

# RAZVOJ TRGOVALNEGA AGENTA Z JEZIKOM MQL4

MIHAILO BOŽOVIĆ, ROBERT LESKOVAR, BORUT WERBER

Univerza v Mariboru, Fakulteta za organizacijske vede, Kranj, Slovenija  
mihailo.bozovic@student.um.si, robert.leskovar@um.si, borut.werber@um.si

Diplomsko delo obravnava razvoj trgovalnega agenta z uporabo jezika MQL4, specifičnega za trgovanje na platformi MetaTrader 4. Naš cilj je bil ustvariti prototip agenta, ki bi samostojno analiziral gibanje cen med najpomembnejšimi valutnimi pari in izvajal trgovanje na podlagi določene strategije. Implementirali smo funkcije v MQL4, ki omogočajo agentu sprejemanje odločitev o nakupu in prodaji valutnih parov na podlagi različnih indikatorjev in analiz gibanja cen. Raziskava je vključevala tudi testiranje in optimizacijo uspešnosti v simuliranih trgovanjih. Obstoječa literatura se večinoma osredotoča na teoretične vidike algoritmičnega trgovanja, vendar pogosto primanjkuje konkretnih študij primerov in praktičnih implementacij v MQL4. Prav tako so številne raziskave usmerjene v splošne trgovalne strategije in funkcije, medtem ko je manj poudarka na prilagodljivosti algoritmov za specifične tržne pogoje in vplivu različnih optimizacijskih tehnik na njihovo dolgoročno uspešnost.

DOI  
[https://doi.org/  
10.18690/um.fov.2.2025.7](https://doi.org/10.18690/um.fov.2.2025.7)

ISBN  
978-961-286-963-2

**Ključne besede:**  
MQL4,  
denarne valute,  
agent,  
umetna inteligenca,  
MetaTrader4



Univerzitetna založba  
Univerze v Mariboru

DOI  
[https://doi.org/  
10.18690/um.fov.2.2025.7](https://doi.org/10.18690/um.fov.2.2025.7)

ISBN  
978-961-286-963-2

**Keywords:**  
MQL4,  
currency pairs,  
agent,  
artificial intelligence,  
MetaTrader5

# DEVELOPMENT OF TRADING AGENT USING MQL4 LANGUAGE

MIHAILO BOŽOVIĆ, ROBERT LESKOVAR, BORUT WERBER

University of Maribor, Faculty of Organizational Sciences, Kranj, Slovenia  
[mihailo.bozovic@student.um.si](mailto:mihailo.bozovic@student.um.si), [robert.leskovar@um.si](mailto:robert.leskovar@um.si), [borut.werber@um.si](mailto:borut.werber@um.si)

The thesis deals with the development of a trading agent using the MQL4 language, specific for trading on the MetaTrader 4 platform. Our goal was to create a prototype of an agent that would independently analyze price movements among the most important currency pairs and execute trades based on a specific strategy. We implemented functions in MQL4 that allow the agent to make decisions about buying and selling currency pairs based on various indicators and price movement analyses. The research also included testing and optimizing performance in simulated trades. The existing literature mainly focuses on theoretical aspects of algorithmic trading, but there is often a lack of concrete case studies and practical implementations in MQL4. Also, many studies are focused on general trading strategies and functions, while there is less emphasis on the adaptability of algorithms to specific market conditions and the impact of various optimization techniques on their long-term performance.



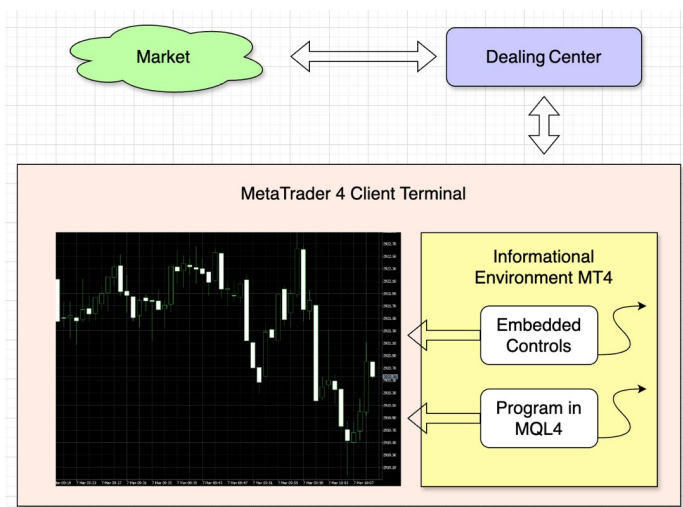
## 1 Algoritmsko trgovanje

Trgovalni agent predstavlja napredno programsko rešitev, ki z uporabo sofisticiranih algoritmov in funkcij avtomatizira trgovalne procese. Ti agenti temeljito analizirajo tržne podatke, prepoznavajo ključne vzorce in izvajajo transakcije z izjemno hitrostjo in natančnostjo. S tem omogočajo optimizacijo trgovalnih strategij, zmanjšanje tveganj ter povečanje dobičkonosnosti, kar jih postavlja v ospredje kot nepogrešljivo orodje za sodobne investitorje, ki si prizadevajo za učinkovitost in uspeh na finančnih trgih.

To delo je osredotočeno na razvoj trgovalnega agenta z uporabo programskega jezika MQL4 (MetaQuotes Language 4). MQL4 je specializiran programski jezik, namenjen razvoju trgovalnih sistemov, indikatorjev in skript za platformo MetaTrader 4 (v daljem tekstu MT4), eno izmed najbolj priljubljenih in razširjenih trgovalnih platform na svetu. Z njegovo pomočjo lahko ustvarimo zmogljive in prilagodljive trgovalne algoritme, ki so sposobni delovati v realnem času ter prilagajati se dinamičnim tržnim razmeram.

Algoritmsko trgovanje je tehnika trgovanja na finančnih trgih, ki za sprejemanje trgovalnih odločitev uporablja računalniške algoritme. V tem procesu računalniški programi analizirajo obsežno količino tržnih podatkov, vključno s preteklimi cenami, cenovnimi trendi, kazalniki tehnične analize, novicami in drugimi pomembnimi informacijami. Na podlagi teh podatkov in vnaprej določenih funkcijah algoritmi ustvarijo trgovalne signale in samodejno izvajajo trgovalne naloge. Glavna prednost algoritmskega trgovanja je njegova hitrost in sposobnost hitrega odzivanja na spremembe v tržnem okolju. Algoritmi lahko spremljajo in analizirajo gibanje cen v realnem času ter izvajajo transakcije brez odlašanja. Poleg tega so algoritmi programirani na podlagi posebnih pravil in strategij, kar zmanjšuje vpliv človeških čustev in subjektivnih odločitev pri trgovanju (Okonkwo, 2016).

Trgovalni agenti so lahko napisani v Pythonu, MQL4, Javi, MATLAB, R ali C++. To so najpogostejši jeziki. Algoritmsko trgovanje je način izvajanja 92 % uspešnih transakcij na borznem trgu (The Foreign Exchange Committee, 2016).



**Slika 1: Shema delovanja Metatrader4 platforme**

Vir: Lasten

MQL4, programska platforma v MT4, predstavlja ključno orodje za sodelovanje na trgu valut. Programski jezik, zasnovan za platformo MetaTrader 4, omogoča trgovcem razvijanje in implementacijo avtomatiziranih trgovalnih strategij, imenovanih "Expert Advisors" (EA), ali trgovalnih robotov.

MT4 ponuja obsežno funkcionalnost, vključno z dostopom do trgov valut, podporo, varnostnimi mehanizmi in možnostjo demo trgovanja. Demo računi omogočajo trgovcem preizkušanje strategij brez tveganja finančnih izgub, kar je ključno za razvoj in testiranje strategij z uporabo zgodovinskih podatkov (Linton M., 2012).

Programski jezik MQL4 omogoča razvoj avtomatiziranih trgovalnih strategij, ki temeljijo na tehnični analizi, indikatorjih in zgodovinskih podatkih. S tem lahko trgovalni agenti samodejno analizirajo tržne pogoje in izvajajo transakcije brez človeškega posredovanja, kar zagotavlja hitrejše odzivanje na tržne spremembe in boljši nadzor (Rechetin, 2014).

Spletna stran [book.mql4.com](http://book.mql4.com) je izčrpen vir za razvoj trgovalnih agentov, saj ponuja primere, vzorce, forume za izmenjavo znanja ter možnost najemanja freelancerjev za pomoč pri programiranju. Kljub temu je odgovornost za razvoj strategij na samih trgovcih, ki morajo imeti jasno predstavo o svojih ciljih in načrtih (Rechetin, 2014).

## 2 Razvoj agenta

Devizni trg vključuje več udeležencev, vključno z bankami, investicijskimi skladi, multinacionalkami, samostojnimi podjetniki in drugimi institucijami. Glavni cilj trgovanja na deviznem trgu je pridobiti dobiček z izkoriščanjem gibanj vrednosti valute. Imenovanje valutnih parov na trgu stranih valut sledi določenemu vzorcu, ki omogoča jasno identifikacijo obeh valut, vključenih v par.

**Tabela 1: Glavni valutni pari v borznem trgu**

EUR/USD	EUR/AUD
AUD/USD	EUR/NZD
NZD/USD	GBP/AUD
GBP/USD	GBP/CAD
USD/JPY	GBP/CHF
USD/CHF	GBP/NZD
USD/CAD	AUD/JPY
EUR/JPY	AUD/NZD
GBP/JPY	AUD/CAD
EUR/CHF	AUD/CHF
EUR/CAD	NZD/JPY
EUR/GBP	NZD/CAD
	NZD/CHF
	CHF/JPY
	CHF/CAD
	CAD/JPY

Vir: Lasten

Ko ustvarjamo trgovalnega agenta, je ključnega pomena, da se osredotočimo na funkcije, ki jih ustvarimo sami na podlagi našega poznavanja sveta borznega trgovanja. Medtem, ko ima vsak program nekaj standardnih funkcij, jih moramo pravilno uporabiti v naših lastnih funkcijah, da bodo smiselne. Uporabniško določene funkcije so bistvenega pomena, ko je treba strategijo trgovanja natančno definirati. Strategija trgovanja je načrt, ki ga trgovci uporabljajo za doseganje določenih ciljev trgovanja na Trgu stranih valut, uporabljajo pa različne strategije glede na njihov slog, cilje in toleranco do tveganja. Ko je strategija določena, je naslednji korak programiranje in prevajanje v algoritemski jezik, kot je MQL4.

```

bool IsTimeTrue() {
    datetime currentTime = TimeCurrent(); //trenuten čas programa (1 ura razlike od CET)
    int hour = TimeHour(currentTime);
    int minute = TimeMinute(currentTime);

    bool range1 = (hour == 3 && minute <= 60) || (hour == 4 && minute <= 37);

    bool range2 = ((hour == 9 || hour == 10) && minute <= 60) || (hour == 11 && minute <= 37);

    bool range3 = (hour == 15 && minute >= 30 && minute <= 60) || (hour == 16 && minute >= 0 && minute <= 60) ||
        (hour == 17 && minute >= 0 && minute <= 37);

    bool result = false;

    if ( range1 || range2 || range3) {
        result = true;
    }
    return result;
}

```

**Slika 2: Prikaz funkcije za preverjanje trenutnega časa, IsTimeTrue**

Vir: Lasten

Slika 2 prikazuje eno ključnih funkcij pri delovanju agenta. Ko ta funkcija vrne 'true', agent prejme signal, da lahko začne z delovanjem.

Trgovalni agent uporablja izjemno napreden algoritem, ki vključuje 223 funkcij, ki delujejo hkrati v zgolj 500 milisekundah. Ta kompleksnost omogoča izredno natančen in strog menedžment tveganj, kar posledično vodi v velike dobičke, ki presežejo morebitne izgube. Z optimalno razmerje med tveganjem in dobičkom, ki se giblje od 6 do 23, si lahko privoščimo tudi do pet neuspešnih transakcij zapored, saj bomo z vsakim naslednjim uspešnim poslom dosegli minimalen dobiček, ki nas bo v končni bilanci ohranil v plusu. Povprečna uspešnost transakcij v obdobju od oktobra 2022 do marca 2024 znaša 8.17, kar nakazuje stabilnost in učinkovitost strategije na dolgi rok. Ta visoka stopnja uspešnosti je rezultat temeljite analize tržnih podatkov in hitrega prilagajanja algoritma na spremembe v tržnih pogojih, kar omogoča ohranjanje stabilnosti in rasti v različnih tržnih scenarijih.

**Tabela 2: Parametri delovanja**

Parameter	Opis
Čas delovanja	Vnaprej določen
Odnos med tveganjem in dobičkom	Vsaj 1:6 za vsako uspešno transakcijo
Čas odločitve	10-500 milisekund

Vir: Lasten

### 3 Rezultati in SWOT analiza

Rezultati temeljijo na analizi zgodovinskih tržnih podatkov, ki so bili izbrani za valutni par zlato-ameriški dolar (XAUUSD). Za testiranje trgovalnega agenta smo določili ključne parametre, kot so časovni okvir trgovanja, obdobje optimizacije in uporabljeni indikatorji tehnične analize. Interpretacijo podatkov smo izvedli s pomočjo platforme TradingView, medtem ko načrtujemo uporabo knjižnice Backtrader v Pythonu za podrobnejšo analizo in natančnejši prikaz rezultatov. Uspešnost trgovalnega agenta smo ocenili na podlagi meritev, kot so donosnost strategije (ang. profit factor), maksimalna izguba (ang. Drawdown), razmerje med tveganjem in nagrado (ang. risk-reward ratio ali sharp ratio) ter odstotek uspešnih transakcij.

Meseca novembra 2023 je bilo izvedeno testiranje prototipa trgovalnega agenta na dokazano uspešni strategiji. Testiranje je potekalo na valutnem paru XAUUSD, kar pomeni, da je bila analiza usmerjena v ceno zlata v dolarjih. Rezultati analize prikazujejo, kako bi agent deloval, ko so vse funkcije agenta v celoti aktivirane in ko je agent povezan na virtualni računalnik preko Remote Desktop. Testiranje je omogočilo preverjanje delovanja agenta v realnih tržnih pogojih ter oceno njegovih sposobnosti za optimizacijo trgovinskih odločitev.



Slika 3: Prikaz rezultatov za mesec november v TradingView okolju

Vir: Lasten

Rezultati za prvi teden novembra so pokazali razmerje 8 : 33, kar pomeni, da je bilo 19 % uspešnih in 81 % neuspešnih transakcij. V tem tednu smo imeli 41 transakcij, od tega 8 uspešnih, s povprečnim dobičkom 9,375 tveganih enot, ter 33 neuspešnih, kar pomeni skupno 33 tveganih enot. Seštevek dobička in izgube je pokazal 42 tveganih enot dobička. Če je tvegana enota velikosti 0,2 % kapitala, to pomeni 8,4 % dobička v tem tednu.

V našem primeru SWOT analiza pokaže, da so moči trgovalnega agenta v avtonomiji, hitrosti in natančnosti, kar omogoča optimizacijo strategij in hitrejšo izkoriščanje tržnih priložnosti brez človeških napak. Šibkosti pa so povezane z omejenostjo strategij, potrebnim naprednim tehničnim znanjem ter tveganjem tehničnih napak, ki lahko vplivajo na uspešnost. Priložnosti vključujejo povečano učinkovitost trgovanja in dostopnost trga 24/7, medtem ko so nevarnosti predvsem sistemska tveganja, regulativne omejitve in konkurenca, ki lahko ogrozijo donosnost in varnost agenta.

**Tabela 3: SWOT analiza**

SWOT Analiza:	
MOČI	ŠIBKOSTI
Hitro delovanje	Omejen na MataTrader4
Brez emocionalnih ovir	Odvisen od podatkov
PRILOŽNOSTI	NEVARNOSTI
Integracija podatkov v strojno učenje	Volatilnost trga
Podpora večih platform	Restrikcije brokerjev

Vir: Lasten

## 4 Zaključek

Naša naloga potrjuje predpostavko, da uporaba trgovalnih algoritmov pripomore k večji natančnosti in hitrosti v primerjavi z ročnim trgovanjem, kar na koncu prinese večji dobiček. Osredotočili smo se na programsko okolje MQL4 in opisali postopek izgradnje ključnih funkcij, ki omogočajo delovanje trgovalnega agenta. Z uporabo uporabniško definiranih funkcij v programih MQL4 ali MQL5 lahko trgovalne strategije delujejo samodejno brez potrebe po stalnem nadzoru trga.



Pomembna ovira, ki smo se ji uspešno izognili, je vloga čustev pri trgovanju. Čustva lahko vplivajo na naše odločitve, kar je še posebej tvegano na finančnih trgih, kjer lahko celo najmanjše skrbi ali vznemirjenost vodijo do napak. Z uporabo algoritmov se lahko izognemo vplivu čustev in tako dosežemo bolj stabilne ter uspešne rezultate pri trgovanju.

Z analizo SWOT smo dokazali, da je uporaba trgovalnega agenta z uporabo jezika MQL4 lahko izjemno koristna za trgovce. S poudarkom na močeh, kot so avtonomija, hitrost, natančnost in optimizacija, ter priložnostih, kot so povečana učinkovitost, razvoj in prilagajanje ter dostopnost trgovine, smo pokazali, da lahko trgovalni agenti nudijo konkurenčno prednost v dinamičnem okolju finančnih trgov. Hkrati pa smo prepoznali tudi šibkosti, kot so omejenost strategij in tehnično znanje, ter nevarnosti, kot so sistemska tveganja in tveganje kraje intelektualne lastnine, ki zahtevajo ustrezno upravljanje in nadzor.

Kljub potrditvi prednosti uporabe trgovalnih algoritmov na podlagi MQL4 je treba upoštevati nekatere omejitve naše raziskave. Prvič, analiza temelji na določenem programskem okolju in specifičnih strategijah, kar omejuje njeno posplošitev na širši nabor trgovalnih pristopov ali platform. Drugič, naša raziskava ne upošteva vpliva zunanjih dejavnikov, kot so nenadne spremembe tržnih pogojev ali tehnične omejitve platforme, ki lahko vplivajo na delovanje algoritmov v realnih razmerah.

Za prihodnje raziskave bi bilo smiselno prevesti trgovalne strategije iz MQL4 v Python ter uporabiti knjižnico Backtrader za poglobljeno analizo in interpretacijo rezultatov. S tem bi omogočili večjo fleksibilnost pri razvoju strategij, boljšo vizualizacijo podatkov ter lažjo integracijo z naprednimi metodami strojnega učenja kar posledično poveča odnos med številom pozitivnih in negativnih transakcij.

## Opombe

Trgovalni agenti se soočajo s številnimi šibkostmi, kot so omejitve strategij, ki jih lahko prilagodijo le z napredno programiranjem, in tehničnimi napakami, ki lahko povzročijo nepričakovane izgube. Poleg tega so izpostavljeni nevarnostim, kot so napake v sistemih, regulativne omejitve ter naraščajoča konkurenca na trgu, kar lahko vpliva na njihovo donosnost. Prav tako obstaja tveganje kraje intelektualne lastnine, kar lahko ogrozi konkurenčno prednost in poslovno varnost trgovcev.

Avtor tega besedila ni finančni svetovalec, zato informacije v tem članku ne smejo biti razumete kot finančni nasveti. Pred vsako odločitvijo glede trgovanja ali investiranja se posvetujte s strokovnjakom.

**Literatura**

- Antonov, A. (2015). Trading systems: A new approach to system development and portfolio optimisation. Harriman House Limited.
- Linton, M. (2012). Expert advisor programming for MetaTrader 4: Creating automated trading systems in the MQL4 language. CreateSpace Independent Publishing Platform.
- Okonkwo, R. (n.d.). Full stack expert advisor programming for Meta Trader 4: Learn how to develop the perfect trading algorithm for gold/forex markets.
- Rechenthin, M. R. (2014). The FX robot method: Trading Forex for a living and getting results with real-world proof! CreateSpace Independent Publishing Platform.
- Schwager, J. D. (2012). Market wizards: Interviews with top traders. John Wiley & Sons.
- The Foreign Exchange Committee. (2016). Foreign exchange transactions: Execution to settlement recommendations for non-dealer participants.