



Študentská vedecká aktivita

Využitie genetického programovania pri navrhovaní ziskových AOS

Martin Müller

Sekcia: Financie, bankovníctvo a investovanie

Konzultant: Ing. Ľubomír Pintér, PhD.

Banská Bystrica

2017

ABSTRAKT

Cieľ riešenia

Cieľom práce je na základe metodického postupu vytvoriť automatický obchodný systém (AOS) pre vybraný menový pár s využitím metódy genetického programovania a overiť testami robustnosti citlivosť vybraných AOS na rôzne zmeny.

Obsah riešenia

V prvej časti práce je teoreticky vymedzená problematika genetického programovania a zároveň je charakterizované obchodovanie prostredníctvom automatických obchodných systémov. V druhej časti je v platforme StrategyQuant, ktorá využíva genetické programovanie, vytvorená populáciu kandidátov za časové obdobie od roku 2004 do roku 2017 pre zvolený menový pár. Pri vytváraní populácie sú použité dve rôzne účelové funkcie fitness s cieľom ich komparácie. Vybrané AOS, ktoré spĺňajú hodnotiace kritériá, sú podrobené testom robustnosti, aby sa zmiernilo riziko preoptimalizácie a overilo sa, či výsledné AOS majú potenciál byť ziskové aj pri reálnom obchodovaní. Pri testovaní robustnosti sú použité nasledovné metódy: historická simulácia a Monte Carlo analýza.

Dosiahnuté výsledky

Navrhnutie metodického postup pre stavbu AOS.

Porovnanie dvoch rôznych hodnotiacich funkcií, aby sa zistil ich vplyv na kvalitu vytvorených AOS.

Preskúmanie z ktorých technických indikátorov a komponentov sa skladajú najrobustnejšie AOS a ako sú na jednotlivých komponentoch závislé.

Kľúčové slová

automatický obchodný systém, genetické programovanie, technická analýza, devízový trh

Obsah

Úvod.....	1
1. Teoretická časť práce.....	2
1.1. Genetické programovanie	2
1.2. Automatické obchodné systémy	3
2. Stavba a testovanie AOS	6
2.1. Vytvorenie populácie kandidátov (AOS) s využitím GP.....	7
2.2. Analýza a testovanie robustnosti AOS	12
Záver.....	21
Zoznam použitej literatúry	22
Prílohy	24

ÚVOD

V súčasnej dobe patria najrôznejšie heuristické metódy založené na darwinovskom princípe evolúcie medzi veľmi populárne a často používané optimalizačné techniky. Evolučné algoritmy sú výsledkom snahy využiť princípy, mechanizmy a vzory vynájdené a overené prírodou pri riešení všeobecných nebiologických problémov (Hynek, 2008). Spúšťáčom týchto snáh, podobne ako vo veľa iných oblastiach, bol rozvoj výpočtovej techniky, ktorá umožnila simulovať tisíce až milióny evolučných procesov, ktoré prebiehajú v prírode.

V tejto práci chceme tento prístup aplikovať do generovania ziskových automatických obchodných systémov na devízovom trhu, konkrétne pre menový pár EURUSD. Devízový trh je z hľadiska obchodovania cez automatické obchodné systémy veľmi zaujímavým trhom, pretože je vysoko likvidný a počas pracovných dní otvorený 24 hodín denne. Jednou z výhod je, že sklzy (gapping) medzi obchodnými dňami sú zriedkavé, čo sa nedá povedať napr. o akciových trhoch. Vďaka vysokej likvidite môžeme obchodovať akýkoľvek objem za aktuálnu cenu bez rizika, že silno ovplyvníme cenu.

V súčasnosti sú automatické obchodné systémy široko používané veľkými hedge fondami, avšak ich spôsob tvorby a zloženie obchodných stratégií nie sú verejnosti známe. Z toho dôvodu sme sa rozhodli bližšie preskúmať jednu z často využívaných metód pre tvorbu automatických obchodných systémov, ktorou je genetické programovanie. Pri tejto metóde sú pravidlá a podmienky v zdrojovom kóde reprezentované stromovou štruktúrou, napr. jeden syntaktický strom môže predstavovať podmienku pre vstup do dlhej pozície a druhý pre vstup do krátkej pozície. Pri tomto type obchodovania, príkazy na nákup alebo predaj sú generované automaticky v obchodnej platforme a milisekunde sú zadávané na trh. V našej práci sa budú vytvorené obchodné systémy skladať najmä z objektívnej technickej analýzy, ktorá zahŕňa technické indikátory a iné dodatočné informácie o cene. Technické indikátory sa vypočítavajú z historických dát a zahŕňajú rôzne matematicko-štatistické metódy. Pri historickej simulácii ich úspech či neúspech je vždy nesporný, pretože nákupné a predajné signály sú generované na základe objektívnych a nemenných pravidiel.

Metóda genetického programovania dokáže zautomatizovať množstvo krokov konvenčného vývoja obchodnej stratégie. Ak je najvzácnejšia komodita čas, tak práve táto metóda ho dokáže ušetriť. Pomocou nej dokážeme, otestovať obrovského množstva kombinácií vstupných, výstupných podmienok alebo nastavenia risk manažmentu za účelom nájdania najvhodnejšieho obchodného systému.

1. TEORETICKÁ ČASŤ PRÁCE

1.1. Genetické programovanie

V 90. rokoch 20. storočia americký informatik John Koza navrhol originálnu modifikáciu genetického algoritmu (GA), ktorú nazval genetické programovanie (GP). Podľa Sekaj (2009, s. 2) sú GA „univerzálnym stochastickým prehľadávacím alebo optimalizačným prístupom, ktorý je v ohraničenom priestore prípustných riešení daného problému schopný nájsť alebo sa aspoň priblížiť ku globálnemu optimu“. Každé potenciálne riešenie je reprezentované usporiadanou množinou hodnôt, ktoré sa nazývajú gény a sú binárneho charakteru.

Na rozdiel od GA pri GP nie sú jedinci reprezentovaní chromozómami pevnej dĺžky, ale predstavujú hierarchické štruktúrované počítačové programy, ktoré po spustení môžu predstavovať potenciálne riešenie daného problému. (Hynek, 2008). Počítačové programy, ktoré v GP predstavujú hľadané riešenie, bývajú najčastejšie reprezentované vo forme syntaktických stromov (Macháček, 2011). Syntaktický strom má svoju špecifickú štruktúru a skladá sa z množiny terminálov $T = \{t_1, \dots, t_n\}$ a množiny funkcií $F = \{f_1, \dots, f_n\}$, ktoré svojimi vlastnosťami zodpovedajú oblasti riešeného problému.

Na začiatku sa náhodnou selekciou terminálov a funkcií vytvorí prvotná populácia kandidátov, ktorá postupom času prechádza evolučným vývojom. Keďže kandidáti predstavujú rôzne kvalitné alternatívy na riešenie, vyžaduje sa existencia nejakej miery, ktorá umožní jednotlivých kandidátov ohodnotiť a porovnať. Zavádza sa na to pojem fitness, pod ktorým sa rozumie vhodnosť, ohodnotenie, sila či reprodukčná schopnosť kandidáta (Mach, 2009). Výber jedincov z počiatočnej populácie sa najčastejšie pri GP využíva turnajovým spôsobom. Pri turnajovej selekcii je náhodne vybraná skupina g ($g \geq 2$) jedincov, ktorí sa následne medzi sebou porovnávajú a lepší z nich sa stane rodičom. Následne sa pre vytvorenie novej populácie z vybraných rodičov použijú genetické operátory kríženia a mutácia. Kríženie je sexuálny operátor, spočívajúci v náhodnom výbere bodu kríženia (uzla) v každom z rodičovských syntaktických stromov a následnej výmene podstromov, ktoré sa nachádzajú pod takto určenými bodmi kríženia. Operátor mutácia pracuje len s jedným kandidátom. Operátor mutácia vyberie náhodne miesto mutácie v syntaktickom strome a celý podstrom vychádzajúci z tohto uzla sa nahradí novým náhodne vygenerovaným stromom (Poli a kol., 2008). Týmto spôsobom vznikne nová generácia kandidátov, v ktorej sú vlastnosti potomkov čiastočne zdedené po rodičoch a čiastočne sú ovplyvnené náhodnými mutáciami v procese reprodukcie. „Ak sa uvedený postup opakuje mnohokrát – počas mnohých generácií, riešenie konverguje k

najlepšiemu riešeniu – ku globálnemu optimu. Pod pojmom mnohokrát si môžeme predstaviť číslo 100, 1000 alebo aj 1 milión“ (Sekaj, 2009, s.2).

1.2. Automatické obchodné systémy

Automatické obchodné systémy (AOS) reprezentuje súbor obchodných pravidiel vo forme programu, pri ktorom sú príkazy na nákup alebo predaj generované automaticky v platforme, ktorá je pripojená na server brokera u ktorého má investor účet. AOS umožňuje obchodovanie bez prítomnosti investora, čo dáva priestor na zlepšenie rýchlosti a presnosti zadaného pokynu, obchodovanie nižších časových úsekov s vyššou frekvenciou obchodov a v neposlednom rade odstraňuje psychickú záťaž pri rozhodovaní. Medzi synonymá používané v kontexte AOS nasledovne: automatické obchodné stratégie, mechanické obchodné systémy alebo obchodné pravidlá (Bryant, 2010).

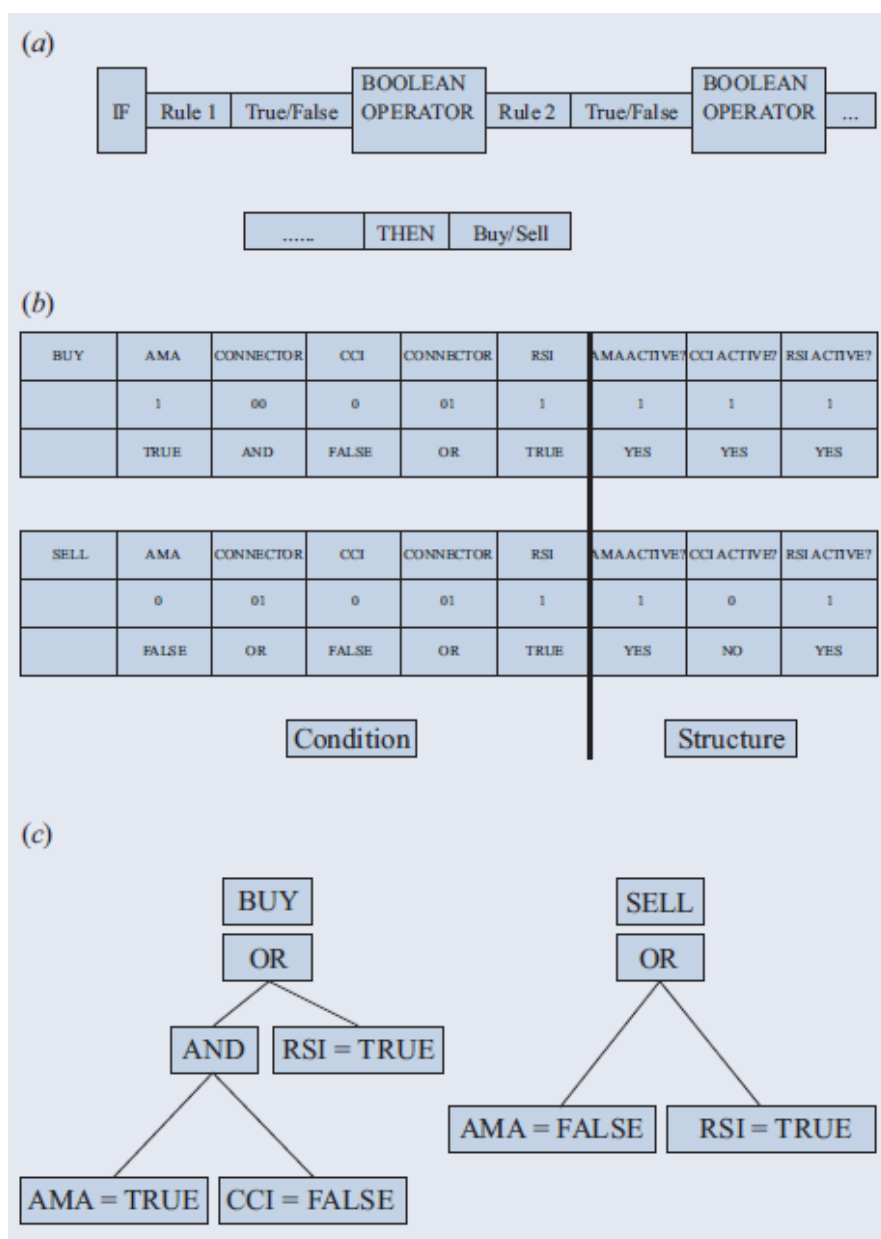
Každý rozumný obchodný systém či už automatický, alebo diskrečný má tri základné komponenty: 1. vstupné a výstupné pravidlo, 2. money (risk) manažment (RM) a 3. stanovenie veľkosti pozície (position sizing).

1. Vstupné a výstupné pravidlá sú hnacím motorom obchodnej stratégie. Môžu byť veľmi jednoduché alebo extrémne komplexné, môžu v sebe kombinovať niekoľko nezávislých technických indikátorov, alebo môže ísť iba o jednoduché časové pravidlo (Pardo, 2008). Obchodné pokyny môžu byť dvoch typov: okamžitá exekúcia alebo čakajúci pokyn. Pri okamžitej exekúcii v momente ako je splnená vstupná podmienka, je vykonaný obchod za aktuálnu trhovú cenu. Čakajúci pokyn je zadaný pod alebo nad aktuálnu trhovú cenu v čase t , pričom k jeho realizácii dochádza v čase $t+1$. Každý čakajúci pokyn má svoju platnosť buď do vyplnenia, alebo do časového vypršania platnosti. Rozlišujeme štyri základné typy obchodných pokynov: Buy Stop, Sell Stop, Buy Limit a Sell Limit. Buy Stop a Sell Limit sa používa ak chceme nakúpiť alebo predať nad aktuálnou cenou. Naopak Sell Stop a Buy Limit sa používa ak chceme predať alebo nakúpiť pod aktuálnou. Stop pokyny sa používajú na prerazenie určitej hodnoty, ak sa očakáva následné pokračovanie v pôvodnom smere. Na druhej strane Limit pokyny sa používajú ak sa očakáva, že cena sa odrazí od určitej úrovne a vráti sa naspäť (Gallo, 2014).

2. Jednou z najdôležitejších častí každej obchodnej stratégie je risk manažment. Jeho cieľom je optimalizovať veľkosť pozícií tak, aby sme maximalizovali svoj zisk a minimalizovali svoje straty. Minimalizovať straty je možné cez tzv. Stop-loss pokyn (SL), ktorý nám buď uzavrie alebo zmenší otvorenú pozíciu. SL je pokyn typu Stop, takže sa realizuje za aktuálnu alebo horšiu cenu. Hlavným cieľom SL je ochrániť investovaný kapitál

tak, aby sme nestratili viac ako máme stanovené v RM. Opačným pokynom ako SL je profit-target pokyn (PT), ktorého hlavným cieľom je uzatvoriť otvorenú pozíciu so ziskom. V prípade, že cena došla na nami stanovenú úroveň, aktivuje sa PT a obchod je uzavretý za Limit pokyn, čo znamená za aktuálnu cenu alebo lepšiu cenu (Demjan a kol., 2011).

3. AOS môže obchodovať buď fixný počet kontraktov v každej jednej pozícií, alebo sa počet kontraktov môže meniť v každom obchode na základe nejakého pravidla. Jedným z často používaných spôsobov, ako nastaviť vhodný počet kontraktov pre určitú pozíciu je tzv. volatilitou upravený position sizing. Tento spôsob stanovuje variabilný počet kontraktov pre každý obchod na základe volatility a veľkosti straty na jeden obchod stanoveného ako fixné percento z účtu alebo fixne stanovená suma (Pardo, 2008).



Obrázok 1 Porovnanie štruktúry reprezentácie AOS

Prameň: Vlastné spracovanie podľa Dempster, Jones, 2001, s. 402.

Z obrázku 6 vyplýva, že vstupné pravidlá sú založené na troch indikátoroch AMA, CCI a RSI. Podmienky sú TRUE alebo FALSE, spojovník je Boolean operátor AND a OR. Príkazy predstavujú pokyn na nákup alebo predaj. Na obrázku 6 (b) je zobrazené, ako sa dá takéto vstupné pravidlo transformovať na binárny reťazec. V tomto prípade je zakódované AND ako 00 a OR ako 01. Posledná časť oddelená čiarou vyjadruje, ktoré indikátory majú byť použité. Nakoniec na obrázku 6 (c) sú vstupné pravidlá reprezentované stromovou štruktúrou, kde spojovníky AND a OR predstavujú vnútorné uzly a indikátory sú vonkajšie uzly, resp. listy (Dempster, Jones, 2001).

Optimalizácia

Optimalizovať obchodnú stratégiu, znamená získať najlepšie a najefektívnejšie nastavenia jej parametrov. Napríklad máme nápad na obchodný systém, ktorý sa skladá z 5 premenných ako je SMA, EMA, indikátor ATR, SL a PT. Každá premenná ma pritom 50 rôznych možností nastavenia. Aby sme zistili kombinácie ich najlepších nastavení, musíme vykonať optimalizáciu. Ak by sme chceli otestovať všetky možné kombinácie nastavení a výpočet nášho počítača pri historickej simulácii jednej kombinácie by trval jednu sekundu, tak by to trvalo $50^5 = 312\,500\,000$ sekúnd, čo je približne 12 rokov. Efektívnejším riešením je využiť evolučné algoritmy, ktoré by takúto úlohu zvládli za niekoľko hodín (Nesnídal, 2011).

Pri optimalizácii obchodnej stratégie na historických dátach, je rozumné rozdeliť historické údaje na účely optimalizácie a hodnotenia. Pôvodné historické údaje, na ktorých je obchodná stratégia testovaná a optimalizovaná, sa označujú ako in-sample dáta (IS), nazývané aj tréningová množina. Výberový súbor údajov, ktorý bol rezervovaný pre účely verifikácie sa nazýva out-of-sample dáta (OOS), alebo testovacia množina. Výkonnosť stratégie na IS dátach by mala byť konzistentná s výkonnosťou na OOS dátach. Zvyčajne sa rozdeľuje celková množina dát v pomere 2/3 IS a 1/3 OOS (Folger, 2014). Ak je optimalizácia vykonaná správne, bude to mať za následok robustný obchodný systém, ktorý spoľahlivo produkuje v reálnom čase výsledky porovnateľné s historickou simuláciou.

Preoptimalizácia (Overfitting) je pojem vypožičaný zo strojového učenia a označuje situáciu, keď sa optimalizácia zameriava skôr na konkrétne zistenie ako všeobecnú štruktúru. Napríklad obchodník môže navrhnúť obchodný systém založený na určitých parametroch, o ktorých vie, že viedli predtým k stratám v IS, ale po niekoľkých iteráciách nájde „optimálne parametre“, ktoré profitujú z vlastností, typických iba pre danú vzorku, ale v celkovej populácii sa vyskytujú ojedinele (Bailey a kol., 2014).

2. STAVBA A TESTOVANIE AOS

V praktickej časti práce budeme využívať pre stavbu AOS software StrategyQuant (SQ), ktorý dokáže na základe genetického programovania automaticky generovať potenciálne AOS na akýkoľvek trh alebo časové pásmo. Samozrejme, že automaticky generovať neznamena, že stačí zahrnúť do stavebných blokov všetky prípustné možnosti a vývojový software SQ nájde AOS, ktorý bude ziskový aj na reálnom účte. K tomu sú potrebné skúsenosti s výberom vhodných nastavení, ktoré sa získavajú postupne používaním tohto softwaru. V našej práci budeme SQ využívať najmä na tvorbu AOS a následne testovanie robustnosti vybraných AOS, ktoré splnia nami stanovené kritériá.

Výberový súbor historických dát

Vzhľadom k vysokej likvidite a nízkym transakčným nákladom budeme vytvárať AOS na najobchodovanejšom menovom páre EURUSD podľa reportu BIS (2016). Výberovú vzorku dát budú tvoriť bid a ask historické ceny za časové obdobie od 1.1.2004 do 30.12.2016. Základom kvalitného backtestu sú kvalitné dáta, čo sa týka štruktúry a presnosti. V práci budeme používať dáta pochádzajúce z dátového centra švajčiarskej banky Dukascopy Bank SA, ktorá poskytuje online brokerské služby. Tieto dáta stiahneme prostredníctvom voľne šíriteľného programu tickdownloader vo formáte CSV a následne ich importujeme do softwaru SQ.

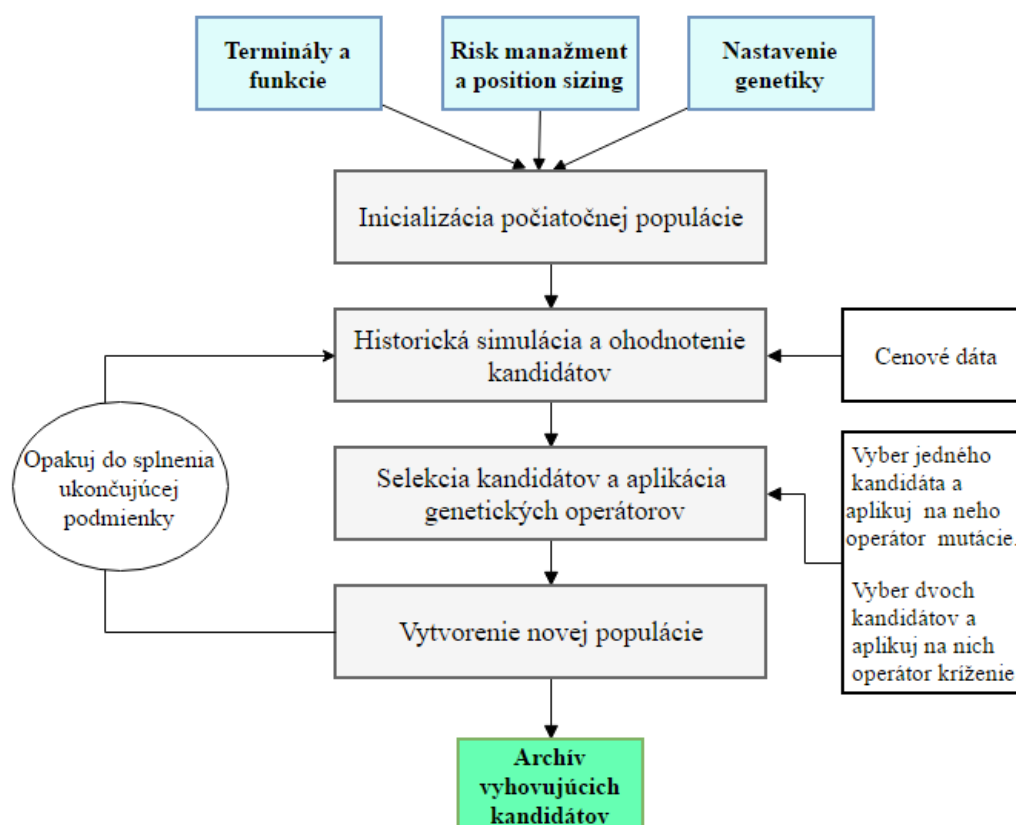
Zaňka (2016), vo svojom článku ukazuje, že medzi najkvalitnejšími tickovými dátami a jednominútovými dátami sú iba minimálne odchýlky pri vyšších časových pásmach. Keďže sa budeme zameriavať na stavbu AOS na vyšších časových rámcoch ako hodinový (H1) časový interval postačuje nám použiť jednominútové dáta. Jednominútové dáta získame vo formáte OHLC, čiže údaje obsahujúce informácie o otváracíj cene (Open), najvyššej cene (High), najnižšej cene (Low) a zatváracíj cene (Close) za časový interval jedna minúta. Dáta je možné v softwari SQ agregovať do vyšších časových intervalov, napr. pre výpočet indikátorov na danom časovom pásme, no pohyb vo vnútri sviečky je simulovaný na základe M1 sviečok.

Historické dáta rozdelíme na tréningovú množinu a testovaciu množinu, pričom testovaciu množinu rozdelíme ešte na dve časti. Tréningovú množinu, na ktorej sa bude algoritmus učiť, zvolíme od 1.1.2004 do 27.06.2012 a testovaciu množinu, na ktorej sa neoptimalizovalo, zvolíme od 27.06.2012 do 30.6.2015, pričom si ponecháme ešte dáta od 30.6.2015 do 30.12.2016 na druhý OOS test. V tejto výberovej vzorke sú obsiahnuté všetky typy trhov, ako býčí trh od roku 2006 po rok 2008, keď EURUSD vzrástlo z úrovne 1,2 na úroveň 1,55; obdobie zvýšenej volatility počas finančnej krízy v USA v roku 2008; medvedí

trh počas dlhovej krízy v európskej menovej únii odkedy Euro klesá smerom k parite s dolárom a obdobie extrémne nízkej volatility v roku 2014.

2.1. Vytvorenie populácie kandidátov (AOS) s využitím GP

Na začiatku sa metódou náhodného výberu pravidiel a podmienok vytvorí počiatočná populácia kandidátov. Následne sa vyberú kandidáti, na ktorých sa aplikujú genetické operátory s cieľom zvýšenia ich ohodnocujúcej funkcie. Počas každej iterácie sa vykonaná historická simulácia pre ohodnotenie fitness každého kandidáta. Počas celého behu sú kandidáti, ktorí spĺňajú výberové kritériá, ukladaní do archívu. Archív sa nastavuje na maximálnu veľkosť, pričom kandidáti s vyšším ohodnotením nahrádzajú kandidátov s nižším ohodnotením. Presná schéma, ako funguje stavba AOS na základe genetického programovania v software SQ, je znázornená na obrázku 2.



Obrázok 2 Vývojový diagram genetického programovania v SQ

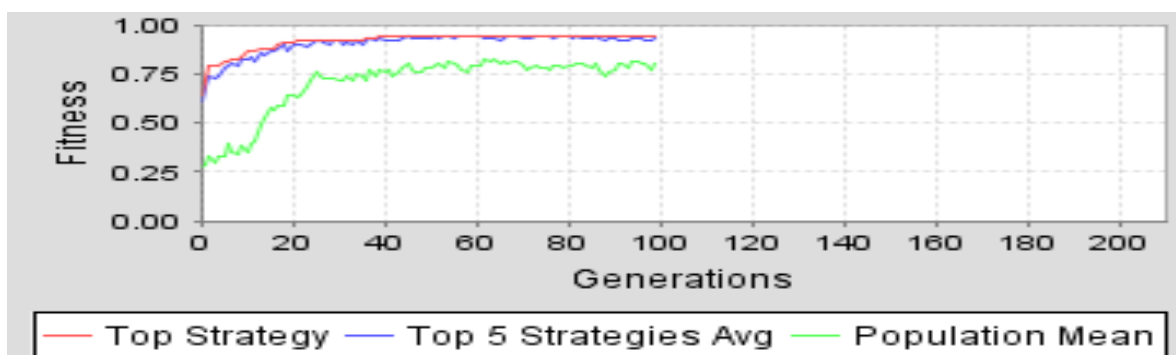
Prameň: Vlastné spracovanie podľa Fric, 2014, s. 16.

Východiskové nastavenia pre GP

Množina terminálov a funkcií, z ktorých sa bude AOS skladať sú uvedené v prílohe 1. Vo vstupných podmienkach sa môžu vyskytovať indikátory ako MACD, RSI, Ichimoko, alebo variácie MA. Vstupovať do obchodov budeme iba cez pokyn Stop. V sekundárnej podmienke použijeme indikátory cenového pásma ako Bolinger Bands a jeho variácie, True Range, Highest atď. Veľkosť riskovaného kapitálu na jeden obchod stanovíme pevne vo výške 100 USD pri kapitále 5000 USD, čiže na jeden obchod budeme maximálne riskovať 2 % z počiatočného kapitálu. Výšku Stop Lossu nastavíme buď ako fixné číslo v pipoch alebo ako volatilitou upravený SL vypočítaný na základe indikátora ATR. Výšku SL a PT od 30 pips do 350 pips.

Kvalita výsledných AOS vytvorených prostredníctvom genetického programovania je ovplyvnená najmä nastavením nasledujúcich parametrov genetického programovania. Pri zvolení ich nastavení budeme vychádzať z (Deng a kol., 2012), (Myszkowski, Bicz, 2010), (Fric, 2014).

- Veľkosť populácie – tento parameter predstavuje maximálny počet kandidátov v počiatočnej populácii a ďalších generáciách. Veľkosť počiatočnej populácie obmedzíme na 200 kandidátov a veľkosť populácie pre ďalšie generácie stanovíme na 100 kandidátov.
- Počet generácií – tento parameter určuje, po koľkých generáciách algoritmus ukončí prehľadávanie a nastavíme ho na 100 generácií. Stanovíme tiež ukončovaciu podmienku, ak po piatich generáciách priemerné fitness 5 najlepších stratégií stagnuje celý proces sa reštartuje a začne odznova. Na grafe 1 simulujeme priebeh evolúcie počas celých 100 generácií. Zhruba od 45. generácie začína evolúcia stagnovať a ďalších 5 generácií neprineslo skoro žiadne zlepšenie fitness. Z hľadiska úspory času je lepšie stagnujúcu evolúciu reštartovať.



Graf 1 Stagnácia evolúcie

Prameň: Vlastné spracovanie.

- Hĺbka stromu – je maximálna hĺbka syntaktického stromu, ktorú nastavíme pre počiatočnú populáciu na 5, a v ďalších generáciách môže byť aj väčšia.

- Množina funkcií – predstavujú najmä aritmetické a logické operátory, ktorých výber je popísaný vyššie.
- Množina terminálov – zahŕňa premenné alebo konštanty v syntaktickom strome napr. indikátory, informácie o cene ako high, low, highest atď.,
- Selekcia – sa uskutoční turnajovým spôsobom. Veľkosť turnaja je stanovená na 2. Pričom je nastavená uchovávací selekcii, že dvaja kandidáti s najvyššou fitness sa dostanú do ďalšej generácie bez zmeny.
- Podiel mutácie – mutácia zabráňuje predčasnej konvergencii a uviaznutiu v lokálnom optime. Podiel mutácie vyjadruje, koľko percent kandidátov v novej generácii má vzniknúť pomocou genetickej operácie mutácia. Tento podiel nastavíme na 10 %. Bod mutácie sa určí uniformne pseudonáhodným spôsobom.
- Podiel kríženia – koľko percent kandidátov v novej generácii má vzniknúť pomocou genetickej operácie kríženie. Tento podiel nastavíme na 90%. Bod kríženia sa určí náhodne v prvom rodičovi a následne sa identifikuje množina kompatibilných bodov kríženia v druhom rodičovi, z ktorej je náhodne vybraný jeden bod pre kríženie.
- Fitness – zvolený typ ohodnocujúcej funkcie má dominantný vplyv na kvalitu výsledného riešenia. Táto funkcia by mala byť úzko zviazaná s konkrétnym problémom.

Existuje mnoho rôznych spôsobov, ako merať výkonnosť obchodných stratégií. V teórii portfólia sa bežne používa Sharpe ratio, ktoré sa počíta ako nadmerný výnos delený rizikom meraným pomocou štandardnej odchýlky. Avšak vo svete obchodovania (tradingu) nie sú obchodníci zatiaľ volatilitou výnosov. Namiesto toho sa viac sleduje maximálny pokles kapitálu od už dosiahnutých maxím označovaný v anglickej literatúre „drawdown“. Maximálny pokles kapitálu väčšinou predstavuje hranicu, ktorú sú investori alebo sponzori ochotní akceptovať a jeho prekročenie má za následok výber investovaného kapitálu. Často používaným ukazovateľom výkonnosti alternatívnych investícií a hedge fondov je pomer zisku voči maximálnemu poklesu. Pri tomto spôsobe hodnotenia výkonnosti sa považuje za najlepšiu aproximáciu rizika najhorší kontinuálny sled stratových a ziskových obchodov za dostatočne dlhé časové obdobie v percentuálnom alebo absolútnom vyjadrení. Zisk predstavuje anualizovaný, priemerný alebo celkový čistý zisk za sledované obdobie. (Dempster, Jones, 2001).

Hlavným cieľom hodnotiacej funkcie bude nájdenie kvalitných obchodných systémov. Pod kvalitným systémom nebudeme rozumieť najziskovejšie nastavenie parametrov, ale skôr schopnosť systému zostať ziskový aj pri zmenách parametrov a trhovými podmienok. V práci

použijeme dve rôzne účelové funkcie fitness s cieľom ich komparácie. Ako prvú účelovú funkciu si zvolíme najjednoduchší variant pomeru zisku vo vzťahu k maximálnemu poklesu:

$$\text{Ret/DD ratio} = \text{celkový čistý zisk} / \text{maximálny pokles}$$

kde celkový zisk predstavuje súčet všetkých ziskových a stratových obchodov v absolútnom vyjadrení za celé obdobie a maximálny pokles vyjadruje najhorší prepád kapitálu v absolútnom vyjadrení za celé obdobie. Čím vyššie číslo, tým väčší zisk vo vzťahu k riziku. Z toho dôvodu budeme maximalizovať túto účelovú funkciu. Za dostatočné hodnoty sa považuje hodnota 0,8-1 za jeden rok historického obdobia. Tento pomerový ukazovateľ v podstate vyjadruje, koľko sme zarobili na jeden riskovaný dolár za dané obdobie.

Ako druhú fitness funkciu zvolíme súčet prvého ukazovateľa Ret/DD ratio s ukazovateľom kvality obchodného systému nazývaným System Quality Number® (SQN®). SQN® meria vzťah medzi strednou hodnotou R-násobku a štandardnou odchýlkou R-násobku, pričom sa ešte zohľadňuje počet obchodov. R-násobok je séria ziskov a strát vyjadrená z hľadiska risk reward ratio. Ak máme počiatočný risk na obchod 100 USD a zarobíme 200 USD, tak náš R-násobok je 2R. To isté platí aj pri strate len s opačným znamienkom. R-násobok vypočítame ako:

$$R_t = P\&L_t / \bar{L}$$

kde $P\&L_t$ je zisk alebo strata, $t = \{1, \dots, n\}$ kde n vyjadruje počet obchodov, $\bar{L}$ je celková priemerná strata za n obchodov v absolútnom vyjadrení. Potom SQN® môžeme formulovať ako:

$$\text{SQN}^\circledast = (\text{priemerné } R / \text{smerodajná odchýlka } R) * \sqrt{N}$$

kde N predstavuje celkový počet obchodov. Čím vyšší priemerný R-násobok a menšia jeho odchýlka, tým dostávame lepšie ohodnotenie. N sa vníma, tiež pozitívne a dá interpretovať ako počet obchodných príležitostí. Štandardná interpretácia SQN® je nasledovná: 1,6-1,9 je minimálne skóre; 2-2,4 je priemerné skóre; 2,5-2,9 je dobré skóre, 3-5 celkom dobré skóre, 5-6,9 vynikajúce skóre; 7 a viac je „holy grail“ (Tharp, 2008).

Výberové kritéria

Pri genetickom programovaní vytvoríme tisícky kandidátov, z ktorých vyberieme a uložíme do archívu pre ďalšie hodnotenie a testovanie iba tých, ktorí spĺňajú určité požiadavky. Keďže v tejto fáze sú obchodné systémy ohodnotené iba z pohľadu historickej výkonnosti, pri výberových kritériách nebudeme príliš prísni, pretože príliš prísne kritériá vedú k preoptimalizácii. Nehľadáme teda AOS s rovnou kapitálovou priamkou a najvyššou ziskovosťou, ale skôr robustné systémy fungujúce pri rôznych trhových podmienkach.

Pri hodnotení budeme využívať hlavne pomerové ukazovatele, pretože sa nedajú ovplyvniť veľkosťou pozície a sú nezávislé od veľkosti kapitálu. V tabuľke 1 môžeme vidieť minimálne požiadavky ukazovateľov tvoriacich výberové kritériá. Minimálny počet obchodov za celé obdobie od 1.1.2014 do 30.6.2015 (74% IS a 26% OOS) sme stanovili na 300. Pre Ret/DD ratio požadujeme hodnotu aspoň 0,8 za rok. SQN[®] nastavíme na 1,6 a profit faktor na 1,3. Pri % Wins požadujeme aspoň 30 % podiel ziskových obchodov na celkovom počte obchodov. Priemerný zisk za deň v IS období nesmie klesnúť o viac ako polovicu v OOS období.

Tabuľka 1 Výberové kritériá pre uloženie kandidátov do archívu

Ukazovateľ		IS	OOS
Počet obchodov	≥	220	80
Ret/DD ratio	≥	6,8	2,4
SQN [®]	≥	1,6	1,6
Profit Factor	≥	1,3	1,3
% Wins	≥	30	30
Priemerný zisk za deň (IS/OOS Ratio) ≥ 50 %			

Prameň: Vlastné spracovanie.

Výsledky GP

Vykonalí sme dve generovania, v ktorých bola odlišná iba účelová funkcia fitness. V prvom prípade sme maximalizovali hodnotu ukazovateľa Ret/DD ratio a v druhom prípade súčet hodnôt zložených z Ret/DD ratio a SQN[®]. Pri oboch generovaniach sme naplnili archív na 5000 kandidátov, pričom pri dosiahnutí maximálnej kapacity archívu, lepší kandidáti nahrádzali tých slabších. Pri oboch generovaniach sme nechali čas na naplnenie do archívu 24 hodín. V nasledujúcej tabuľke 2 sú zobrazené priemerné hodnoty vybraných ukazovateľov za celú populáciu, pričom sú porovnané výsledky dvoch fitness funkcií.

Tabuľka 2 Výsledky GP a porovnanie fitness

	IS				OOS			
	% Win	PF	Ret/DD	SQN [®]	% Win	PF	Ret/DD	SQN [®]
1. fitness	45,45	1,64	14,26	4,19	41,46	1,5	4,38	2,10
2. fitness	44,73	1,56	13,49	4,52	41,44	1,42	3,99	2,11

Prameň: Vlastné spracovanie.

V tabuľke 2 vidíme, že priemerný podiel ziskových obchodov sa pohybuje v intervale 41 % až 46 %, čiže máme viac stratových ako ziskových obchodov. Medzi priemernými hodnotami ukazovateľov porovnávaných fitness nepozorujeme značné rozdiely, 1. fitness má mierne lepšie výsledky. Prirodzene každá fitness funkcia dosahuje najlepšie hodnoty v IS pri ukazovateli, ktorý maximalizovala. Hodnoty v IS perióde sú lepšie ako v OOS perióde. Pri SQN a Ret/DD sú hodnoty ovplyvnené aj počtom obchodov, preto sú výrazne lepšie v IS. Ak

sa pozrieme na Profit Factor, tak pri 1. fitness sa zhoršil o 8,5 % a pri 2. fitness o 8,97% v OOS oproti IS, čo považujeme za celkom stabilný výsledok. Ak si prepočítame Ret/DD na jeden rok, tak získame pri 1. fitness výsledok 1,68 v IS a 1,46 v OOS, čo je v relatívnom vyjadrení pokles o 13,9 %, pri 2. fitness výsledok 1,59 v IS a 1,33 v OOS, čo je v relatívnom vyjadrení pokles o 16,4 %. Celkovo považujeme výsledky medzi IS a OOS periódou ako konzistentné.

2.2. Analýza a testovanie robustnosti AOS

Vývoj obchodných stratégií prostredníctvom genetického programovania je v podstate metóda zhora nadol, pretože najprv sa vytvorí dostatočný počet kandidátov a následne sa analyzuje a hodnotí, ktorí kandidáti sú vhodní pre nasadenie na reálny účet. Zatiaľ máme informácie iba o historickej výkonnosti vytvorených kandidátov, avšak nemáme žiadne informácie o tom ako sú citliví na rôzne zmeny. Cieľom testov robustnosti je overiť, ako sa AOS budú správať, keď uskutočníme malé zmeny vstupných údajov, pozmeníme historické dáta alebo parametre AOS. Z toho dôvodu v tejto podkapitole podrobíme vytvorených 10000 kandidátov testom robustnosti. Hlavným cieľom testov robustnosti bude overiť, či AOS spĺňajú 4 hlavné vlastnosti robustných stratégií podľa Pardo (2008). Obchodná stratégia sa považuje za robustnú, ak si udrží stabilnú a konzistentnú výkonnosť pri:

1. rôznych typoch trendov a podmienok (rastúci, klesajúci, bočný trend, zvýšená volatilita, zníženie likvidity atď.),
2. inom korelujúcom trhu (GBPUSD),
3. nižšom a vyššom časovom pásme (M30 a H4)
4. širokej škále okolitých nastavení parametrov (WFA).

Test na druhých OOS dátach, inom trhu a časovom pásme

Účelom týchto testov je získanie predbežnej predstavy o robustnosti vytvorených AOS a vyfiltrovaní nepostačujúcich kandidátov. Tieto testy sú najmenej časovo náročné a vyžadujú najmenší výpočtový výkon počítača.

Pri vývoji stratégie a ich výbere do archívu sme nepoužili celé historické dáta a uchovali sme si posledný rok a pol pre druhý OOS test. Našou požiadavkou je, aby obchodné systémy boli schopné udržať si výkonnosť aj na dátach, na ktorých predtým neboli optimalizované a hodnotené. Výsledky nám umožnia zistiť koľko z vytvorených AOS si dokáže udržať výkonnosť aj na druhých OOS dátach, t.j. od 30.6.2015 do 30.12.2016. Pri týchto testoch požadujeme, aby kandidáti spĺňali aspoň dve výberové kritéria $Ret/DD \text{ ratio} \geq 1,2$ a $PF \geq 1,3$. Ak nesplnia tieto kritériá, tak ich vymažeme. Po spustení historickej simulácie za vybrané

obdobie a odfiltrovaní AOS nespĺňajúcich kritériá, nám ostalo z 1. fitness sady 2886 kandidátov a z 2. fitness sady 2858 kandidátov. Testami teda neprešlo zhruba 42 % populácie.

Ďalším dôležitým testom je historická simulácia na inom korelujúcom, resp. podobnom trhu. Predpokladáme, že podobný trh ako EURUSD je GBPUSD. Pri overení výkonnosti kandidátov na menovom páre GBPUSD požadujeme $r^2 \geq 0,8$ a $PF \geq 1,2$, tieto kritériá splnilo z 1. fitness 1619 a z 2. fitness 1665 kandidátov. Skutočnosť, že si tieto AOS dokázali udržať výkonnosť na viac ako jednom trhu, bez toho aby to bol na začiatku úmysel, vypovedá o tom, že nie sú závislé iba na určitom type trhových podmienok. Na grafe 2 sú zobrazené dva AOS, v hornej časti grafu je zobrazený AOS ziskový na obidvoch menových pároch a v dolnej časti je AOS ziskový iba na EURUSD. Zelená čiara predstavuje kapitálovú krivku na GBPUSD, svetlomodrá zobrazuje kapitálovú krivku na EURUSD a tmavomodrá čiara zobrazuje súhrnné výsledky za obidva menové páry.



Graf 2 Porovnanie AOS na EURUSD a GBPUSD

Prameň: Vlastné spracovanie.

Tretím testom je test na inom časovom pásme. Ako iné časové pásmo sme zvolili jeden nižší a vyšší časový rámec. Chceme zistiť ako sa AOS správajú na rovnakom trhu, ale s inou štruktúrou ceny. Pri tomto teste overujeme ako komponenty AOS, ktoré počítajú napr. stop loss cez ATR, alebo periódu indikátora na vybranom časovom rámci, reagujú na rýchlejšie alebo pomalšie cenové zmeny. Prvý test bude historická simulácia na časovom rámci M30, pri ktorom požadujeme $r^2 \geq 0,8$ a $PF \geq 1,2$. Druhý test vykonáme na časovom intervale H4. Pri tomto teste vynecháme ukazovateľ stability (r^2), pretože mnoho kandidátov malo iba okolo 50

obchodov. Požadujeme hodnoty ukazovateľov PF aspoň 1,2 a Ret/DD aspoň 1. Získali sme nasledovné výsledky: pri teste na M30 sme odfiltrovali z 1. fitness 474 a pri 2. fitness 301 kandidátov, pri H4 sme vymazali z 1. fitness 183 a pri 2. fitness 448 kandidátov. Celkovo nám po všetkých testoch v tejto fáze ostalo z 1. fitness 949 a pri 2. fitness 916.

Využitie Monte Carlo (MC) analýzy pri testovaní robustnosti AOS

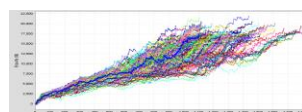
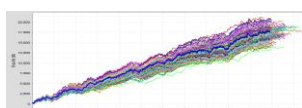
MC analýza je stochastická metóda pre analýzu javov, na ktoré majú vplyv náhodné faktory (Davay, 2014). Existuje viacero prístupov, ako sa metóda MC uplatňuje. V našom prípade princíp spočíva v mnohonásobnom simulovaní vybraného parametra, na základe čoho sa získajú simulované hodnoty kapitálovej krivky, z ktorých sa určí spoľahlivosť, s akou sledovaný výstup našej analýzy nastane. Jednotlivé simulácie nemajú veľkú vypovedaciu hodnotu, avšak čím je vyšší počet opakovaní, tým je presnejší dosiahnutý výsledok. V programe SQ existujú v podstate dva druhy Monte Carlo simulácií. Prvý jednoduchší spôsob využíva už výsledné obchody, pri ktorých sa mení iba poradie obchodov alebo zloženie množiny obchodov. V druhom zložitejšom spôsobe sa menia vybrané parametre testovania ako perióda indikátorov, historické dáta, kurzové rozpätie atď. Pri každej iterácii sa urobí zmena daného parametru v povolenom rozsahu a následne sa spustí kompletne nová simulácia s takto pozmeneným parametrom. Pred spustením MC analýzy si program vyžaduje aby sme nastavili počet simulácií, menené parametre a hladinu významnosti (zvyčajne sa volí $\alpha = 0,05$) na ktorej chceme testovať zvolený výstup. Tak isto ako v predchádzajúcej časti cieľom MC analýz bude vyfiltrovať nerobustných kandidátov.

Kapitálová krivka nášho konkrétneho AOS vyjadruje historické výsledky a má nespornú vypovedajúcu schopnosť, ale súčasne aj svoje limity. Predovšetkým reflektuje iba jednu konkrétnu historickú postupnosť obchodov. Monte Carlo analýzu využijeme na simuláciu náhodného poradia historických obchodov. Ak sčítame zisky a straty z týchto obchodov, náš celkový čistý zisk, Profit Factor, alebo priemerný zisk sa s ohľadom na zmenu poradia jednotlivých obchodov nijako nezmení. Čo sa však pri inom poradí realizovaných obchodov daného obchodného systému zmení, je maximálny pokles kapitálu. V softwari SQ spustíme 500 simulácií, pri ktorých najprv náhodným spôsobom iba zamiešame poradie obchodov a následne ešte vykonáme prevzorkovanie, tzv. „resampling“. Pod pojmom resampling sa v softwari SQ rozumie, že náhodný počet obchodov je vymazaný a náhodný počet obchodov je duplikovaný pri zachovaní celkového počtu obchodov. Táto metóda zmení aj ukazovatele ako čistý zisk, alebo Profit Factor. Nech je N počet obchodov originálneho AOS, tak pre každú simuláciu sa náhodne vyberie N obchodov zo zoznamu existujúcich obchodov, pričom niektoré z nich môžu

byť vybrané aj viac krát a niektoré nemusia byť vybrané vôbec. Pri náhodnom výbere sa v SQ nepoužíva žiadne štatistické rozdelenie a obchody sa vyberajú uniformne pseudonáhodne z celej existujúcej množiny, až kým sa nenaplní nová množina. V každej simulácii sa teda mení poradie aj množina obchodov. Podrobnejší popis prvej aj druhej MC analýzy je uvedený v prílohe 2 vo forme Java kódu. Po spustení simulácie získavame 500 nových kapitálových kriviek, ktorých výstupy (výsledné ukazovatele) sú zoradené od najmenej hodnoty po najväčšiu s cieľom určenia intervalu spoľahlivosti. Pri týchto dvoch testoch budeme požadovať, aby si ukazovateľ Ret/DD pri spoľahlivosti 95 % zachoval aspoň polovicu zo svojej originálnej hodnoty a ukazovateľ maximálny pokles kapitálu v % pri spoľahlivosti 95 % sa zvýšil najviac dvojnásobne. Prvým testom, v ktorom sa zmenilo iba poradie obchodov, neprešlo 569 kandidátov z 1. fitness a 519 kandidátov z 2. fitness. Druhým testom - prevzorkovaním neprešlo 254 kandidátov z 1. sady a 327 kandidátov z 2. sady. Celkovo nám ostalo 93 kandidátov z 1. fitness a 103 kandidátov z 2. fitness.

V snahe vyhnúť sa preoptimalizácii parametrov obchodného systému využijeme Monte Carlo analýzu pri simulovaní nastavení okolitých hodnôt parametrov. Budeme teda testovať citlivosť kandidátov na malé zmeny v nastavení parametrov. Každý z vytvorených AOS má SL ak by bol SL napr. 70 pips a zmenili by sme ho o 20 %, tak vyžadujeme aby fungoval aj pri hodnote 56 pips alebo 84 pips, to isté pri indikátoroch, ak nastavíme maximálny rozsah zmeny 20 %, tak vyžadujeme aby indikátor SMA s periódou 100 fungoval v rozmedzí hodnôt 80 až 120. Tým sledujeme, aby vytvorené obchodné systémy neboli svojím nastavením hodnôt parametrov napasované na historické dáta. MC analýza si teda vyžaduje zvoliť maximálny rozsah zmeny hodnôt parametrov a pravdepodobnosť zmeny parametrov, čo znamená pravdepodobnosť, že nejaký parameter zmení svoje hodnoty. V našom prípade zvolíme maximálnu zmenu parametrov 20 %, pravdepodobnosť ich zmeny na 20 %, hladinu významnosti 0,05 a počet simulácií 500. Pri MC simuláciách použijeme celé historické obdobie. Ako kritérium na základe ktorého vymažeme preoptimalizované AOS, si zvolíme pomerový ukazovateľ Ret/DD pri spoľahlivosti 95 % a zároveň budeme vizuálne subjektívne hodnotiť vzhľad kapitálovej krivky. Pri subjektívnom filtrovaní budeme požadovať, aby simulované krivky boli v čo najväčšej tesnosti s originálnou krivkou a neboli oveľa dlhšie/kratšie, resp. nezvýšil/neznižil sa im extrémne počet obchodov. Na grafe 3 príkladne ilustrujeme tieto kritériá. Na prvej časti grafu je znázornený AOS, ktorý spĺňa naše kritériá a na druhej časti grafu je AOS, ktorý nespĺňa naše kritériá. Z týchto simulácií sme dostali nasledovné výsledky: pri zohľadnení Ret/DD sme vymazali 2 kandidátov z 1. fitness a iba 1 kandidáta z 2.

fitness, avšak pri vizuálnom testovaní sme vymazali 85 kandidátov z 1. fitness 84 kandidátov z 2. fitness.



Graf 3 Citlivosť AOS na zmeny parametrov simulovaných MC analýzou

Prameň: Vlastné spracovanie.

Analýza závislosti AOS na svojich parametroch

V tejto časti preskúmame, z ktorých technických indikátorov a komponentov sa skladajú najrobustnejšie AOS a ako sú na jednotlivých komponentoch závislé. Z logiky fungovania metódy genetického programovania, evolúcia počas generácií konverguje k optimálnemu riešeniu. Z toho dôvodu je dosť možné, že AOS sa budú skladať z podobných parametrov, ktoré sa odlišujú iba v rôznych hodnotách nastavení alebo z indikátorov s podobnou logikou.

Celkovo nám z počiatočného súboru 10000 kandidátov ostalo iba 6 kandidátov z 1. fitness a 18 kandidátov z 2. fitness. Z dôvodu možnej podobnosti AOS ich zatriedime do skupín na základe vzhľadu kapitálových kriviek a zloženia zdrojového kódu. Z každej skupiny následne vyberieme jednu obchodnú stratégiu, ktorá má najmenší počet celkových podmienok a jej ukazovateľ Ret/DD nie je výrazne horší od priemeru. Obchodné systémy sme zatriedili do troch skupín s rovnakým vstupným pravidlom a podobnou podmienkou umiestnenia čakajúceho pokynu pre vstup do obchodu. AOS v jednotlivých skupinách sa od seba odlišovali

najmä vo výstupnej podmienke, v parametroch risk manažmentu ako výška SL, PT a v odchýlke sekundárnej podmienky. Do prvej skupiny môžeme zaradiť AOS, ktoré majú vstupné pravidlo neobchoduj počas nedele alebo pondelka a ich podmienka pre umiestnenie Stop pokynu je otváracia cena novej sviečky + odchýlka meraná indikátorom Bolinger Bands Width Ratio. Do tejto skupiny môžeme zaradiť 11 kandidátov z toho 5 z 1. fitness. Druhá skupina sa vyznačuje tým, že vstupné pravidlo tvorí indikátor MACD a Stop pokyn umiestňuje na najvyššiu/najnižšiu (Highest/Lowest) sviečku z 55 posledných sviečok. Túto skupinu tvorí 6 kandidátov. Vstupné pravidlo poslednej skupiny je indikátor RSI a Stop pokyn sa umiestňuje na Highest/Lowest sviečku za posledných 96 sviečok + odchýlka meraná indikátorom ATR, alebo iba na Highest/Lowest sviečku za posledných 47 sviečok. Do tejto skupiny zaraďujeme 7 kandidátov z toho jedného z 1. fitness.

Z každej tejto skupiny sme vybrali 3 AOS a ich zdrojový kód vo formáte pseudo kódu uvádzame v nasledujúcich prílohách: AOS z prvej skupiny, ktorého označíme ako STR 21.5 je uvedený v prílohe 3. AOS z druhej skupiny, ktorého označíme ako STR 41.5 je uvedený v prílohe 4 a AOS z tretej skupiny, ktorého označíme ako STR 20.3 je uvedený v prílohe 5. Popis a výpočet indikátorov tvoriacich zdrojový kód je uvedený v prílohe 6. Ako je možné vidieť zo zdrojových kódov obchodné systémy sa skladajú z:

Primárnej podmienky – ide o vstupné pravidlo na základe ktorého sa AOS rozhoduje, či vstúpiť do dlhej, krátkej alebo žiadnej pozície. Slúži v podstate ako filter. Vstupná podmienka v STR 41.5 a 20.3 definuje, či je trh v rastúcom alebo klesajúcom trende. Sekundárnej podmienky – predstavuje cenovú úroveň, na ktorú sa umiestni čakajúci pokyn Stop. Tento pokyn sa zadáva iba ak nie je otvorený obchod v žiaducom smere zadávania. Pri všetkých troch AOS sa čakajúce pokyny nahrádzajú. Risk manažmentu – zahŕňa SL, PT, stop tralling (ST), profit trailing (PTg) a posun na otváraciu cenu obchodu (BE). Stop trailing je pokyn, ktorý posúva SL na určitú cenovú úroveň, ak sa obchod dostáva do zisku, profit trailing je pokyn, pri ktorom sa SL najprv posunie na BE a následne sa SL posúva vyššie pri rastúcom zisku. Výstupnej podmienky (exit rule) – je podmienka, ktorá určuje, kedy bude pozícia ukončená mimo Risk manažmentu.

Aby sme zistili, ako sú vybrané AOS závislé na svojich parametroch, urobíme menšie úpravy v zdrojovom kóde. Najprv nahradíme primárnu podmienku pravidlom Always True, resp. $1=1$, v podstate to znamená, že AOS nebudú mať žiadny vstupný filter a po celý čas bude aktivovaný signál do dlhej alebo krátkej pozície. Po každej úprave vrátime AOS do východiskového stavu a prevedieme ďalšiu úpravu. Vynecháme tiež parametre risk manažmentu ako ST, PTg a BE. Poslednou zmenou bude odstránenie výstupnej podmienky. Nasledujúca tabuľka 3 zobrazuje

ako sa zmenia pôvodné hodnoty ukazovateľov AOS ak odstránime určitú podmienku a teda môžeme pozorovať ako sú AOS závislé na svojich parametroch.

Tabuľka 3 Závislosť AOS na svojich parametroch

	Ukazovateľ	Pôvodná hodnota	1=1	Bez ST	Bez PTg	Bez BE	Bez exit rule	Bez BE a exit rule
STR 21.5	Ret/DD	11,92	11,92	10,6	X	11,91	11,1	11,09
	r^2	0,94	0,94	0,92	X	0,94	0,93	0,93
	#obchodov	925	925	924	X	925	924	924
STR 41.5	Ret/DD	17,93	8,54	9,34	X	17,7	18,28	17,98
	r^2	0,96	0,94	0,95	X	0,96	0,97	0,96
	#obchodov	1094	2369	1051	X	1073	1090	1070
STR 20.3	Ret/DD	9,65	1,34	X	2,91	X	12,56	X
	r^2	0,85	0,55	X	0,78	X	0,87	X
	#obchodov	978	5999	X	822	X	974	X

Prameň: Vlastné spracovanie.

Z tabuľky 3 vidíme, že všetky AOS si dokážu udržať výkonnosť aj po odstránení určitého parametra, dokonca niektoré potom dosahujú lepšie výsledky. Ak sa zameriame na STR 21.5 môžeme sledovať, že po odstránení primárnej podmienky si zachovala úplne identickú výkonnosť. Pod zachovaním výkonnosti rozumieme, že hodnota ukazovateľa neklesne o viac ako polovicu. Je to z toho dôvodu, že tento AOS má v primárnej podmienke iba pravidlo neobchoduj v nedeľu a keďže sa na forexe v nedeľu obchoduje iba od 23:00 do 24:00 má minimálny vplyv. Aj napriek tomu túto podmienku ponecháme, pretože počas tejto hodiny je nízka likvidita a vznikajú riziká s ňou spojené. Z tabuľky 3 tiež vidíme, že STR 21.5 nie je závislá ani na ostatných analyzovaných parametroch. Najviac citlivo reagovala na odstránenie posuvného SL, kedy sa jej Ret/DD v relatívnom vyjadrení znížil o 11 %. Odstránenie posunu na BE a zároveň aj exit rule malo vplyv zhruba 7 % na zníženie hodnoty ukazovateľa Ret/DD. Aby sme zvýšili robustnosť AOS, odstránime natrvalo posledné dve podmienky (BE a exit rule), pretože čím menej parametrov má obchodný systém, tým je menšia šanca preoptimalizácie. Pri STR 41.5 odstránenie primárnej podmienky spôsobilo viac ako zdvojnásobenie počtu obchodov a Ret/DD sa zmenšil o 48 % oproti pôvodnej hodnote. Pri odstránení ST sa výraznejšie zmení iba Ret/DD, ktorý sa zníži z 17,93 na 9,34 a počet obchodov sa zmenší o 43 obchodov. Konštatujeme, že AOS je závislé na týchto podmienkach, ale stabilita kapitálovej krivky ako indikuje r^2 sa veľmi nezhoršila. Na druhej strane odobratie parametru posun na BE a výstupnej podmienky má dokonca za následok zlepšenie ukazovateľov STR 41.5, z toho dôvodu tieto zmeny ponecháme a v ďalšej časti práce budeme pracovať bez nich. Z analyzovaných troch STR je najviac citlivý na odobratie primárnej podmienky STR 20.3,

ktorého počet obchodov sa zvýšil z 978 na 5999, r^2 sa zhoršilo z 0,85 na 0,55 a Ret/DD sa znížil o 86 % oproti pôvodnej hodnote. Odstránenie profit trailingu spôsobilo zníženie počtu obchodov o 156, zhoršenie r^2 o 8 % a Ret/DD o 70 % oproti originálnej hodnote. Konštatujeme, že STR 20.3 je závislá na týchto parametroch. Jediné zlepšenie nastalo v prípade odstránenia výstupnej podmienky. Z dôvodu výraznej závislosti na svojich parametroch nebudeme už s STR 20.3 ďalej pracovať.

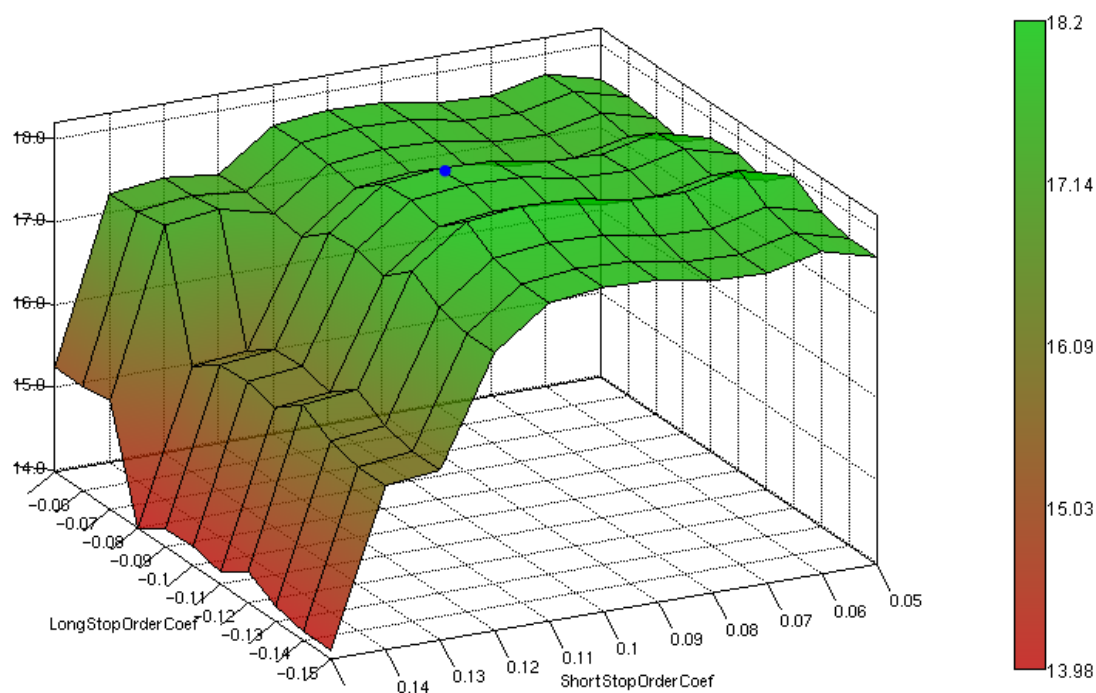
V tejto časti práce využijeme optimalizáciu, aby sme zanalyzovali či nastavené hodnoty vybraných parametrov sú vo funkčnej oblasti. Pod funkčnou oblasťou budeme rozumieť plochu v priestore prehľadávania, ktorá vykazuje podobné vlastnosti. Vyžadujeme teda aby sa hodnota nášho parametru nachádzala v priestore, v ktorom sú aj okolité hodnoty parametrov ziskové, resp. ziskové vo vzťahu k riziku. Využijeme pri tom metódu úplného prehľadávania, tzv. brute force, ktorá postupne otestuje všetky kombinácie nastavení parametrov zo zadaného rozsahu. Vytvoríme logické dvojice analyzovaných parametrov, ktorými budú SL a PT a koeficienty pre umiestnenie pokynu do dlhej pozície (coef long) a krátkej pozície (coef short). Rozsah možných hodnôt môžeme sledovať v tabuľke 4:

Tabuľka 4 Hodnoty pre optimalizáciu parametrov

	Paremeter	Originál	Štart	Koniec	Krok	# Kombinácií
STR 21.5	PT	127	92	162	5	192
	SL koeficient	2,32	1,82	3,00	0,10	
	Long coef	0,5	0,25	0,75	0,05	121
	Short coef	-0,5	-0,25	-0,75	-0,05	
STR 41.5	PT	199	149	249	10	110
	SL koeficient	2,88	2,48	3,28	0,10	
	Long coef	-0,10	-0,05	-1,50	-0,01	121
	Short coef	0,10	0,05	1,50	0,01	

Prameň: Vlastné spracovanie.

Pri optimalizácii sme menili vždy hodnoty dvoch parametrov. Optimalizovaného AOS a na 3D grafe 6 môžeme sledovať, aký vplyv má zmena týchto hodnôt na výsledný ukazovateľ Ret/DD.



Graf 4 3D graf STR 41.5 optimalizovaných parametrov long coef a short coef

Prameň: Vlastné spracovanie.

Kvôli rozsahu práce sú kombinácie parametrov: PT a SL koeficient pre STR 21.5; long coef a short coef pre STR 21.5; PT a SL koeficient pre STR 41.5 uvedené v prílohe 7. Z grafických výsledkov optimalizovaných parametrov je zrejmé, že pri všetkých kombináciách nastavení si obidva AOS zachovali svoju ziskovosť. Konkrétne z grafu 6 môžeme pozorovať, že STR 41.5 vyhovuje skôr menšie hodnoty koeficientu pre umiestnenie Limit pokynu. Najhorší výsledok je pri kombinácii hodnôt parametrov -0,15 a 0,15 kedy dosahuje ukazovateľ Ret/DD hodnotu 13,98 a najlepší výsledok 18,2 je pri kombinácii hodnôt parametrov -0,13 a 0,1. Záverom konštatujeme, že nastavenia parametrov nie sú preoptimalizované a nachádzajú sa vo funkčnej oblasti.

ZÁVER

V prvej časti práce sme teoreticky vymedzili problematiku genetického programovania a zároveň sme priblížili problematiku obchodovania prostredníctvom automatických obchodných systémov.

V druhej časti práce sme na základe metodického postupu vytvorili AOS pre menový pár EURUSD na hodinovom časovom rámci. Pri zostavení metodického postupu sme vychádzali z vlastných skúseností so stavbou AOS v software SQ a tiež sme sa nechali inšpirovať: Pardo (2008), Myszkowski a kol. (2010), Deng a kol. (2012), Davey (2014), Fric, M. (2014). Pri výbere trhu pre stavbu AOS sme uprednostnili devízový trh pred akciovým alebo komoditným trhom z dôvodu vysokej likvidity, nízkych poplatkov, úzkemu kurzovému rozpätiu a možnosti obchodovať 24 hodín denne (nedeľa večer – piatok večer). V softwari StrategyQuant za použitia metódy genetické programovanie, sme vytvorili za časové obdobie 8 a pol roka (IS perióda) celkovú populáciu 10000 kandidátov, ktorí spĺňali výberové kritériá aj v OOS perióde. Spustili sme dve generovania, pri ktorých sme použili dve rôzne účelové funkcie fitness. Z kandidátov vygenerovaných metódou GP sme zistili, že priemerný AOS má menej ako 50 % úspešnosť ziskových obchodov. To znamená, že ziskové AOS sa vyznačujú pozitívnym risk reward ratio. Logika risk manažmentu priemerného AOS je teda v limitovaní rizika aj na úkor väčšieho počtu strát, ktoré sú kompenzované síce menším počtom ziskových obchodov, ale zato s viacnásobne väčším ziskom. Pri generovaní v druhom behu sme túto skutočnosť priamo zakomponovali do účelovej funkcie. Avšak celkové priemerné výsledky nepreukázali výrazne rozdielne hodnoty ukazovateľov medzi použitou prvou a druhou fitness funkciou. Skôr naopak, pri použití prvej fitness sme dosiahli (okrem ukazovateľa SQN[®]) celkovo lepších výsledkov v IS aj OOS perióde.

Keďže metóda genetického programovania sa snaží prostredníctvom optimalizácie cielene nájsť globálne optimum daného problému, tak touto metódou sa vystavujeme značnému riziku preoptimalizácie. Z toho dôvodu vytvorená populácia spĺňajúca hodnotiace kritériá, bola následne podrobená testom robustnosti. Testy robustnosti boli zostavené ako logicky na seba nadväzujúce kroky, ktoré začínali od časovo najmenej náročných testov až po časovo najnáročnejšie testy. Týmto postupom sme dosiahli hlavný cieľ práce, čo znamená, že sa nám podarilo vytvoriť dva rôzne AOS, ktoré boli natoľko robustné, že prešli všetkými testami. Z analýzy závislosti AOS na svojich parametroch sme zistili, že všetky tri analyzované AOS si dokážu udržať výkonnosť pri odstránení určitého parametra. STR 21.5 vykázala najstabilnejšie výsledky a bez výraznej zmeny si udržala výkonnosť pri jednotlivom odstránení všetkých analyzovaných parametrov.

ZOZNAM POUŽITEJ LITERATÚRY

Knižné publikácie

1. DAVEY, K. 2014. *Building algorithmic trading systems: a trader's journey from data mining to Monte Carlo simulation to live trading*. New Jersey : John Wiley & Sons Inc, 2014. 284 s. ISBN 978-1-118-77891-3.
2. HYNEK, J. 2008. *Genetické algoritmy a genetické programování*. Praha : Grada Publishing a.s., 2008. 187 s. ISBN 978-8024726953.
3. MACH, M. 2009. *Evolučné algoritmy: prvky a princípy*. Košice : Elfa s.r.o., 2009. 250 s. ISBN 978-80-8086-123-0.
4. THURP, V. 2008. *Van Tharp's Definitive Guide To Position Sizing*. USA :The Van Tharp Institute, 2008. 415 s. ISBN 978-0-935-21909-8.
5. PARDO, R. 2008. *The Evaluation and Optimization of Trading Strategies*. 2. vydanie. New Jersey : John Wiley & Sons Inc, 2008. 367 s. ISBN 978-0-470-12801-5.
6. POLI, R., LANGDON W., MCPHEE N. 2008. *A Field Guide to Genetic Programming*. UK : Lulu Enterprises, 2008. 250 s. ISBN 978-1-4092-0073-4.

Odborné časopisy, zborníky

1. BAILEY, D., BORWEIN, J., PRADO, M., ZHU, Q. 2014. Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-of-Sample Performance. In *Notices of the American Mathematical Society*. 2014. Číslo 6. s. 458–471. ISSN 1088-9477.
2. DEMPSTER, M. A. H., JONES, C. M. 2001. A real-time adaptive trading system using genetic programming. In *Quantitative Finance*. 2001. Číslo 1. s. 397-413. ISSN 1469-7688.

Internetové zdroje

1. BIS. 2016. Triennial Central Bank Survey. [online]. 2016. [cit. 2016-22-11]. Dostupné na internete: <<http://www.bis.org/publ/rpfx16fx.pdf>>.

2. CARR, M. 2016. Measure Volatility With Average True Range. [online]. 2016. [cit. 2017-11-02]. Dostupné na internete: <<http://www.investopedia.com/articles/trading/08/average-true-range.asp>>.
3. FOLGER, J. 2014. Backtesting And Forward Testing: The Importance Of Correlation. [online]. 2014. [cit. 2016-20-11]. Dostupné na internete: <<http://www.investopedia.com/articles/trading/10/backtesting-walkforward-important-correlation.asp>>.
4. FRIC, M. 2014. StrategyQuant: User's Guide. Verzia 3.8.0. [online]. 2014. [cit. 2016-24-10]. Dostupné na internete: <<http://www.strategyquant.com/download>>
5. MACHÁČEK, M. 2011. Genetické programovaní. In *Portál pre odborné publikovanie* [online]. 2011. [cit. 2016-20-12]. Dostupné na internete: <<http://www.posterus.sk/?p=10198>>.
6. MYSZKOWSKI, P., BICZ, A. 2010. Evolutionary Algorithm in Forex Trade Strategy Generation. In *Proceedings of the 2010 International Multi-Conference on Computer Science and Information Technology (IMCSIT)* [online]. 2010 [cit. 2016-25-12]. Dostupné na internete: <<http://ieeexplore.ieee.org/abstract/document/5679921/>>.
7. SEKAJ, I. 2009. Evolúcia v počítači alebo evolučné a genetické algoritmy. In *Portál pre odborné publikovanie* [online]. 2009. [cit. 2016-20-12]. Dostupné na internete: <www.posterus.sk/?p=3364&output=pdf>.
8. NESNÍDAL, T. 2011. Trading a genetické algoritmy - otázky a odpovedi. [online]. 2011. [cit. 2017-5-02]. Dostupné na internete: <http://www.financnik.cz/komodity/fin_home/trading-a-geneticke-algoritmy-otazky-odpovedi.html>.
9. DENG, S., SUN, Y., SAKURAI, A. 2012. Robustness test of genetic algorithm on generating rules for currency trading. In *Proceedings of the International Neural Network Society Winter Conference (INNS-WC2012)*. [online]. 2012 [cit. 2016-25-12]. Dostupné na internete: <<http://www.sciencedirect.com/science/article/pii/S1877050912007247>>.
10. ZAŇKA, Z. 2016. Jaká data používat při backtestu? [online]. 2016. [cit. 2017-10-02]. Dostupné na internete: <<http://finakademie.cz/jaka-data-pouzivat-pri-backtestu/>>.

PRÍLOHY

Príloha 1 Nastavenia zvolených stavebné bloky v softwari SQ

Main data for testing

Backtest Engine: Allow only engine symbols ☐

Symbol: Period: Available Data: 05.05.2003 - 30.12.2016

Start Date: End Date:

Out of Sample Period: In Sample: 01.01.2004 - 27.06.2012

☐ Reverse Out of Sample / In Sample Out Of Sample: 27.06.2012 - 30.06.2015

Test Parameters

Spread: Market Slippage: Min distance of order from price:

Test precision: ☐ Use real spread (only for real tick data)

...	Indicator / Name	Weight
☒	(EMA) Exponential Moving Average	1
☒	(SMA) Simple Moving Average	1
☒	(WMA) Weighted Moving Average	1
☒	(TEMA) Triple Exponential Moving Average	1
☒	(CCI) Commodity Channel Index	1
☒	(RSI) Relative Strength Index	1
☒	(MOM) Momentum	1
☒	(WILLPR) Williams % Range	1
☒	(MACD) MACD	1
☒	(STOCH) Stochastic	1
☒	(BBANDS) Bollinger Bands	1
☒	(KELTNER) Keltner Channel	1
☒	(ADX) Average Directional Movement Index	1
☒	(PSAR) Parabolic SAR	1
☒	(HIGHEST) Highest	1
☒	(LOWEST) Lowest	1
☐	(HR) Highest in range	1
☐	(LR) Lowest in range	1
☒	(LINREG) Linear Regression	1
☒	(QQE ADV) Qualitative Quantitative Estimation	1
☒	(ICHIMOKU) Ichimoku	1

☒	Open	1
☒	Close	1
☒	High	1
☒	Low	1
☐	Volume	1
☒	Heiken Ashi Open	1
☒	Heiken Ashi Close	1
☒	Heiken Ashi High	1
☒	Heiken Ashi Low	1
☒	Today Open	1
☒	This Bar Open	1
☒	Pivots	1
☒	Open D	1
☒	High D	1
☒	Low D	1
☒	Close D	1
☒	Price Ranges	
☒	(ATR) Average True Range	1
☒	(BAR_RANGE) Bar Range	1
☒	(TRUE_RANGE) True Range	1
☒	(PRICE_DIFF) Price Difference	1

Order Types & Exit Building Blocks			
...	Indicator / Name	Wei...	
☒	Enter at Stop	1	
☐	Enter at Limit	1	
☒	Stop Loss	1	
☒	Profit Target	1	
☒	Profit Trailing	1	
☒	Exit after X bars	1	
☒	Move Stop Loss to BE	1	
☒	Stop Trailing	1	

...	Indicator / Name	Weight
☒	Price Ranges	
☒	(ATR) Average True Range	1
☒	(BAR_RANGE) Bar Range	1
☒	(TRUE_RANGE) True Range	1
☒	(PRICE_DIFF) Price Difference	1
☒	(BBRANGE) Bollinger Bands Range	1
☒	(BBWR) Bollinger Bands Width Ratio	1
☒	Operators	
☒	(>) Greater	1
☒	(<) Lower	1
☒	(/) Crosses Above	1
☒	(\) Crosses Below	1
☒	(/) ClosesAbove	1
☒	(/) ClosesBelow	1
☐	(&&) Logical And	1
☐	() Logical Or	1
☒	(+) Addition	1
☒	(-) Subtraction	1
☒	(*) Multiplication	1
☒	(=) Equals	1
☒	(<>) Not Equals	1
☒	(X) Number Constant	1

Setting	Value
Fill Type	Default
Strict Stop Prices	☒
Pending Orders Valid One Bar	☐
Replace Pending Orders	☒
Maximum Trades Per Day	0
Maximum Total Trades	10000
Limit Signals to Time Range	☐
Time Range From	08:00
Time Range To	16:00
Exit At End Of Range	☐
Exit At End Of Day	☐
Exit On Friday	☐
Max Period for Indicators	100
Max Period for Price Patterns	20

Stop Loss & Profit Target Options	
SL is Required	☒
PT is Required	☒
SL/PT: ☒ in pips Minimum: 30 Maximum: 350 ☐ in money Minimum: 10 Maximum: 500 SL/PT in money is computed from trading of 1 standard lot/contract	
SL/PT based on ATR ☒ fixed pips ☒	
ATR Multiple Minimum: 0.5 Maximum: 5	
☐ Limit Risk Reward Ratio (SL vs PT) to a range from 1 : 1 to 1 : 1	
☐ Dont use price range indicators (ATR, Bar Range etc.) in conditions	

Príloha 2 Java kód Monte Carlo analýzy

```
package com.strategyquant.extend.MonteCarloMethods;

import java.util.Collections;

import com.strategyquant.lib.items.ComboBoxStrItemList;
import com.strategyquant.lib.language.L;
import com.strategyquant.lib.random.RandomGen;
import com.strategyquant.lib.results.SQOrder;
import com.strategyquant.lib.results.SQOrderList;
import com.strategyquant.lib.snippets.MonteCarloSim.MonteCarloMethod;

public class RandomizeTrades extends MonteCarloMethod {

    public RandomizeTrades() {
        setName(L.t("Randomize trades order"));

        ComboBoxStrItemList comboboxItemList = new ComboBoxStrItemList();
        comboboxItemList.addItem("Exact", L.t("Exact"));
        comboboxItemList.addItem("Resampling", L.t("Resampling"));

        addComboBoxParameter("Method", L.t("Method"), comboboxItemList);

        // set name of this method that will be displayed with parameter
        setFormattedName(L.t("Randomize trades order with {Method} method"));
    }

    /**
     * Function receives list of all orders sorted by open time and it could manipulate
     * the list and/or each order in any way to create new simulation case.
     * To skip the trade you have to remove them from the list.
     *
     * Order structure is available in the documentation here:
     * http://www.strategyquant.com/doc/api/com/strategyquant/lib/results/SQOrder.html
     *
     * @param originalOrders - list of original orders.
     */
    @Override
    public void compute(SQOrderList originalOrders) throws Exception {
        String method = getStringParameterValue("Method");

        if(method.equals("Exact")) { //Exact
            // only shuffles the order of trades
            Collections.shuffle(originalOrders);

        } else { //Resampling
            int sizeOfPctOrders = originalOrders.size();

            // copy randomly picked trades to temporary array
```

```

SQOrderList randomizedOrdersList = new SQOrderList();

for(int k=0; k<sizeOfPctOrders; k++) {
    SQOrder order = originalOrders.get(RandomGen.nextInt(sizeOfPctOrders));
    randomizedOrdersList.add(order.clone());
}

// clear original orders array
originalOrders.clear();

// add resampled orders to original array
for(SQOrder order : randomizedOrdersList) {
    originalOrders.add(order.clone());
}
}
}

```

Príloha 3 Zdrojový kód STR 21.5

Generated by StrategyQuant version 3.8.1

Generated at Sun Mar 12 10:11:00 GMT 2017

Tested on EURUSDM1, H1, 01.01.2004 - 30.12.2016

Spread: 1.5, Slippage: 0.0, Min distance of stop from price: 0.0

```
=====Entry
conditions
=====
```

```
LongEntryCondition = (Day of week <> Sunday)
```

```
ShortEntryCondition = (Day of week <> Sunday)
```

```
=====Entry
orders
=====
```

```
-- Long entry
```

```
if LongEntryCondition is true {
```

```
if No position is open then Buy at ThisBarOpen() + (0.5 * BBWidthRatio(66, 2.5)) Stop;
    Stop/Limit order expires after 9 bars.
```

```
    Stop Loss = (2.32 * ATR(41)) pips;
```

```
    Profit Target = 127 pips;
```

```
    // Move SL to BE (on close)
```

```
    Move Stop Loss to Entry price when in profit at least (4.43 * ATR(73)) pips;
```

```
    // Stop trailing (on close)
```

```
    Move Stop to (Ichimoku(9, 26, 52, Kijun-sen) + (0.2) * ATR(61))) on bar close;
```

```
}
```

```
-- Short entry
```

```
if ShortEntryCondition is true {
```

```
if No position is open then Sell at ThisBarOpen() + (-0.5 * BBWidthRatio(66, 2.5)) Stop;
    Stop/Limit order expires after 9 bars.
```

```
    Stop Loss = (2.32 * ATR(41)) pips;
```

```
    Profit Target = 127 pips;
```

```
    // Move SL to BE (on close)
```

```
    Move Stop Loss to Entry price when in profit at least (4.43 * ATR(73)) pips;
```

```
    // Stop trailing (on close)
```

```
    Move Stop to (Ichimoku(9, 26, 52, Kijun-sen) + (-0.2) * ATR(61))) on bar close;
```

```
}
```

```
===== Exit
orders
=====
```

```
-- Long exit
```

```
if MarketPosition is Long {  
  if (Bars Since Entry >= 24) {  
    Close position at market;  
  }  
  
}  
  
-- Short exit  
if MarketPosition is Short {  
  if (Bars Since Entry >= 24) {  
    Close position at market;  
  }  
  
}
```

Príloha 4 Zdrojový kód STR 41.5

Generated by StrategyQuant version 3.8.1

Generated at Sun Mar 12 10:10:00 GMT 2017

Tested on EURUSDM1, H1, 01.01.2004 - 30.12.2016

Spread: 1.5, Slippage: 0.0, Min distance of stop from price: 0.0

```
=====Entry
conditions
=====
LongEntryCondition = Short Term RSI(20) > 50
ShortEntryCondition = Short Term RSI(20) < 50
=====Entry
orders
=====
-- Long entry
if LongEntryCondition is true {
if No position is open then Buy at Highest(96) + (-0.1 * ATR(96)) Stop;
    Stop/Limit order expires after 238 bars.

    Stop Loss = (2.88 * ATR(60)) pips;
    Profit Target = 199 pips;

    // Move SL to BE (on close)
    Move Stop Loss to Entry price when in profit at least (0.8 * ATR(52)) pips;
    // Stop trailing (on close)
    Move Stop to (TodayOpen() + (0.6) * ATR(37))) on bar close;}

-- Short entry
if ShortEntryCondition is true {
if No position is open then Sell at Lowest(96) + (0.1 * ATR(96)) Stop;
    Stop/Limit order expires after 238 bars.

    Stop Loss = (2.88 * ATR(60)) pips;
    Profit Target = 199 pips;

    // Move SL to BE (on close)
    Move Stop Loss to Entry price when in profit at least (0.8 * ATR(52)) pips;
    // Stop trailing (on close)
    Move Stop to (TodayOpen() + (-0.6) * ATR(37))) on bar close;}
=====Exit
orders
=====
-- Long exit
if MarketPosition is Long {
if (Bars Since Entry >= 50) {
    Close position at market;}}
-- Short exit
if MarketPosition is Short {
if (Bars Since Entry >= 50) {
    Close position at market;}}
```

Príloha 5 Zdrojový kód STR 20.3

Generated by StrategyQuant version 3.8.1

Generated at Sun Mar 12 10:11:00 GMT 2017

Tested on EURUSDM1, H1, 01.01.2004 - 30.12.2016

Spread: 1.5, Slippage: 0.0, Min distance of stop from price: 0.0

=====

Entry conditions

=====

LongEntryCondition = MACD(12, 26, 9) > 0

ShortEntryCondition = MACD(12, 26, 9) < 0

=====

Entry orders

=====

-- Long entry

if LongEntryCondition is true {

if No position is open then Buy at Highest(55) Stop;

Stop/Limit order expires after 17 bars.

Stop Loss = (2.72 * ATR(66)) pips;

Profit Target = 230 pips;

// Profit trailing (on close)

Profit Trailing by (0.1 * ATR(87)) pips;

}

-- Short entry

if ShortEntryCondition is true {

if No position is open then Sell at Lowest(55) Stop;

Stop/Limit order expires after 17 bars.

Stop Loss = (2.72 * ATR(66)) pips;

Profit Target = 230 pips;

// Profit trailing (on close)

Profit Trailing by (0.1 * ATR(87)) pips;

}

===== Exit
orders

=====

-- Long exit

if MarketPosition is Long {

if (Bars Since Entry >= 63) {

Close position at market;

}

}

-- Short exit

if MarketPosition is Short {

if (Bars Since Entry >= 63) {

Close position at market;} }

Príloha 6 Charakteristika technických indikátorov v zdrojových kódach

Indikátory sledujúce trend

Indikátory sledujúce trend ako už z ich názvu vyplýva, slúžia na definovanie smeru trendu. Za najjednoduchší indikátor tohto typu je považovaný jednoduchý kĺzavý priemer (SMA). Vypočíta sa ako aritmetický priemer nasledovne:

$$SMA_{(T)} = \frac{1}{T} \sum_{t=1}^T P_t$$

kde P_t je zatváracia, prípadne iná cena v časovom okamžiku t , T je počet období (perióda), za ktorú sa SMA počíta. Aritmetický priemer sa nazýva kĺzavým preto, že pri pridaní novej hodnoty, posledná cena z výpočtu vypadáva a cena tak plynulo kľže ďalej.

Existujú rôzne variácie kĺzavého priemeru a jedným z nich je exponenciálny vážený kĺzavý priemer (EWMA). EWMA na rozdiel od SMA dáva väčšiu váhu novším cenám, pričom váhy klesajú exponenciálne smerom do minulosti. Výpočet EWMA môžeme formulovať ako:

$$EWMA_{(T)} = \frac{\sum_{t=1}^T \lambda^{T-t} P_t}{\sum_{t=1}^T \lambda^{T-t}}$$

kde pridávame parameter lambda λ , ktorý predstavuje vyhladzovací parameter a platí $0 < \lambda < 1$.

Indikátory volatility

Hlavným cieľom indikátorov volatility je zachytávať kolísavosť hodnôt časového radu. Medzi indikátory volatility zaradíme Bollingerove bands (BB) alebo Average True Range (ATR). Indikátor BB vymyslel John Bollinger v 80-tych rokoch 20. storočia. Indikátor je tvorený SMA s danou periódou, ktorý zachytáva hlavný trend časového radu. Sledovanie SMA je doplnené o hodnotu priemeru zvýšenú, resp. zníženú o dvojnásobok štandardnej odchýlky za dané obdobie. Hodnoty smerodajnej odchýlky určujú pásmo oscilácie okolo kĺzavého priemeru, čo znamená, že pohyb ceny sa dá očakávať v tomto pásme. Ak fluktuácia ceny je väčšia, tak pásmo sa rozšíri, ak menšia, tak pásmo sa zúži. ATR prvý krát predstavil W. Wilder v roku 1978. ATR vychádza z indikátora True Range (TR), ktorý Wilder vytvoril, aby bral do úvahy vznikajúce gapy na trhu a presnejšie meral volatility. TR sa vypočíta ako najvyššia hodnota z nasledujúceho vzťahu:

$$TR = \max(High_t - Low_t, \text{abs}(High_t - Close_{t-1}), \text{abs}(Low_t - Close_{t-1}))$$

kde $High$ je najvyššia a Low je najnižšia cena v čase t , $Close$ predstavuje zatváraciu cenu v čase $t-1$. ATR potom nie je nič iné ako exponenciálne vážený priemer indikátoru True Range za dané obdobie, resp. daný počet sviečok.

Relative strenght index (RSI)

RSI porovnáva veľkosť pozitívnych cenových zmien (rastúca cena) a negatívnych cenových zmien (klesajúca cena) a transformuje túto informáciu do čísla, ktoré leží v intervale od 0 do 100. Indikátor RSI vypočítame nasledovne:

$$RSI = 100 - (100 / (1 + RS))$$

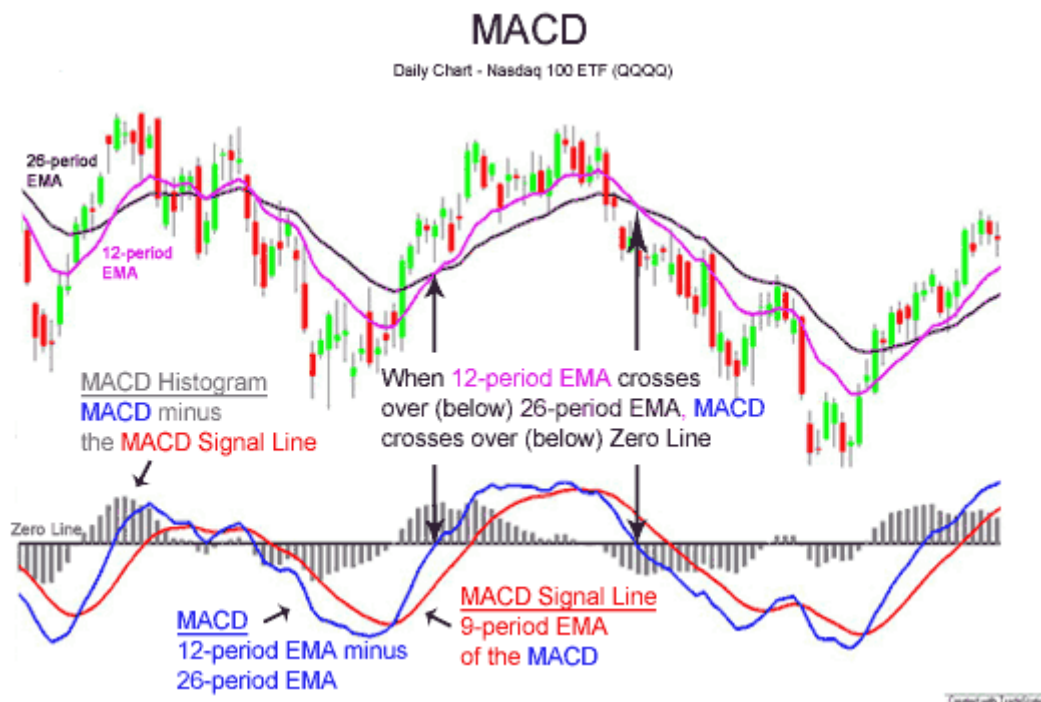
kde

$$RS = U_n / D_n$$

kde U_n je priemer vzrastu Close cien za n sviečok a D_n je priemer poklesu Close cien za n sviečok. Vzrastom a poklesom sa rozumie rozdiel medzi cenou Close novej sviečky a cenou Close predchádzajúcej sviečky. Ak novšia Close uzatvorila vyššie ako predchádzajúca Close rozdiel sa ráta do priemeru v čitateli RS. Ak novšia Close uzatvorila nižšie ako predchádzajúcu Close, ich rozdiel sa ráta do priemeru v menovateli. Priemer strát je vždy vyjadrený ako kladná hodnota. Využitie pri STR 41.5 je založené na prekročení neutrálnej línie indikátora RSI smerom nahor/nadol. RSI nad 50 znamená, že priemerné pozitívne cenové zmeny za n zvolených sviečok začali presahovať priemerné negatívne cenové zmeny za n zvolených sviečok. RSI pod 50 znamená, že priemerné negatívne cenové zmeny za n zvolených sviečok začali presahovať priemerné pozitívne cenové zmeny za n zvolených sviečok.

MACD (Moving Average Convergence/Divergence)

Ako z názvu vyplýva pri MACD sa sleduje zbíhavosť a rozbiehavosť - konvergencia a divergencia kľzavých priemerov. Hodnota ukazovateľa MACD sa vypočítava z rozdielu exponenciálnych kľzavých priemerov. Konkrétne sú to dva exponenciálny kľzavé priemery za rôzne časové obdobie a jeden kľzavý priemer, ktorý je vyhladením dvoch spomínaných línií. V prípade STR 20.3 je nastavenie 12/26/9 čo znamená, že od hodnoty EMA 12 sa odpočíta hodnota EMA 26, ak cena rastie tak $EMA\ 12 > EMA\ 26$ a hodnota MACD je kladná, naopak ak cena klesá. MACD teda ukazuje, či je trend klesajúci alebo rastúci. Hodnota 9 v nastavení MACD je opäť EMA, ktorá má iba informačný charakter na grafe a nevstupuje do výpočtu pri STR 20.3.

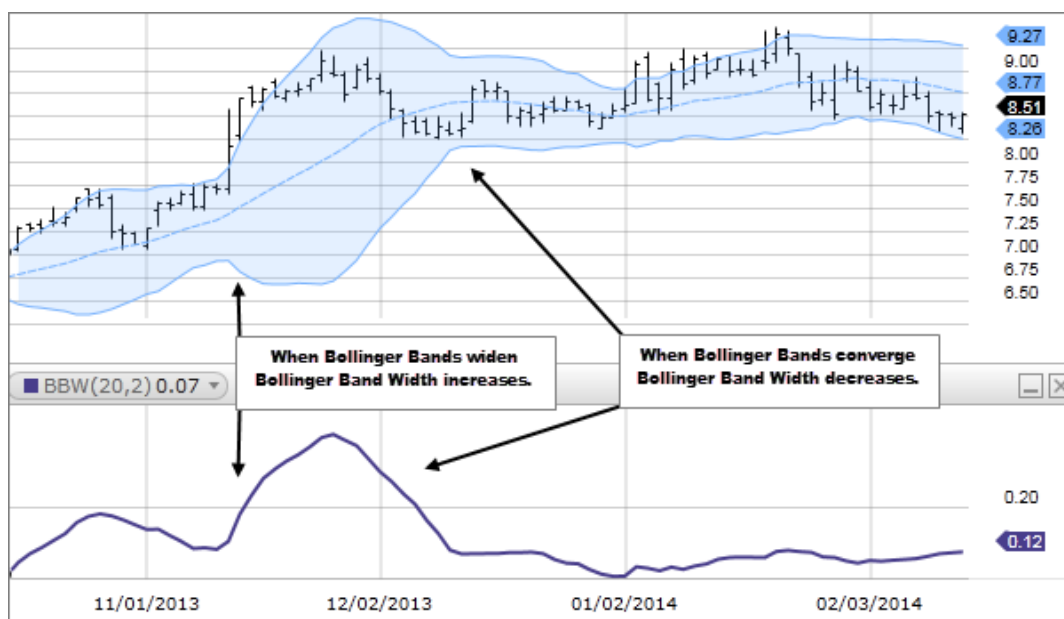


Bollinger Band Width ratio

Bollinger Band Width sa pri STR 21.5 využíva ako odchýlka pre zadanie pokynu Stop v sekundárnej podmienke. Bollinger Band Width používa rovnaké dva parametre ako Bollinger Bands: SMA (pre stredné pásmo) a násobok štandardnej odchýlky, ktorou je horné a dolné pásmo odsadené od stredného pásma. Výpočet je nasledovný:

$$\text{Band Width} = (\text{horné pásmo} - \text{dolné pásmo}) / \text{stredné pásmo}$$

Počas obdobia rastúcej volatility cien, sa bude vzdialenosť medzi dolným a horným pásmom rozširovať a Bollinger Band Width sa zvýši. A naopak, v období nízkej volatility trhu, sa vzdialenosť medzi dvoma zužuje a Bollinger Band Width sa zníži.



Ichikomo

Ichikomosa skladá zo 4 komponentov: Tenkan SEN, Kijun SEN, Kumo a Chikou Span zachytených na obrázku nižšie. V STR 21.5 je použitý iba Kijun SEN, ktorého hodnota slúži na určenie úrovne posuvného SL (trailing stop). Kijun SEN (modrá línia) sa vypočíta ako:

$$\text{Kijun SEN} = (\text{maximum za posledných 26 sviečok} + \text{minimum za posledných 26 sviečok}) / 2.$$


Ďalšie 3 komponenty (nie sú použité v žiadnom AOS) sa vypočítajú ako:

Tenkan SEN (červená línie) = (maximum za posledných 9 sviečok + minimum za posledných 9 sviečok) / 2.

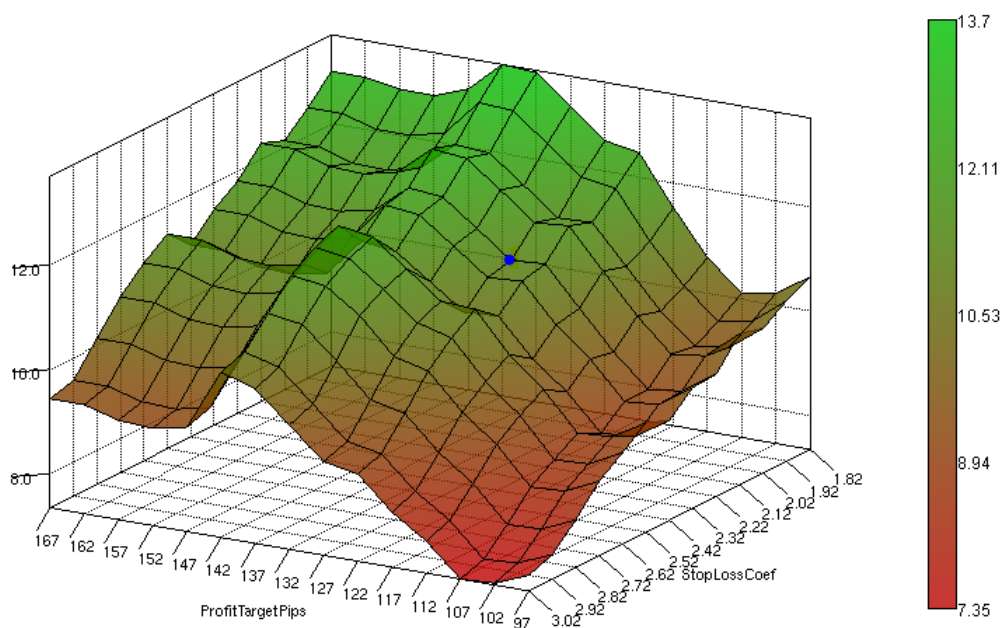
Kumo čiže oblak tvoria 2 úrovne, a to Senkou Span A a Senkou Span B.

Senkou Span A = Kijun sen + Tenkan Sen / 2

Senkou Span B = maximum za posledných 52 periód + minimum za posledných 52 periód / 2

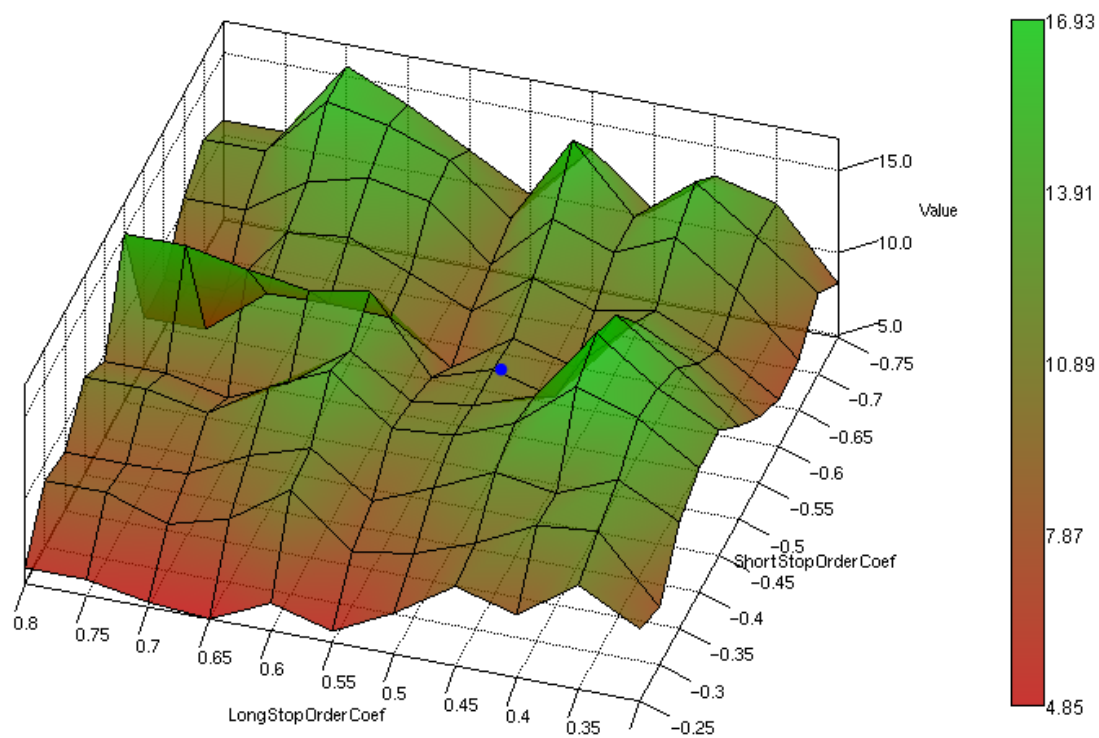
Chik Span nie je ničím iným ako premietnutie súčasnej ceny 26 sviečok naspäť.

Príloha 7 Grafické výsledky analýzy parametrov cez optimalizáciu



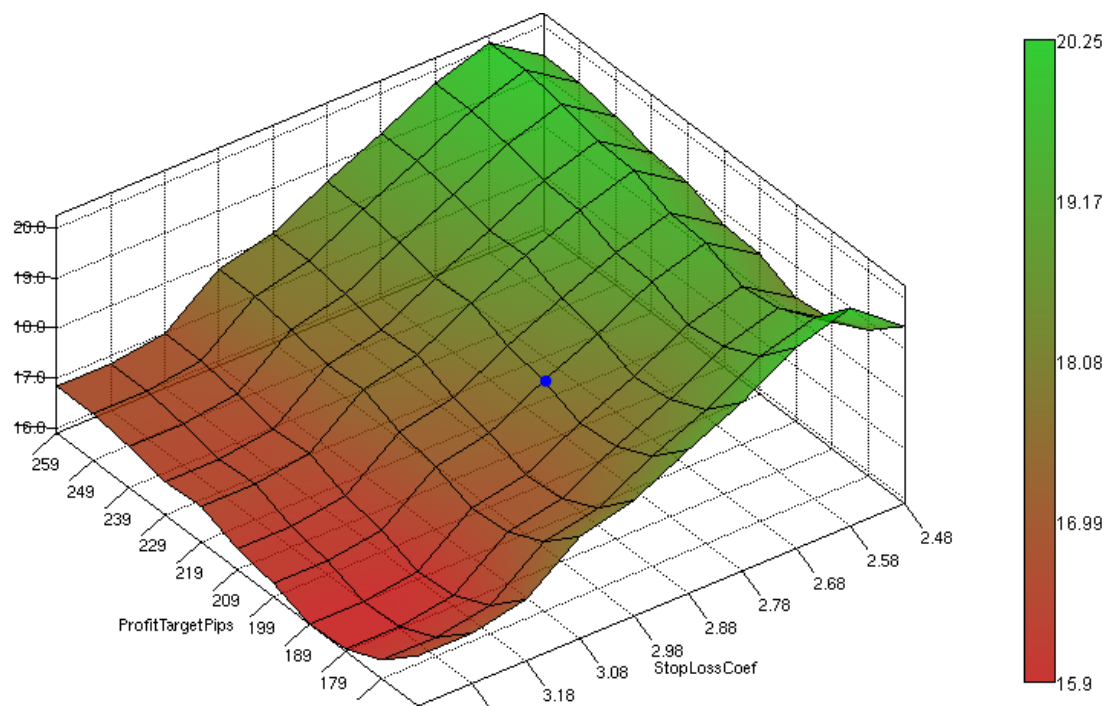
3D graf STR 21.5 optimalizovaných parametrov PT a SL koeficient

Prameň: Vlastné spracovanie.



3D graf STR 21.5 optimalizovaných parametrov coef long a coef short

Prameň: Vlastné spracovanie.



3D graf STR 41.5 optimalizovaných parametrov PT a SL koeficient

Prameň: Vlastné spracovanie.

Príloha 8 Hodnoty optimalizovaných parametrov pre WFM test

Optimization Results Options

Optimization method: Genetic Optimization Number of tests: 15000

Strategy Parameters

☐ Actions

Use	Name	Type	Original	Start	Stop	Step
☒	LongStopOrderCoef	Float	0.5	0.25	0.75	0.01
☒	LongStopTrailingCoef	Float	0.2	0.1	0.3	0.01
☐	OrderBarsValid	Integer	9	9	9	0
☒	ProfitTargetPips	Integer	127	92	162	5
☒	ShortStopOrderCoef	Float	-0.5	-0.25	-0.75	-0.01
☒	ShortStopTrailingCoef	Float	-0.2	-0.1	-0.3	-0.01
☐	StopLossATR	Integer	41	41	41	0
☒	StopLossCoef	Float	2.32	1.16	3.48	0.02
☐	pATR_7	Integer	61	61	61	0
☐	pBBWIDTHRATIO_2	Integer	66	66	66	0
☐	pBBWIDTHRATIO_3	Float	2.5	2.5	2.5	0.0
☐	pConsty_1	Float	1.0	1.0	1.0	0.0
☐	pICHI_KIJUN_4	Integer	9	9	9	0
☐	pICHI_KIJUN_5	Integer	26	26	26	0
☐	pICHI_KIJUN_6	Integer	52	52	52	0

Optimalizované parametre pre STR 21.5

Optimization Results Options

Optimization method: Genetic Optimization Number of tests: 15000

Strategy Parameters

☐ Actions

Use	Name	Type	Original	Start	Stop	Step
☒	LongStopOrderCoef	Float	-0.1	-0.05	-0.15	-0.01
☒	LongStopTrailingCoef	Float	0.6	0.3	0.9	0.01
☐	OrderBarsValid	Integer	238	238	238	0
☒	ProfitTargetPips	Integer	199	99	299	10
☒	ShortStopOrderCoef	Float	0.1	0.05	0.15	0.01
☒	ShortStopTrailingCoef	Float	-0.6	-0.3	-0.9	-0.01
☐	StopLossATR	Integer	60	60	60	0
☒	StopLossCoef	Float	2.88	1.44	4.32	0.03
☐	pATR_4	Integer	96	96	96	0
☐	pATR_5	Integer	37	37	37	0
☐	pConsty_2	Float	50.0	50.0	50.0	0.0
☐	pHIGHEST_3	Integer	96	96	96	0
☐	pLOWEST_3	Integer	96	96	96	0
☒	pRSI_1	Integer	20	20	50	5
☐	Fill Type	Select	Default	Default	Default	Default

Optimalizované parametre pre STR 41.5