslení objektu do grafu. Obchodní systém tak může zakreslovat značky v okamžiku, kdy učinil nějaké rozhodnutí. To pomáhá se zpětnou kontrolou systému a jím uskutečněných obchodů.

Platforma `MetaTrader` bohužel neumožňuje využívání skutečných `break pointů`, při testování strategie. Z toho důvodu byla ve fázi testování a ladění obchodního systému, velmi užitečná funkce `BreakPoint`. Funkce dělá to, že pokud je systém spuštěn v testeru strategií a funkce je zavolána, tak je testování pozastaveno. To sice neumožňuje obchodníkovi nahlédnout do hodnot uložených v proměnných programu, jako by to umožnil skutečný `break point`, umožňuje ale zastavit tester a odladit některé méně časté situace, pro které je potřeba přesně odchytil okamžik trhu kdy nastanou. Funkce využívá metody `keybd_event` z hlavičkového souboru `WinUser32.mhq` pro zaslání události *stisknutí klávesy Pause*. Ta se v programu `MetaTrader` používá pro pozastavení testování.

Posledních pár funkcí ve zdrojovém souboru má na starosti sbírání statistik o provedených obchodech a jejich tisk při ukončení obchodního systému. Díky těmto dodatečným datům je možné sledovat, které dny v týdnu byly pro obchodníka nejziskovější, a které byly naopak nejvíce ztrátové.

5.3. Testování a optimalizace strategie

Testování a optimalizace systému probíhalo v několika fázích. Nejprve byla ověřena správnost implementace systému a daných pravidel. Na to stačí základní in-sample testování. Následně byl systém optimalizován pro to, aby dosáhl co nejlepších výsledků. Testování probíhalo na testovacím účtu se základním kapitálem o velikosti 10 000 USD.

5.3.1. In sample testování

Testování metodou in-sample bylo nejvíce použito ve fázi vývoje základní funkcionality obchodního systému. V této fázi je nejdůležitější prověřit, že principy navržené ve strategii jsou do obchodního systému správně implementovány. V tuto chvíli je výkon systému podprůměrný, protože ještě neprošel fází optimalizace. Výsledky testování AOS jsou vidět níže. Hned na začátku běhu systému je zřetelný značný propad, způsobený delší sérií ztrátových obchodů.



Obrázek 20: Výsledek prvotního běhu systému. Zdroj: vlastní tvorba

5.4. Optimalizace strategie

Při procesu optimalizace byly nejprve optimalizovány jednotlivé parametry samostatně, aby se našly jejich přibližné optimální hodnoty. Následně byly všechny parametry optimalizovány dohromady, pro dosažení co nejlepších výsledků optimalizace.

Optimalizace byla provedena na časovém období 1.1.2016 – 1.12.2016 měnového páru EURUSD.

5.4.1. Optimalizace parametrů indikátoru Parabolic SAR

Nejprve byl optimalizován indikátor Parabolic SAR, který stanovuje signály pro vstup do obchodu a pro výstup z obchodu pomocí stop lossu.

Jako první z parametrů byl testován akcelerační koeficient indikátoru v rozmezí 0,001 – 0,1. Nejlepších výsledků dosahoval indikátor při nastavení akceleračního koeficientu na 0,01

Dalším testovaným parametrem byl maximální krok indikátoru Parabolic SAR. Rozmezí pro testování tohoto parametru bylo 0,01 – 0,6. Nejlepšího výsledku dosáhl tento parametr při nastavení na hodnotu 0,03. Nejlepší výsledek však byl velmi podobný výsledku z optimalizace parametru akceleračního koeficientu.



Obrázek 21: Běh systému po optimalizaci indikátoru Parabolic SAR. Zdroj: vlastní tvorba

Z výsledků běhu neoptimálnějšího řešení je vidět, že propad na začátku se podařilo eliminovat. I přesto však systém nelze zatím považovat za velmi úspěšný, protože křivka dostupného kapitálu je značně nevyrovnaná, a i zisk je zatím poměrně malý.

5.4.2. Optimalizace parametrů klouzavých průměrů

Dalším krokem optimalizace je optimalizace periody klouzavých průměrů. Pro tu bylo zvolen rozsah 50-500 s krokem 50 pro delší klouzavý průměr a 30-300 s krokem 30 pro kratší klouzavý průměr.

Samotnou optimalizaci v tomto kroku značně urychlilo ověření vstupních parametrů při spuštění obchodního systému. Systém u strategií, které měly na vstupu delší periodu

pro kratší klouzavý průměr, ihned vrátil návratovou hodnotu s kódem INIT_FAILED a celý běh se špatnými parametry tak byl přeskočen.

Výsledky optimalizace klouzavých průměrů jsou vidět na následujícím obrázku. Za ideální byly považovány průměry s periodou 400 a 120.



Obrázek 22: Běh systému po optimalizaci klouzavých průměrů. Zdroj: vlastní tvorba

Z výsledků optimalizace je vidět že průběh grafu je vyhlazenější, avšak zvýšil se i propad na konci. Dobrým ukazatelem však je, že maximální drawdown klesl z 28 % na 23 % a také stoupl vypočítaný profit factor.

5.4.3. Optimalizace všech parametrů najednou

Optimalizací všech parametrů najednou právě kolem předem nalezených hodnot bylo nakonec nalezeno optimální řešení, znázorněné na obrázku níže. Výsledné hodnoty tohoto řešení jsou ve sloupci *Value*. Počáteční hodnoty optimalizovaných parametrů jsou ve sloupci *Start*, konečné ve sloupci *Stop* a velikost kroku je ve sloupci *Step*.

Variable	Value	Start	Step	Stop
<input type="checkbox"/> log	false	false		true
<input checked="" type="checkbox"/> sar_step	0.0006	0.0002	0.0004	0.001
<input checked="" type="checkbox"/> sar_max	0.01	0.005	0.005	0.01
<input checked="" type="checkbox"/> long_ma_period	400	380	20	420
<input checked="" type="checkbox"/> short_ma_period	120	110	10	130
<input type="checkbox"/> long_ma_method	Linear weighted	Simple		Linear weighted
<input type="checkbox"/> short_ma_method	Exponential	Simple		Linear weighted
<input type="checkbox"/> long_price	Close price	Close price		Weighted price
<input type="checkbox"/> short_price	Close price	Close price		Weighted price
<input type="checkbox"/> initial_lot	0.5	0.5	0.0	0.0

Obrázek 23: Optimalizované parametry. Zdroj: vlastní tvorba

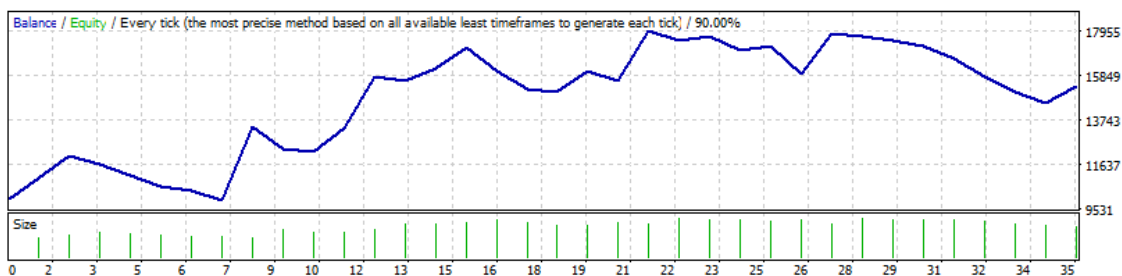
Po konečné optimalizaci dosahuje systém na daném vzorku dat ziskového faktoru 1,46, jak je vidět ve zprávě z testování níže na této stránce. Za povšimnutí stojí i spodní zelený sloupcový graf, na kterém je vidět velikost obchodované pozice. Ta se s narůstajícím kapitálem zvětšuje a tím se zrychluje tempo, jakým systém vydělává.

Strategy Tester Report

dpexpert5

FXDD-MT4 Demo Server (Build 1090)

Symbol	EURUSD (Euro vs. United States Dollar)		
Period	30 Minutes (M30) 2015.01.02 11:00 - 2016.01.13 23:30 (2015.01.01 - 2016.01.14)		
Model	Every tick (the most precise method based on all available least timeframes)		
Parameters	log=false; sar_step=0.0006; sar_max=0.01; long_ma_period=400; short_ma_period=120; initial_lot=0.5;		
Bars in test	13768	Ticks modelled	24198347 Modelling quality 90.00%
Mismatched charts errors	3		
Initial deposit	10000.00	Spread	Current (19)
Total net profit	5331.62	Gross profit	17010.76 Gross loss -11679.14
Profit factor	1.46	Expected payoff	152.33
Absolute drawdown	305.32	Maximal drawdown	6008.84 (29.61%) Relative drawdown 29.61% (6008.84)
Total trades	35	Short positions (won %)	21 (42.86%) Long positions (won %) 14 (28.57%)
		Profit trades (% of total)	13 (37.14%) Loss trades (% of total) 22 (62.86%)
		Largest profit trade	3492.18 loss trade -1268.00
		Average profit trade	1308.52 loss trade -530.87
		Maximum consecutive wins (profit in money)	2 (3535.12) consecutive losses (loss in money) 7 (-3247.88)
		Maximal consecutive profit (count of wins)	3535.12 (2) consecutive loss (count of losses) -3247.88 (7)
		Average consecutive wins	1 consecutive losses 2



Obrázek 24: Výsledek konečné optimalizace. Zdroj: vlastní tvorba

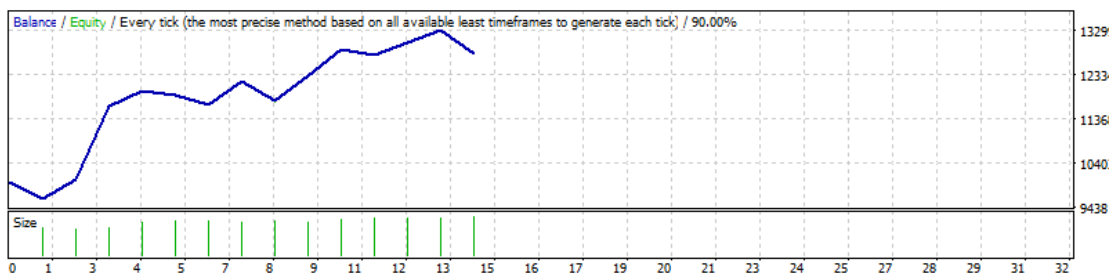
5.4.4. Out of sample test

Out of sample test zvolené strategie byl proveden za období 1.12.2016 – 29.4.2017. Toto téměř půl roku dlouhé období bylo zvoleno proto, že se nachází ještě relativně nedávno v minulosti, a tak výsledky z tohoto testu budou mít větší vypovídací hodnotu než výsledky, které by byly proveden na několik let starých datech.

Z výsledků vidíme, že i přesto, že má systém poměrně vysoký drawdown, tak jinak dosáhl dobrého výsledku.

Strategy Tester Report dpexpert5 FXDD-MT4 Demo Server (Build 1090)

Symbol	EURUSD (Euro vs. United States Dollar)				
Period	30 Minutes (M30) 2016.12.01 00:00 - 2017.04.14 23:30 (2016.12.01 - 2017.04.15)				
Model	Every tick (the most precise method based on all available least timeframes)				
Parameters	log=false; sar_step=0.004; sar_max=0.01; long_ma_period=450; short_ma_period=100; initial_lot=1;				
Bars in test	5641	Ticks modelled	9158723	Modelling quality	90.00%
Mismatched charts errors	2				
Initial deposit	10000.00	Spread	Current (17)		
Total net profit	2808.02	Gross profit	4556.16	Gross loss	-1748.14
Profit factor	2.61	Expected payoff	200.57		
Absolute drawdown	546.00	Maximal drawdown	2731.99 (19.83%)	Relative drawdown	19.83% (2731.99)
Total trades	14	Short positions (won %)	5 (60.00%)	Long positions (won %)	9 (55.56%)
		Profit trades (% of total)	8 (57.14%)	Loss trades (% of total)	6 (42.86%)
		Largest profit trade	1598.58	loss trade	-513.38
		Average profit trade	569.52	loss trade	-291.36
		Maximum consecutive wins (profit in money)	3 (2357.35)	consecutive losses (loss in money)	2 (-308.53)
		Maximal consecutive profit (count of wins)	2357.35 (3)	consecutive loss (count of losses)	-513.38 (1)
		Average consecutive wins	2	consecutive losses	1



Obrázek 25: Výsledek out of sample testu. Zdroj: vlastní tvorba

5.4.5. Optimalizace dalších konfigurovatelných nastavení

Další konfigurovatelné hodnoty, jako typ použitých klouzavých průměrů a volba, zda použít otevírací, zavírací nebo nějakou jinou cenu, neměly na proces optimalizace téměř žádný vliv. Z toho vyplývá, že tato strategie je vůči těmto parametrům neměnná a nemají na ni vliv. Níže je znázorněn výsledek optimalizace některých těchto parametrů v tabulce. Pro velké množství kombinací zde nejsou zobrazeny všechny výsledky, ale jen jejich část.

P., /	Profit	Total trades	Profit factor	Drawdown %	
1	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=3; long_price=0; short_price=2;
2	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=3; long_price=0; short_price=2;
3	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=0; long_price=0; short_price=2;
4	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=0; long_price=0; short_price=2;
5	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=1; long_price=0; short_price=2;
6	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=1; long_price=0; short_price=2;
7	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=2; long_price=0; short_price=2;
8	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=2; long_price=0; short_price=2;
9	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=3; long_price=1; short_price=2;
10	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=3; long_price=1; short_price=2;
11	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=0; long_price=1; short_price=2;
12	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=0; long_price=1; short_price=2;
13	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=1; long_price=1; short_price=2;
14	385.51	22	1.50	5.21	long_ma_method=1; short_ma_method=1; long_price=1; short_price=2;
15	385.51	22	1.50	5.21	long_ma_method=0; short_ma_method=2; long_price=1; short_price=2;

Obrázek 26: Zobrazení nulového vlivu typu klouzavého průměru a použité ceny na výsledky optimalizace. Zdroj: vlastní tvorba

5.5. Přínos navrženého systému

Přínosem této práce je vytvoření automatického obchodního systému, který zvládá sám obchodovat na měnovém trhu, a to s kontrolou podstupeného rizika díky možnosti volby velikosti obchodovaného lotu a díky provádění příkazů chráněných stop lossem. Systém obstál v out of sample testu, a tedy je zde vysoký předpoklad, že systém bude i do budoucna dále ziskový. Aby systém neztratil svou výkonnost, tak je vhodné ho pro případné nasazení dále pravidelně reoptimalizovat. V případě že by na trhu nastala nena-
dálá větší změna, která by výrazně ovlivnila charakter trhu, tak by bylo nejspíše nutné systém upravit.

6. Závěr

Cílem této práce bylo vytvořit funkční automatický obchodní systém na měnovém trhu. Tento hlavní cíl práce splnila, protože navržený systém byl i úspěšně implementován, optimalizován a otestován.

Dalšími dílčími cíli bylo dosáhnout u systému ziskovosti a snažit se omezit riziko ztráty investovaného kapitálu.

Podle testování systému se vhodným výběrem technických indikátorů, správnou volbou pravidel pro vstup a výstup z obchodu a následnou optimalizací podařilo vytvořit takový systém, který je i nadále schopen dosahovat zisku. Pro dlouhodobé udržení tohoto zisku však je důležité, aby byl obchodní systém udržován a přizpůsobován stále se měnícím podmínkám na trhu.

Pro omezení rizika obchodní systém ke každému obchodu automaticky vytváří stop loss. Dalším prvkem omezujícím riziko ztráty veškerého kapitálu kvůli několika neúspěšným obchodům je uživatelsky nastavitelná hodnota obchodovaného objemu. Ta se navíc sama dále přizpůsobuje podle toho, jak se systému daří generovat zisk.

Systém je koncipován pro uskutečňování malého množství déle trvajících obchodů. Tento základní požadavek byl takto stanoven proto, že systém byl zamýšlen jako alternativa k různým spořicími produktům, které nabízejí komerční společnosti. Dle výsledků hodnocení, i ze samotné podstaty trhu, kde nevíme, jak se cena bude pohybovat, se tak jedná o alternativu spíše rizikovější.

Další rozvoj systému si dovedu představit v takové úpravě, aby systém obchodoval častěji, a to i ty trendy, které trvají kratší dobu a které nyní systém nezachytí.

7. Seznam použitých zdrojů

1. **ŠTÝBR, David, Petr KLEPETKO a Pavlína ONDRÁČKOVÁ.** *Začínáme investovat a obchodovat na kapitálových trzích.* Praha : Grada, 2011. ISBN 9788024736488.
2. **Ing. Peter Badura, PhD.** Bid, Ask, Spread - vysvětlení pojmů. *eTrading.sk.* [Online] Ekonomická univerzita v Bratislavě. [Citace: 12. 04 2017.] <https://etrading.sk/cz/forex-trading-obchodovani/728-bid-ask-spread-sell-buy-price>.
3. **Řízení rizika na Forexu.** *Admiral Markets.* [Online] Admiral Markets (UK) Ltd. <https://admiralmarkets.cz/education/articles/forex-basics/rizeni-rizika-na-forexu>.
4. **VESELÁ, Jitka a Martin OLIVA.** *Technická analýza na akciových, měnových a komoditních trzích.* Praha : Ekopress, 2015. ISBN 9788087865224.
5. **Milton, Adam.** Simple, Exponential and Weighted Moving Averages. *The Balance.* [Online] 06 2016. <https://www.thebalance.com/simple-exponential-and-weighted-moving-averages-1031196>.
6. **MACD (Moving Average Convergence/Divergence Oscillator) .** *StockCharts.* [Online] StockCharts.com, Inc., 2016. http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:moving_average_convergence_divergence_macd.
7. **Klimo, Ján.** Parabolic SAR: Inteligentní ochrana vašich zisků. *Investujeme.cz.* [Online] 04 2008. <https://www.investujeme.cz/clanky/parabolic-sar-inteligentni-ochrana-vasich-zisku/>.
8. **Platform for forex trading.** *MetaTrader 4.* [Online] 13. 04 2017. <https://www.metatrader4.com/>.
9. **STP Trading Platform.** *The New Standard in FX Trading.* [Online] Spotware Systems Ltd., 2017. [Citace: 21. 04 2017.] <https://www.spotware.com/products/client-side-applications/ctrader>.
10. **Trading Software & Brokerage.** *NinjaTrader.* [Online] NinjaTrader, LLC, 2017. <http://ninjatrader.com/>.
11. **Quantopian.** [Online] **Quantopian, 2017.** <https://www.quantopian.com/home>.
12. **Tomáš.** WalkForward analýza podrobněji. *Financnik.cz.* [Online] 2017. http://www.financnik.cz/komodity/fin_home/walkforward-analyza-podrobneji.html.
13. **GRAHAM, B.** *Inteligentní investor.* Praha : GRADA, 2007. 504 s. ISBN: 978-80-247-1792-0.

14. REJNUŠ, O. *Finanční trhy. 1. vyd.* Ostrava : KEY Publishing, 2008. ISBN: 978-80-87-8.

8. Seznam obrázků

Obrázek 1: Cena některých významných indexů a její vývoj. Zdroj: https://finance.yahoo.com/world-indices	13
Obrázek 2: Forex obchodní hodiny. Zdroj: https://www.investing.com/tools/market-hours	14
Obrázek 3: Rozdíl mezi sell a buy. Zdroj: Program MetaTrader 4, vlastní tvorba	15
Obrázek 4: Hodinový sloupkový graf EURUSD s ukazatelem aktuální ceny.	21
Obrázek 5: Hodinový sloupkový graf EURNZD.	21
Obrázek 6: Denní svíčkový graf na měnovém páru GBPJPY	22
Obrázek 7: Rozdíl sloupců u sloupkového a svíčkového grafu. Zdroj: vlastní tvorba...	22
Obrázek 8: Candlevolume chart a Point and figure chart.....	23
Obrázek 9: Ukázka klouzavých průměrů. Zdroj: vlastní tvorba	25
Obrázek 10: Indikátor MACD. Zdroj: vlastní tvorba	27
Obrázek 11: Ukázka bollingerových pásem. Zdroj: vlastní tvorba	28
Obrázek 12: Ukázka indikátoru Parabolic SAR. Zdroj: vlastní tvorba	29
Obrázek 13: Ukázka indikátoru RSI. Zdroj: vlastní tvorba.....	30
Obrázek 14: Indikátor stochastic. Zdroj: vlastní tvorba	31
Obrázek 15: Rozhraní programu MetaTrader 4. Zdroj: vlastní tvorba.....	34
Obrázek 16: Ukázka nabídky obchodu s automatickými obchodními systémy. Zdroj: https://www.mql5.com/en/market/mt4/expert	37
Obrázek 17: Ukázka programovacího prostředí platformy MetaTrader 4. Zdroj: vlastní tvorba	40
Obrázek 18: Ukázka obchodu uskutečněného obchodu. Zdroj: vlastní tvorba	47
Obrázek 19: Vývojový diagram obchodního systému. Zdroj: vlastní tvorba.....	48
Obrázek 20: Výsledek prvotního běhu systému. Zdroj: vlastní tvorba	51
Obrázek 21: Běh systému po optimalizaci indikátoru Parabolic SAR. Zdroj: vlastní tvorba	52
Obrázek 22: Běh systému po optimalizaci klouzavých průměrů. Zdroj: vlastní tvorba	53
Obrázek 23: Optimalizované parametry. Zdroj: vlastní tvorba.....	54
Obrázek 24: Výsledek konečné optimalizace. Zdroj: vlastní tvorba.....	54
Obrázek 25: Výsledek out of sample testu. Zdroj: vlastní tvorba	55
Obrázek 26: Zobrazení nulového vlivu typu klouzavého průměru a použité ceny na výsledky optimalizace. Zdroj: vlastní tvorba	56

9. Seznam tabulek

Tabulka 1: Rozdíl mezi plavoucím a ukotveným testováním. Zdroj: vlastní tvorba
.....**Error! Bookmark not defined.**

Tabulka 2: Pravidla pro vstup do obchodu. Zdroj: vlastní tvorba**Error! Bookmark not defined.**

10. Příloha 1: Zdrojový kód automatického obchodního systému

```
//+-----  
//|           dpexpert.mq4  
//|           Copyright 2017, Vaclav Sauer  
//+-----  
  
#include <WinUser32.mqh>  
  
#property copyright "Copyright 2017, Vaclav Sauer"  
#property strict  
  
//----- GLOBAL VARIABLES -----  
  
// ID of current buy ticket  
int buy_order_ticket = -1;  
int sell_order_ticket = -1;  
  
// ID of current sell ticket  
int order_day_of_week = 0;  
double order_stop_loss = 0;  
  
// Drawed arrows count to generate unique names for them  
int arrow_count = 0;  
  
// Count of bars to detect new bar  
int last_bars = -1;  
  
// Store initial account margin to calculate lot size  
double initial_margin;  
  
// Store values of used indicators  
double mva_hist[2] = { -1, -1 };  
bool sar_up = false;  
bool mva_up = false;  
  
//----- INPUT SETTINGS -----  
  
// Enable/disable logging  
input bool log = false;  
  
// Parabolic SAR setup  
input double sar_step = 0.002;  
input double sar_max = 0.05;  
  
// Moving averages periods  
input int long_ma_period = 350;  
input int short_ma_period = 70;  
  
// Moving average methods  
input ENUM_MA_METHOD long_ma_method = MODE_LWMA;  
input ENUM_MA_METHOD short_ma_method = MODE_EMA;  
  
// Moving averages price  
input ENUM_APPLIED_PRICE long_price = PRICE_CLOSE;  
input ENUM_APPLIED_PRICE short_price = PRICE_CLOSE;  
  
//Initial lot size used for trade volume calculation  
input double initial_lot = 0.5;
```

```

// Number of winning/loosing trades per day
double mo_profit = 0;
double mo_loss = 0;
double tu_profit = 0;
double tu_loss = 0;
double we_profit = 0;
double we_loss = 0;
double th_profit = 0;
double th_loss = 0;
double fr_profit = 0;
double fr_loss = 0;
double sa_profit = 0;
double sa_loss = 0;
double su_profit = 0;
double su_loss = 0;

// Profit/loss of trades per day
int mo_trades_profit = 0;
int mo_trades_loss = 0;
int tu_trades_profit = 0;
int tu_trades_loss = 0;
int we_trades_profit = 0;
int we_trades_loss = 0;
int th_trades_profit = 0;
int th_trades_loss = 0;
int fr_trades_profit = 0;
int fr_trades_loss = 0;
int sa_trades_profit = 0;
int sa_trades_loss = 0;
int su_trades_profit = 0;
int su_trades_loss = 0;

//----- INITIALIZATION -----
int OnInit()
{
    Print("----- INIT -----");

    // Cant work if slower MA is faster then faster
    if(long_ma_period <= short_ma_period)
        return INIT_FAILED;

    // Print some useful info
    Print("We have ", OrdersTotal(), " opened orders.");

    // We don't know ho to work with more than 2 orders
    if (OrdersTotal() > 2) {
        Alert("Too many orders, can't continue.");
        return (INIT_FAILED);
    }

    // Load opened orders
    for (int i = 0; i < OrdersTotal(); i++) {
        if (OrderSelect(i, SELECT_BY_POS, MODE_TRADES) == false) {
            Print("Can't get order history. :", GetLastError());
            continue;
        }
        Print("Order ticket: ", OrderTicket(), ", Order Type: ", OrderType(),
            ", Open Price: ", OrderOpenPrice(), ", Profit: ", OrderProfit(),
            ", Stop Loss: ", OrderStopLoss(), ", Symbol: ", OrderSymbol(),
            ", Lots: ", OrderLots());

        if (OrderType() == OP_BUY) {
            // Check that we have only one buy ticket

```

```

        if (buy_order_ticket != -1)
            return (INIT_FAILED);
        // Store ticket
        buy_order_ticket = OrderTicket();
        Print("Buy order: ", buy_order_ticket);
    }
    else if (OrderType() == OP_SELL) {
        // Check that we have only one sell ticket
        if (sell_order_ticket != -1)
            return (INIT_FAILED);
        // Store ticket
        sell_order_ticket = OrderTicket();
        Print("Sell order: ", sell_order_ticket);
    }
    else {
        Alert("Unknown order, can't continue.");
        return (INIT_FAILED);
    }
}

// Store initial margin
initial_margin = AccountFreeMargin();

// Logging
if(log)
Print("|OrderProfit|Profitable|order_day_of_week|OrderOpenTime|OrderSwap|OrderSto
pLoss|order_stop_loss");

    return (INIT_SUCCEEDED);
}

//----- INDICATOR CALLS -----

// Get current value of long Moving Average
double GetLongMaValue()
{
    return iMA(NULL, PERIOD_CURRENT, long_ma_period, 0, MODE_LWMA, PRICE_OPEN, 0);
}

// Get current value of short Moving Average
double GetShortMaValue()
{
    return iMA(NULL, PERIOD_CURRENT, short_ma_period, 0, MODE_EMA, PRICE_OPEN, 0);
}

// Get current value of Parabolic SAR
double GetSarValue()
{
    return iSAR(NULL, PERIOD_CURRENT, sar_step, sar_max, 0);
}

```



```

//----- MAIN AOS FUNCTIONALITY -----
// This function is called each tick
void OnTick()
{
    // Detect start of new bar
    if (last_bars < Bars) {
        last_bars = Bars;
        OnNewBar();
    }

    double sar = GetSarValue();

    if (sar < Close[0] && sar_up == true && buy_order_ticket == -1) {
        double long_ma = GetLongMaValue();
        if (mva_hist[1] < mva_hist[0]) {
            if (GetShortMaValue() > mva_hist[0]) {
                CreateObject(Ask, OBJ_ARROW_UP, true, 3, clrLightPink);
                double volume = GetTradeVolume();
                buy_order_ticket = OrderSend(Symbol(), OP_BUY, volume, Ask, 3, sar,0);
                order_day_of_week = DayOfWeek();
                order_stop_loss = sar - Ask;
            }
        }
    }
    if (sar > Close[0] && sar_up == false && sell_order_ticket == -1) {
        double long_ma = GetLongMaValue();
        if (mva_hist[1] > mva_hist[0]) {
            if (GetShortMaValue() < mva_hist[0]) {
                CreateObject(Bid, OBJ_ARROW_DOWN, false, 3, clrLightSkyBlue);
                double volume = GetTradeVolume();
                sell_order_ticket = OrderSend(Symbol(), OP_SELL, volume, Bid, 3, sar, 0);
                order_day_of_week = DayOfWeek();
                order_stop_loss = Bid - sar;
            }
        }
    }
    sar_up = sar > Close[0];
}

// Called when new bar is creted
void OnNewBar()
{
    // Shift Moving Average history
    mva_hist[1] = mva_hist[0];
    mva_hist[0] = GetLongMaValue();

    // Get new SAR value
    double sar = GetSarValue();

    // Move buy ticket stop loss
    if (buy_order_ticket != -1 && OrderSelect(buy_order_ticket, SELECT_BY_TICKET)) {
        // Clear closed order
        if (OrderCloseTime() != 0) {
            LogTrade();
            buy_order_ticket = -1;
        }
        else {
            // Move stop loss
            double stop_loss = sar;
            if (OrderStopLoss() < stop_loss) {
                bool res = OrderModify(
                    buy_order_ticket, OrderOpenPrice(), stop_loss, 0, 0, Blue);
                if (!res) {

```

```

        Print("Error in OrderModify. Error code = ", GetLastError());
    }
}

// Move sell ticket stop loss
if (sell_order_ticket != -1 && OrderSelect(sell_order_ticket, SELECT_BY_TICKET)) {
    // Clear closed order
    if (OrderCloseTime() != 0) {
        LogTrade();
        sell_order_ticket = -1;
    }
    else {
        // Move stop loss
        double stop_loss = sar;
        if (OrderStopLoss() > stop_loss) {
            bool res = OrderModify(
                sell_order_ticket, OrderOpenPrice(), stop_loss, 0, 0, Blue);
            if (!res) {
                Print("Error in OrderModify. Error code = ", GetLastError());
            }
        }
    }
}

}

double GetTradeVolume()
{
    double available_margin = AccountFreeMargin();
    double ratio = available_margin / initial_margin;
    double min_lot = MarketInfo(Symbol(), MODE_MINLOT);
    double lot = MathMax(min_lot, initial_lot *ratio);
    return lot;
}

//----- HELPER METHODS -----

// Draws an object into graph
void CreateObject(double price, ENUM_OBJECT arrow_type, bool anchor_top, int
width, color item_color)
{
    string arrow_name = "Arrow_" + string(arrow_count);
    arrow_count++;
    ObjectCreate(ChartID(), arrow_name, arrow_type, 0, TimeCurrent(), price);
    ObjectSetInteger(ChartID(), arrow_name, OBJPROP_WIDTH, width);
    ObjectSetInteger(ChartID(), arrow_name, OBJPROP_COLOR, item_color);
    ObjectSetInteger(ChartID(), arrow_name, OBJPROP_ANCHOR, anchor_top ?
ANCHOR_TOP : ANCHOR_BOTTOM);
}

// Pauses the execution when running in strategy tester by pressing pause
void BreakPoint()
{
    // Return if we are not in strategy tester with enabled visual mode
    if (!IsVisualMode())
        return;

    // Press and release Pause button, to pause test execution
    keybd_event(19, 0, 0, 0);
    Sleep(10);
    keybd_event(19, 0, 2, 0);
}

```

```

//----- LOGGING -----
// Log some useful information
void LogTrade()
{
    if(log == false)
        return;

    Print("|", OrderProfit(), "|", (OrderProfit() > 0) ? 1 : 0,
          "|", order_day_of_week, "|", OrderOpenTime(),
          "|", OrderSwap(), "|", OrderStopLoss(), "|", order_stop_loss);

    // Log profit/loss count
    mo_trades_profit += (OrderProfit() > 0 && order_day_of_week == 1) ? 1 : 0;
    mo_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 1) ? 1 : 0;
    tu_trades_profit += (OrderProfit() > 0 && order_day_of_week == 2) ? 1 : 0;
    tu_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 2) ? 1 : 0;
    we_trades_profit += (OrderProfit() > 0 && order_day_of_week == 3) ? 1 : 0;
    we_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 3) ? 1 : 0;
    th_trades_profit += (OrderProfit() > 0 && order_day_of_week == 4) ? 1 : 0;
    th_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 4) ? 1 : 0;
    fr_trades_profit += (OrderProfit() > 0 && order_day_of_week == 5) ? 1 : 0;
    fr_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 5) ? 1 : 0;
    sa_trades_profit += (OrderProfit() > 0 && order_day_of_week == 6) ? 1 : 0;
    sa_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 6) ? 1 : 0;
    su_trades_profit += (OrderProfit() > 0 && order_day_of_week == 0) ? 1 : 0;
    su_trades_loss  += (OrderProfit() < 0 && order_day_of_week == 0) ? 1 : 0;

    // Log profit/loss sum
    mo_profit += (OrderProfit() > 0 && order_day_of_week == 1) ? OrderProfit() : 0;
    mo_loss  += (OrderProfit() < 0 && order_day_of_week == 1) ? OrderProfit() : 0;
    tu_profit += (OrderProfit() > 0 && order_day_of_week == 2) ? OrderProfit() : 0;
    tu_loss  += (OrderProfit() < 0 && order_day_of_week == 2) ? OrderProfit() : 0;
    we_profit += (OrderProfit() > 0 && order_day_of_week == 3) ? OrderProfit() : 0;
    we_loss  += (OrderProfit() < 0 && order_day_of_week == 3) ? OrderProfit() : 0;
    th_profit += (OrderProfit() > 0 && order_day_of_week == 4) ? OrderProfit() : 0;
    th_loss  += (OrderProfit() < 0 && order_day_of_week == 4) ? OrderProfit() : 0;
    fr_profit += (OrderProfit() > 0 && order_day_of_week == 5) ? OrderProfit() : 0;
    fr_loss  += (OrderProfit() < 0 && order_day_of_week == 5) ? OrderProfit() : 0;
    sa_profit += (OrderProfit() > 0 && order_day_of_week == 6) ? OrderProfit() : 0;
    sa_loss  += (OrderProfit() < 0 && order_day_of_week == 6) ? OrderProfit() : 0;
    su_profit += (OrderProfit() > 0 && order_day_of_week == 0) ? OrderProfit() : 0;
    su_loss  += (OrderProfit() < 0 && order_day_of_week == 0) ? OrderProfit() : 0;

}

void OnDeinit(const int reason)
{
    Print("Mo+ ", mo_trades_profit, " sales, total:", mo_profit);
    Print("Mo- ", mo_trades_loss , " sales, total:", mo_loss );
    Print("Tu+ ", tu_trades_profit, " sales, total:", tu_profit);
    Print("Tu- ", tu_trades_loss , " sales, total:", tu_loss );
    Print("We+ ", we_trades_profit, " sales, total:", we_profit);
    Print("We- ", we_trades_loss , " sales, total:", we_loss );
    Print("Th+ ", th_trades_profit, " sales, total:", th_profit);
    Print("Th- ", th_trades_loss , " sales, total:", th_loss );
    Print("Fr+ ", fr_trades_profit, " sales, total:", fr_profit);
    Print("Fr- ", fr_trades_loss , " sales, total:", fr_loss );
    Print("Sa+ ", sa_trades_profit, " sales, total:", sa_profit);
    Print("Sa- ", sa_trades_loss , " sales, total:", sa_loss );
    Print("Su+ ", su_trades_profit, " sales, total:", su_profit);
    Print("Su- ", su_trades_loss , " sales, total:", su_loss );
}

```