VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ
FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY
INSTITUTE OF INFORMATICS

VYUŽITÍ UMĚLÉ INTELIGENCE JAKO PODPORY PRO ROZHODOVÁNÍ
THE USE OF ARTIFICIAL INTELLIGENCE FOR DECISION MAKING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE Katarína Nezbedová
AUTHOR

VEDOUCÍ PRÁCE prof. Ing. Petr Dostál, CSc.
SUPERVISOR

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA PODNIKATELSKÁ

Zadání bakalářské práce

Ústav:	Ústav informatiky
Studentka:	**Katarína Nezbedová**
Studijní program:	Kvantitativní metody v ekonomice
Studijní obor:	Matematické metody v ekonomice
Vedoucí práce:	**prof. Ing. Petr Dostál, CSc.**
Akademický rok:	2017/18

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 bakalářskou práci s názvem:

Využití umělé inteligence jako podpory pro rozhodování

Charakteristika problematiky úkolu:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Vymezení řešeného problému a stanovení celkového a dílčích cílů. Provedení teoretického popisu základů použité teorie prostředků umělé inteligence, popis a analýza problému, vyhodnocení současné situace, provedení návrhu řešení a zhodnocení přínosu návrhu řešení.

Základní literární prameny:

DOSTÁL, P. Advanced Decision Making in Business and Public Services. Brno: CERM, 2011. 168 s. ISBN 978-80-7204-747-5.

DOSTÁL, P. Pokročilé metody rozhodování v podnikatelství a veřejné správě. Brno: CERM, 2012. 718 s. ISBN 978-80-7204-798-7.

DOSTÁL, P. Soft computing v podnikatelství a veřejné správě - doplněk. Brno: CERM, 2016. 118 s. ISBN 978-80-7204-958-5.

HANSELMAN, D. a B. LITTLEFIELD. Mastering MATLAB. Upper Saddle River: Pearson Education International Ltd., 2012. 852 s. ISBN 978-0-13-185714-2.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2017/18

V Brně dne 28.2.2018

L. S.

doc. RNDr. Bedřich Půža, CSc. ředitel	doc. Ing. et Ing. Stanislav Škapa, Ph.D. děkan

Abstrakt

Táto bakalárska práca sa zaoberá problematikou Tamariho atraktoru a jeho aplikáciou pre formovanie predikčného modelu. Jadrom práce je vytvorenie programu na simuláciu, vo vývojovom prostredí MATLAB, a jeho využitie pre vytvorenie a porovnanie niekoľkých prípadových štúdií predikčného modelu založeného na odlišných parametroch. Tento model je zobrazený graficky a doplnený o ekonomickú interpretáciu.

Abstract

This bachelor thesis deals with the Tamari attractor problem and its application for forming a prediction model. The core of the work is to create a simulation program in the MATLAB development environment and to use it to create and compare several case studies of a predictive model based on different parameters. This model is graphically illustrated and supplemented by economic interpretation.

Kľúčové slová

umelá inteligencia, Tamariho atraktor, predikcia, teória chaosu, chaotický atraktor, MATLAB

Key words

artificial intelligence, Tamari attractor, prediction, chaos theory, chaotic attractor, MATLAB

Bibliografická citácia

NEZBEDOVÁ, K. *Využití umělé inteligence jako podpory pro rozhodování.* Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2018. 73 s. Vedoucí bakalářské práce prof. Ing. Petr Dostál, CSc..

Čestné prehlásenie

Prehlasujem, že predložená bakalárska práca je pôvodná a spracovala som ju samostatne. Prehlasujem, že citácie použitých prameňov sú úplné, že som vo svojej práci neporušila autorské práva (v zmysle Zákona č. 121/2000 Sb., o práve autorskom a o právach súvisiacich s právom autorským).

V Brne dňa 15. mája 2018

.....

Katarína Nezbedová

Poďakovanie

Rada by som sa poďakovala vedúcemu mojej bakalárskej práce prof. Ing. Petrovi Dostálovi, CSc. za odborné vedenie, cenné rady a praktické pripomienky k práci. Odborné vedenie pána profesora skvalitnilo vypracovanie práce a tiež mi dodalo veľa entuziazmu pri hľadaní potrebných zdrojov. Moje poďakovanie patrí aj mojej rodine a priateľom, ktorých psychická podpora mi v kritických momentoch veľmi pomohla.

Obsah

ÚVOD **10**

VYMEDZENIE PROBLÉMU A CIELE PRÁCE **11**

1 TEORETICKÉ VÝCHODISKÁ PRÁCE **12**

1.1 UMELÁ INTELIGENCIA 12

1.2 TEÓRIA CHAOSU 12

1.3 DIFERENCIÁLNE ROVNICE 13

1.3.1 Základné pojmy 13

1.4 ATRAKTORY 14

1.5 ZÁKLADNÉ EKONOMICKÉ POJMY 16

1.5.1 Elasticita 16

1.5.2 Hrubý domáci produkt (HDP) 18

1.5.3 Nezamestnanosť 19

1.5.4 Produktivita práce 21

1.5.5 Úrok 21

1.5.6 Menový kurz 22

1.6 MATLAB 23

1.6.1 Pracovné prostredie 23

1.6.2 GUI 25

2 ANALÝZA PROBLÉMU A SÚČASNEJ SITUÁCIE **27**

2.1 BEN TAMARI 27

2.2 TAMARIHO ATRAKTOR 28

2.2.1 Matematická interpretácia 28

2.2.2 Primárne komponenty 29

2.3 IDEA A VZNIK ATRAKTORU 30

3 VLASTNÉ NÁVRHY RIEŠENIA, PRÍNOS NÁVRHOV RIEŠENIA **33**

3.1 TAMARIHO ATRAKTOR V MATLABE 33

3.1.1 Tvorba pomocou GUI 33

3.1.2 Zdrojový kód 35

3.2 RIEŠENÁ PROBLEMATIKA 43

3.2.1 Prípadové štúdie Českej republiky 43
3.2.2 Prípadové štúdie porovnania viacerých vybraných krajín 49
3.2.3 Prínos riešenia 55

ZÁVER 56

ZOZNAM POUŽITÝCH ZDROJOV 57

ZOZNAM POUŽITÝCH OBRÁZKOV 59

ZOZNAM POUŽITÝCH TABULIEK 61

ZOZNAM PRÍLOH 62

PRÍLOHY I

Úvod

Umelá inteligencia sa stala každodennou súčasťou života dnešného človeka. Pôvodne bola vytvorená na základe inteligencie človeka, ale jej pokrok bol za posledné roky neopísateľný. Počítače dokážu kvalitnejšie a rýchlejšie vytvoriť predikciu budúcnosti. Predikcie sú užitočné, a preto veľmi žiadané, pretože ich môžeme využiť na prípravu nadchádzajúcej udalosti.

Z ekonomicko-matematickej oblasti nás predovšetkým zaujímajú predikcie týkajúce sa zmien vývoja ekonomiky krajín a hodnôt, ktoré uvedený vývoj v ekonomike ovplyvňujú. V tejto bakalárskej práci využijeme pre predikciu Tamariho atraktor, ktorý ešte nie je dostatočne rozšírený v povedomí ekonómov a analytikov. Tento atraktor využíva niekoľko parametrov, pomocou ktorých môžeme opísať úroveň ekonomiky jednotlivých krajín, vytvoriť predikciu na najbližšie roky a tiež pozorovať zmenu v správaní ekonomiky danej krajiny.

Využijeme teoretické poznatky o Tamariho atraktore pre vytvorenie jeho interaktívneho modelu. Pomocou štatistických tabuliek zistíme potrebné údaje na predikciu. Na základe modelu a endogénnych veličín navrhneme vlastné riešenie problematiky predikcie správania ekonomických v krajine. Toto riešenie môže byť potom využívané štátnymi inštitúciami, pre ďalšie študijné účely a podobne. Existuje nekonečne veľa variant skúmaní.

Vymedzenie problému a ciele práce

Cieľom tejto práce je predstavenie a rozšírenie povedomia o Tamariho atraktore a o jeho využiteľnosti v ekonomike. Pre tento účel bol vytvorený model Tamariho atraktoru, ktorý je využívaný pre simuláciu what-if analýzy, následne bol využitý na konkrétne prípadové štúdie.

Pre pochopenie problematiky Tamariho atraktoru bolo najskôr potrebné získať teoretický základ. Medzi základné poznatky, potrebné k porozumeniu skúmaného problému, patrí teória o atraktoroch, umelej inteligencii, oboznámenie so základnými ekonomickými pojmami, ktoré súvisia s Tamariho atraktorom a taktiež použiteľnosť programu MATLAB.

Ďalším krokom bolo stručné oboznámenie sa s históriou a vznikom Tamariho atraktoru. Podstatnou časťou bolo správne vytvorenie programu na simuláciu Tamariho atraktoru. Dôležitosť sa kládla na správne fungovanie programu, ale aj, aby bol program užívateľsky prívetivý.

Tieto poznatky boli následne využité na analýzu ekonomiky Českej republiky, pre ktorú bola vytvorená predikcia. Pre vytvorenie predikcie bolo potrebné preštudovať rozsiahlu databázu IFS, naučiť sa v nej orientovať a získať z nej potrebné dáta. Taktiež bolo poukázané na prípadné situácie, ktoré by mohli nastať, ak by prišlo k náhlym zmenám v krajine. Rovnako bola vytvorená predikcia pre vybrané krajiny, pričom sme poukázali na rozdiely v ekonomikách jednotlivých krajín. Zistené výsledky boli graficky a ekonomicky interpretované.

1 Teoretické východiská práce

V tejto časti si zadefinujeme nutné teoretické poznatky, z ktorých budeme vychádzať v praktickej časti. Začneme vymedzením pojmov z oblasti *umelej inteligencie*, predovšetkým problematikou, ktorá sa týka *teórie chaosu*. Kľúčovým bude zavedenie pojmov *diferenciálne rovnice* a *atraktory*, na ktorých bude pozostávať celá práca. Konkrétne chceme poukázať na ich využitie v ekonómii, ktoré budeme simulovať pomocou programu MATLAB. Pre lepšie pochopenie problematiky pridáme niekoľko príkladov.

1.1 Umelá inteligencia

Umelá inteligencia je vedecko-technická disciplína s mimoriadne rýchlym vývojom, ktorý začal v 50. rokoch minulého storočia. Aj napriek tomu, že je to pomerne mladá vedná disciplína, je už veľmi rozšírená.

Je to disciplína, ktorá nemá pevne vyhradený predmet a jednoliaty teoretický základ. Ide o súbor algoritmov a metód, pomocou ktorých získavame počítačové riešenia zložitých úloh. Tieto výsledky sa aplikujú v iných odboroch alebo tvoria nové, preto sa hovorí, že sa snaží dosiahnuť „stále sa pohybujúci cieľ“.

Mnoho autorov ju definuje rôzne. Napríklad podľa Minského rieši umelá inteligencia zložité príklady pomocou postupov, ktoré sú považované za znak prejavu inteligencie, ak sú používané človekom. Kotkova definícia hovorí, že je to vlastnosť systémov, ktoré sú schopné analyzovať javy, situácie, predmety a vzťahy medzi nimi, a tak tvoriť vnútorné modely sveta. Vďaka tejto definícií vieme rozdeliť teoretické úlohy umelej inteligencie na: riešenie úloh, rozpoznávanie, strojové učenie, kvalitatívne modelovanie, neurónové siete atď. [1].

1.2 Teória chaosu

Prvá zmienka o teórii chaosu bola v roku 1880, kedy Henri Poincaré zistil, že niektoré obežné dráhy troch telies môžu byť neperiodické. Neskôr sa o ňu zaujímal napríklad Eduard Lorenz, ktorý dokázal, že nie je možné predpovedať počasie na dlhšiu dobu dopredu.

Teória chaosu nám hovorí o nelineárnych systémoch, so skrytým rádom, ktoré sú zjavne riadené náhodnými javmi. Ak máme skutočný lineárny systém, vieme vytvoriť lineárny model, ktorý ho bude opisovať. Keďže ekonomické procesy sú nelineárne, dokážeme vytvoriť model, ktorý bude za ideálnych podmienok opisovať skutočný systém iba počas krátkej doby [2].

Mnoho javov, ktoré sa nám zdajú byť náhodné, môžu v skutočnosti obsahovať určitý rád, ktorý tieto náhodnosti ovplyvňuje. Napríklad náhodné množstvo ľudí na letisku je ovplyvňované rádom, ktorý je tvorený odletmi a príletmi. V rozdielnych odvetviach sú odlišné pôsobenia náhodnosti a rádov. V matematike prevláda pôsobenie rádu – dokážeme jednoznačne vyjadriť rovnicu. Na druhej strane, pri ekonomických javoch môže skôr prevládať náhodnosť – politické zásahy, prírodné katastrofy atď. [3].

1.3 Diferenciálne rovnice

Rovnice, v ktorých vystupuje ako neznáma funkcia jednej premennej a obsahujú deriváciu (prípadne deriváciu neznámej funkcie), nazývame obyčajné diferenciálne rovnice. Pomocou týchto rovníc môžeme popísať rôzne fyzikálne, matematické, technické a inžinierske problémy, ako sú napríklad voľný pád, Newtonov zákon ochladzovania, pohyb kyvadla, pružnosť a pevnosť materiálu, riadenie jadrových reakcií, lety do vesmíru a pod [4].

1.3.1 Základné pojmy

Definícia 1.1 Rád diferenciálnej rovnice je najväčší rád derivácie hľadanej funkcie v uvažovanej diferenciálnej rovnici.

Definícia 1.2 Lineárna diferenciálna rovnica je rovnica, ktorá je lineárna voči hľadanej funkcii a jej derivácie, značí sa LODR.

Obyčajné diferenciálne rovnice prvého rádu značíme ODR1 (lineárne LODR1), rovnice n-tého rádu značíme ODR$_n$ (lineárne LODR$_n$). Riešením ODR$_n$ je každá n-krát spojito derivovateľná funkcia na intervale I takom, že na tomto intervale, po dosadení funkcie a jej derivácií do danej rovnice, dostaneme identickú rovnosť. Pri riešení ODR hľadáme obecné, partikulárne a výnimočné riešenie.

Obyčajná diferenciálna rovnica, kde f je funkcia definovaná v G, čo je podmnožina euklidovského priestoru $\mathbb{R}^{n+1}$, má tvar

$$y^{(n)}(t) = f\left(t, y(t), y'(t), \dots, y^{(n-1)}(t)\right).$$

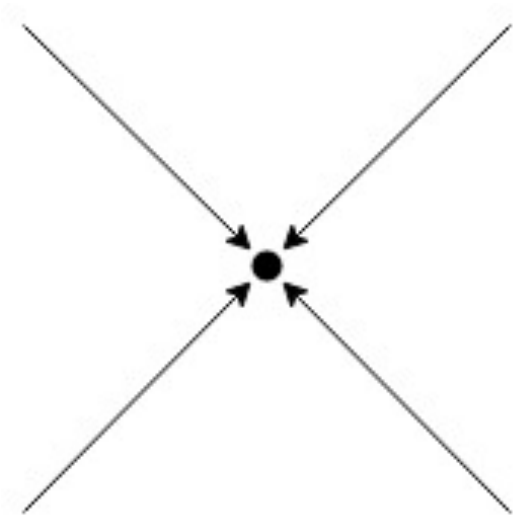
Takáto rovnica môže mať nekonečne mnoho riešení (ak nejaké existuje). Pre jednoznačnosť riešenia je potrebné doplniť podmienku [4].

1.4 Atraktory

Stavovú veličina, pomocou ktorej môžeme definovať systémovú rovnováhu, nazývame atraktor. Atraktor môže byť:

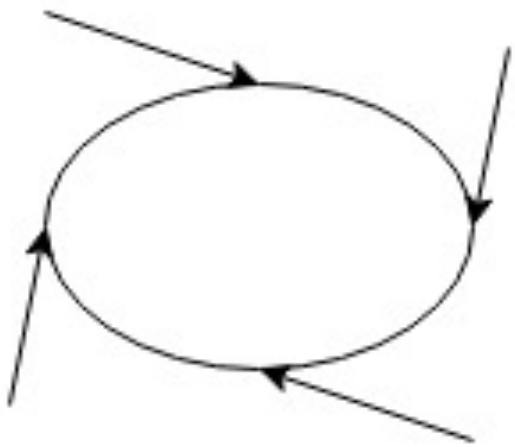
- bodový,
- cyklický,
- chaotický.

Príkladom bodového atraktoru je kyvadlo v stálej polohe, kde tvorí rovnováhu bod. Jeho grafické znázornenie môžeme vidieť na obrázku 1.



Obrázok 1: Bodový atraktor
(Zdroj: vlastný)

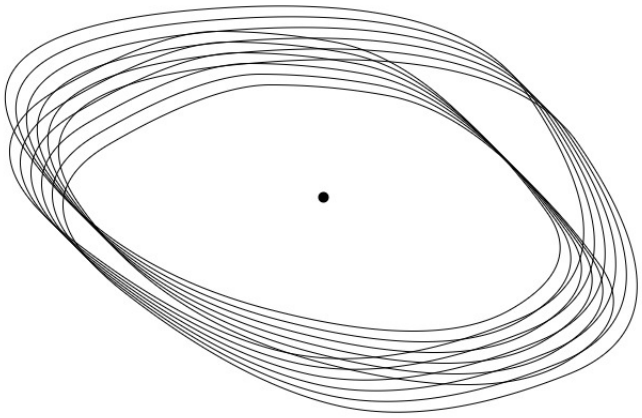
Pri cyklickom atraktore dostáva kyvadlo energiu, ktorá zabezpečuje jeho pohyb okolo stáleho bodu, ale rovnováhu predstavuje limitný cyklus. Cyklický atraktor je zobrazený na obrázku 2.



Obrázok 2: Cyklický atraktor

(Zdroj: vlastný)

Nakoniec, na obrázku 3, pri chaotickom atraktore, budeme kyvadlu dodávať energiu nepravidelne, čím sa dopracujeme k tzv. dynamickej rovnováhe, ktorá je tvorená určitou oblasťou [3].

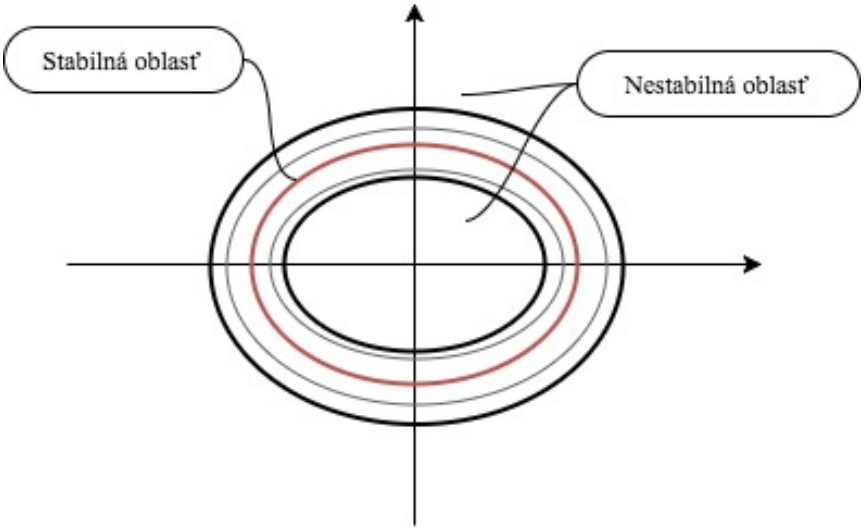


Obrázok 3: Chaotický atraktor

(Zdroj: Vlastný)

Ekonomické systémy dokážeme popísať chaotickým atraktorom. Stabilnú oblasť tvorí pásmo medzi vnútornou a vonkajšou elipsou. V tejto časti je ekonomický systém rovnovážny. Aj napriek občasným výchylkám, ku ktorým dôjde na základe zmeny ekonomického systému, je opätovne trajektória priťahovaná k prostrednej elipse. Tieto malé zmeny môžu byť dôsledkom zmien v ekonomike niektorých štátov, výrazné zmeny by mohli mať za následok kolaps systému (zrútenie ekonomiky štátu). Grafické

zobrazenie chaotického atraktoru v ekonomickom systéme je možné pozorovať na obrázku 4 [3], [5].



Obrázok 4: Chaotický atraktor v ekonomickom systéme

(Zdroj: Vlastný)

1.5 Základné ekonomické pojmy

V praktickej časti použijeme pre výpočet niekoľko ekonomických činiteľov, ktoré si teraz zadefinujeme.

1.5.1 Elasticita

Definícia 1.3 Elasticita, alebo cenová pružnosť, vyjadruje vzťah medzi zmenou ceny za tovar a zmenou množstva dopytu alebo ponuky vzťahujúcemu sa k tomuto tovaru. Zaoberá sa reakciou dopytu a ponuky na trhu po zmene ceny, čo následne ovplyvňuje sklon krivky dopytu a krivky ponuky.

Mieru tejto skúmanej reakcie vieme vyjadriť pomocou koeficientu cenovej elasticity. Tento koeficient nám udáva, o koľko percent sa zvýši alebo zníži dopyt alebo ponuka daného množstva, ak sa cena zvýši alebo zníži o 1%.

Koeficient cenovej elasticity delíme na:

- koeficient cenovej elasticity dopytu E_D, ktorý vyjadríme ako $\frac{percentuálna\ zmena\ dopytu\ množstva}{percentuálna\ zmena\ ceny}$,

- koeficient cenovej elasticity ponuky E_S, ktorý vyjadríme ako $\frac{percentuálna\ zmena\ ponuky\ množstva}{percentuálna\ zmena\ ceny}$.

Pre ich výpočet využívame nasledujúci vzorec

$$E_{D,S} = \frac{Q_2 - Q_1}{P_2 - P_1} \times \frac{P_1 + P_2}{Q_1 + Q_2},$$

kde P_1 je pôvodná cena, P_2 nová cena, Q_1 predstavuje pôvodnú veľkosť dopytovaného alebo ponúkaného množstva a Q_2 množstvo po zmene ceny. Koeficient cenovej elasticity dopytu vyjadrujeme v absolútnej hodnote, pretože jeho výsledok dostaneme záporný, ak sa chovanie dopytu zhoduje so zákonom klesajúceho dopytu.

Hodnota koeficientu nám udáva, že cenová elasticita je:

- absolútne nepružná - $E_{S,D} = 0$, zmena ceny nevyvolá zmenu množstva,
- nepružná - $0 < E_{S,D} < 1$, percentuálna zmeny ceny zapríčiní menej ako jednopercentnú zmenu v množstve,
- jednotkovo pružná - $E_{S,D} = 1$, percentuálna zmena ceny vyvolá percentuálnu zmenu v množstve,
- pružná - $1 < E_{S,D} < \infty$, je zapríčinená väčšia ako jednopercentná zmena v množstve,
- absolútne pružná - $E_{S,D} = \infty$, znamená, že bude vyvolaná nekonečne veľká zmena v množstve.

Elasticita dopytu je ovplyvnená mnohými faktormi. Jedným z nich je význam tovaru pre spotrebiteľa. Zvyšovaním významu sa znižuje elasticita. Ďalej je to existencia a dostupnosť substitútov, čím je ich viac, tým je vyššia elasticita dopytu. Z časového hľadiska je ťažké dostať sa k substitútom v krátkom období, preto dlhšia časová postupnosť vedie k elastickejšiemu dopytu.

Z pohľadu faktorov, ktoré ovplyvňujú elasticitu ponuky, je to z časového hľadiska podobné. V dlhšom časovom období je elasticita ponuky vyššia, pretože v krátkom období je zvýšenie ponúkaného množstva obmedzené. Dostatočné množstvo zásob zvyšuje elasticitu ponuky, čo umožňuje rýchle zvýšenie výroby. Existencia voľných zdrojov a ich dostupnosť zvyšuje elasticitu, pretože umožňuje produkovať väčšie množstvo tovaru. Taktiež medzi tieto faktory patrí technologická náročnosť výroby,

pričom produkty náročnejšie na výrobu z technologického hľadiska sú menej cenovo elastické [6].

1.5.2 Hrubý domáci produkt (HDP)

Definícia 1.4 Hrubý domáci produkt je celková peňažná hodnota všetkých finálnych statkov a služieb vyprodukovaných za dané obdobie (najčastejšie jeden rok) výrobnými faktormi na území daného štátu (v národnom hospodárstve) bez ohľadu na to, či vlastnia výrobné faktory občania štátu, alebo cudzinci.

HDP môže za dané obdobie nadobudnúť rôzne hodnoty. Záleží, akým spôsobom ho vyjadríme, poznáme tri typy:

- Nominálny HDP:
    - HDP vyjadrené v tržných cenách každého roku,
    - produkcia štátu je oceňovaná tržnými cenami,
    - zahŕňa vplyv inflácie.
- Reálny HDP:
    - predstavuje nominálny HDP očistený od vplyvu inflácie,
    - fyzický produkt oceňuje stálymi cenami alebo cenami východiskového roku,
    - umožňuje porovnávať skutočný rast produkcie ekonomiky.
- Potenciálny HDP:
    - predstavuje úroveň výstupu ekonomiky pri plnom a efektívnom využívaní výrobných faktorov.

HDP môžeme vypočítať tromi odlišnými metódami, od ktorých získame výsledok s drobnými rozdielmi, ktoré nastanú kvôli komplexnosti zdrojov pre výpočet. Týmito metódami sú:

- Produkčná metóda:
    - HDP získame ako súčet hodnôt pridaných spracovaním nakúpených polotovarov a materiálu vo všetkých odvetviach národného hospodárstva,
    - HDP je suma predajov na strane finálnej ponuky, od ktorej odčítame hodnoty medziproduktov.

- Výdajová metóda:
    - HDP je predstavený ako súčet výdajov jednotlivých ekonomických subjektov (hrubé domáce výdaje),
    - predstavuje úroveň skutočného agregátneho dopytu za určitý rok,
    - HDP = osobné výdaje na spotrebu + hrubé domáce investície + vládne výdaje na nákup produktov a služieb + čistý export.
- Dôchodková metóda:
    - HDP je súčet jednotlivých dôchodkov a nedôchodkových položiek (odpisy a nepriame dane),
    - HDP = hrubé mzdy + čisté úroky + renta z pôdy a nehnuteľností + zisky firiem (vrátane dividend) + amortizácia + nepriame dane.

Pomocou HDP môžeme porovnávať výkony ekonomík rôznych krajín. Po prepočte HDP na niektorú zo svetových mien, ako je USD alebo EUR, získame ukazovateľ ekonomickej sily. Tento ukazovateľ neudáva skutočnú ekonomickú vyspelosť jednotlivých krajín, pretože väčšie krajiny dosahujú vyššie hodnoty. Ďalším je ukazovateľ ekonomickej úrovne, ktorý získame po vydelení HDP počtom obyvateľov danej krajiny. Vyjadruje priemerné množstvo produktu na jedného obyvateľa, presnejšie vypovedá o vyspelosti krajiny a kvality životnej úrovne.

Hrubý domáci produkt nepredstavuje ideálny ukazovateľ, pretože má veľa nedostatkov. Nezachytáva produkciu domácej práce, transakcie neprechádzajúce legálnym trhom, neplatenú produkciu, tieňovú ekonomiku, kvalitu produktov a kvalitatívne zmeny produkcie výrobkov a služieb, čo predstavuje podhodnotenie HDP. Naopak nadhodnotenie HDP je predstavené tým, že zachytáva negatívne externality a výdaje, ktoré sú výsledkom škôd. Taktiež neberie do úvahy zmeny v životnom štýle a lokálne rozdiely v kvalite výrobkov a služieb [6].

1.5.3 Nezamestnanosť

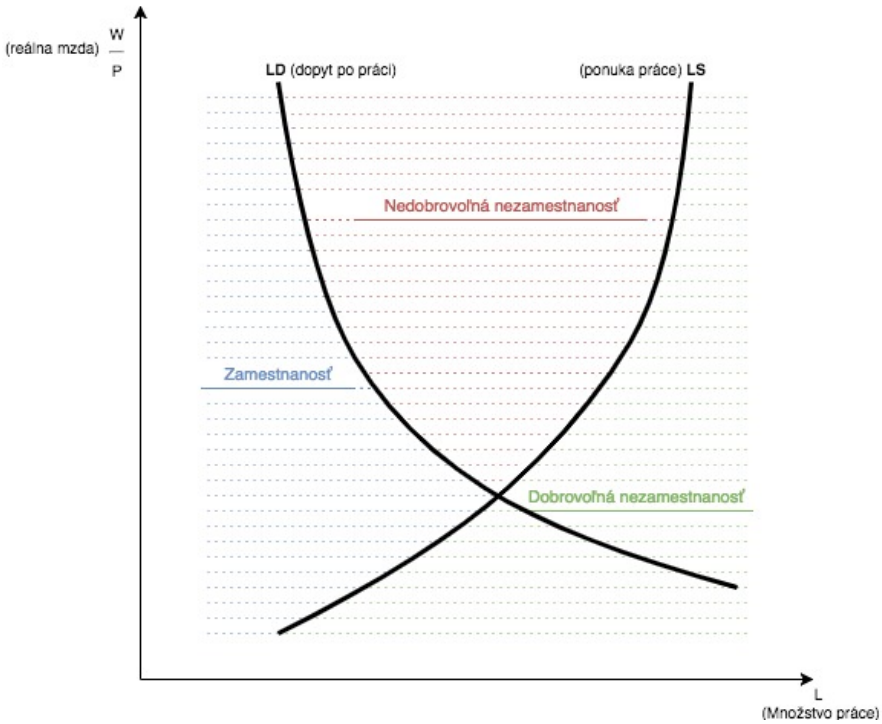
Definícia 1.5 Za nezamestnaného sa považuje osoba splňujúca tri podmienky:

- je staršia ako 15 rokov,
- za posledné štyri týždne si aktívne hľadala prácu, alebo si našla zamestnanie, do ktorého nastúpi v budúcich troch mesiacoch,

- do 14 dní je pripravená nastúpiť do zamestnania.

Ak nespĺňa aspoň jednu z podmienok, potom je zamestnaná, alebo ekonomicky neaktívna (študenti, deti, dôchodcovia atď).

Základné rozdelenie nezamestnanosti je na dobrovoľnú a nedobrovoľnú, jej znázornenie môžeme pozorovať na obr. 5. Pri dobrovoľnej nezamestnanosti si osoba nevie nájsť prácu, pretože požaduje vyššiu mzdu ako je tá, ktorá prevláda na trhu práce. Počas nedobrovoľnej nezamestnanosti by osoba akceptovala aj nižšiu mzdu, ako prevláda na trhu práce, ale aj tak si nevie prácu nájsť – nedostatočný dopyt po práci.



Obrázok 5: Nezamestnanosť

(Zdroj: Vlastné spracovanie podľa: 6)

Ďalšie delenie nezamestnanosti:

- podľa príčiny vzniku:
    - frikčná – spôsobená pohybom ľudí medzi zamestnaniami a oblasťami,
    - štrukturálna – dôsledkom zmien v štruktúre národného hospodárstva,
    - cyklická – dôvodom je nízky dopyt po práci,
    - sezónna – je krátkodobá, príčinou sú sezónne zamestnania,
    - technologická – zariadenia nahrádzajú ľudskú silu,

- podľa vekovej štruktúry – absolventi, preddôchodkový vek, atď.,
- podľa dĺžky – dlhodobá, strednodobá, krátkodobá,
- registrovaná nezamestnanosť – zahŕňa iba registrovaných na úrade práce,
- skutočná nezamestnanosť – odhad registrovaných na úrade práce, aj neregistrovaných,
- podľa pohlavia, profesie, vzdelania, atď.

Nezamestnanosť vyjadrujeme pomocou ukazovateľa miery nezamestnanosti u:

$$u = \frac{U}{L} = \frac{U}{E + U} \times 100\ [\%],$$

kde U predstavuje počet nezamestnaných, E počet zamestnaných a L celkový počet pracovných síl [6].

1.5.4 Produktivita práce

Produktivita práce vyjadruje účinnosť ľudskej práce vynaloženej pri tvorbe užitočných statkov. Je vyjadrená pomocou množstva týchto statkov pripadajúcich na jednotku vynaloženej práce. Rast produktivity práce je kľúčovým faktorom, preto je jeho zvyšovanie prioritou pre veľa podnikov.

Je to údaj, ktorý sa dá zmerať v prepočte na jedného zamestnanca a zároveň ho sledovať aj z pohľadu celého podniku alebo na jednotlivých oddeleniach. Na jej vyjadrenie využívame ukazovatele produktivity práce:

- mzdová produktivita – vyjadruje, aká časť výnosov pripadá na 1 korunu miezd, $mzdová\ produktivita = \frac{pridaná\ hodnota}{mzdové\ náklady}$,
- produktivita práce – vyjadruje, aká časť výnosov pripadá na jedného zamestnanca, $produktivita\ práce = \frac{pridaná\ hodnota}{počet\ zamestnancvo}$ [7].

1.5.5 Úrok

Definíciu úroku vieme rozobrať z dvoch uhlov pohľadu. Z hľadiska veriteľa je to odmena za požičaný kapitál. Počas pôžičky bol vystavený riziku v podobe straty kontroly nad požičanými peniazmi, ich hodnota klesla kvôli inflácii a pod. Z hľadiska dlžníka je to cena, ktorú musí zaplatiť za požičaný kapitál.

Úrok najčastejšie vyjadrujeme z hodnoty požičaného kapitálu v podobe úrokovej miery (v percentách). Veľkosť úrokovej miery záleží na mnoho činiteľoch, medzi ktoré patria:

- Facilitné sadzby – sú to úrokové miery, ktoré používa centrálna banka pri pôžičkách obchodným bankám. Patrí sem diskontná sadzba, lombardná sadzba a repo sadzba.
- Medzibankové úrokové miery – úrokové miery využívané za úvery poskytované medzi obchodnými bankami.
- Stratégie banky – závisí na rozdiele medzi úrokovými mierami úverov a vkladov. Napríklad stratégiou novej banky môže byť lákanie zákazníkov na vyššiu úrokovú mieru za vklad.
- Riziko pôžičky – s rastom rizika rastie aj úroková miera. Z toho dôvodu napríklad tzv. bezrizikové investície, ako sú cenné papiere, vynášajú nízky úrok.
- Doba pôžičky – napríklad pri raste termínovaného vkladu rastie aj úroková miera, z dôvodu zohľadnenia vyššej odmeny pri nasledujúcom vklade.
- Výška požičaného kapitálu – pri väčšom množstve požičaného kapitálu je vyššia úroková miera.
- Daňová politika štátu [8].

1.5.6 Menový kurz

Definícia 1.6 Menový kurz predstavuje cenu jednej meny vyjadrenú v jednotkách inej meny. Udáva sa ako podiel domácej meny k zahraničnej mene.

Menové kurzy sa vytvárajú na menovom trhu podľa pôsobenia dopytu a ponuky. Na menovom trhu sa kupuje jedna mena za druhú, podľa toho, či je daná mena oslabená alebo posilnená. To sa odvádza od rastu alebo poklesu nákupu danej meny.

Menové kurzy delíme na:

- voľne pohyblivé – vytvárajú sa na základe pôsobenia dopytu a ponuky, v tejto dobe sa tento systém už nevyužíva, pretože môže znížiť ekonomickú stabilitu štátu,

- riadené pohyblivé – vytvárajú sa na základe pôsobenia dopytu a ponuky a zároveň občasnými intervenciami zo strany centrálnej banky, za účelom usmernenia vývoja kurzu,
- fixné – úradne stanovené kurzy, na ktoré centrálna banka pôsobí intervenciami, podľa predčasne daného režimu, aby boli stanovené kurzy udržané na určitej hladine.

Momentálne je najpoužívanejší systém riadených pohyblivých kurzov, tzv. dirty floating. Vláda alebo centrálna banka ovplyvňuje ich vývoj pomocou nakupovania alebo predávania svojej meny, tzv. intervencie na devízovom trhu. Ak by štát využíval systém fixných kurzov, musí vybrať referenčnú menu, ktorou môže byt jedna mena alebo sa zvolí niekoľko mien z menového koša, ku ktorej sa stanoví fixný kurz. Výhodou fixných kurzov je, že vytvárajú stabilné prostredie pre zahraničné investície a pomáhajú udržiavať nízku infláciu. Naopak, medzi nevýhody patrí riziko útoku na menu, rozdiel medzi nominálnym a reálnym menovým kurzom, neustále zásahy centrálnej banky a pod.

Aktuálny menový kurz na kurzovom lístku je nominálny. Reálny kurz predstavuje pomer kúpnych síl mien. Jeho vzťah k nominálnemu kurzu je taký, že ho spája s pomerom cenovej hladiny doma a v zahraničí [6].

1.6 MATLAB

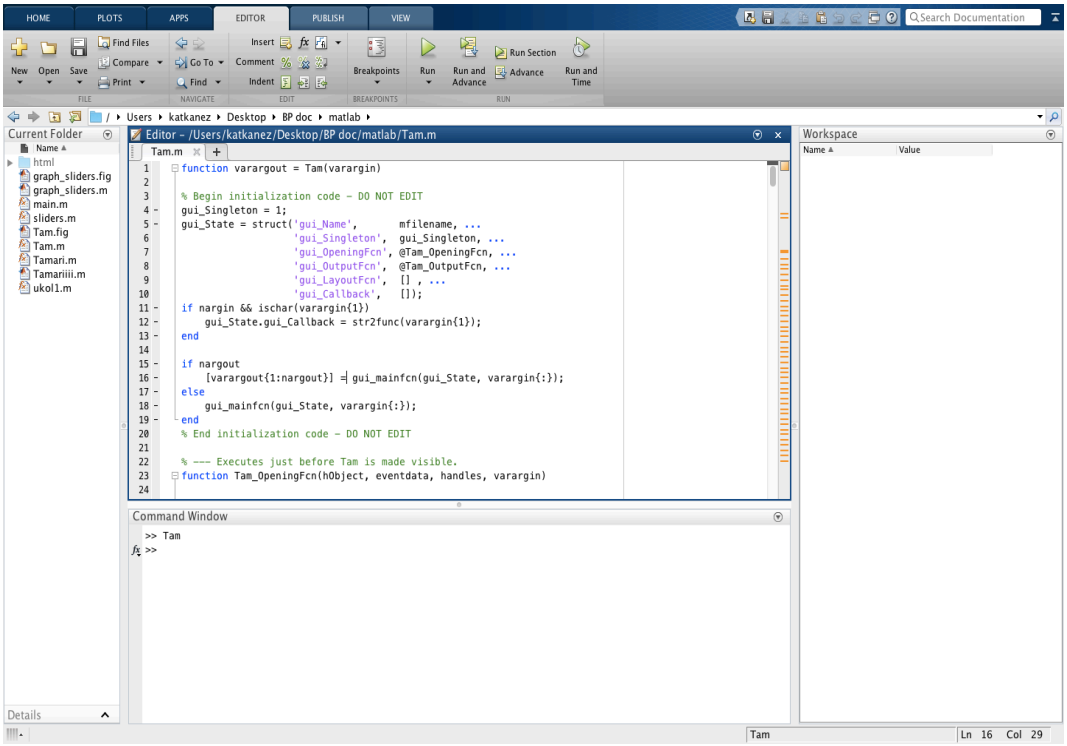
Program MATLAB je nástroj využívaný pre riešenie mnoho matematických problémov, medzi ktoré patrí dátová analýza, optimalizačné problémy a podobne. Využíva matice, polia a operácie s nimi, ktoré nám umožňujú manipuláciu so súbormi údajov rôznymi spôsobmi. Taktiež ponúka grafické užívateľské rozhranie [9].

1.6.1 Pracovné prostredie

Pracovné prostredie MATLABu, ktorého rozloženie si užívateľ môže prispôsobiť na základe svojich preferencií, sa skladá z niekoľkých okien:

- príkazové okno (command window) – zadá príkazy na spracovanie,
- história príkazov (command history) – uchováva históriu predchádzajúcich príkazov zadaných v príkazovom okne,
- aktuálny priečinok (current folder) – grafické užívateľské rozhranie pre manipuláciu s adresármi a súbormi,

- pracovná plocha (workspace) – grafické užívateľské rozhranie pre prezeranie, editovanie, načítanie a ukladanie premenných,
- pomocník (help) – grafické užívateľské rozhranie pre vyhľadávanie a prehliadanie online dokumentácie,
- profiler – nástroj na optimalizáciu výkonu M-súborov,
- výmena súborov (file exchange) – grafické užívateľské rozhranie slúžiace na výmenu súborov,
- editor – textový editor pre vytváranie, úpravu a odstraňovanie chýb v M-súboroch,
- schémy (figures) – vytvára, zobrazuje a upravuje grafy a ďalšie schémy,
- webový prehliadač (web browser) – prehliadač HTML,
- editor premenných (variable editor) – nástroj na prezeranie a úpravu polí vo formáte tabuľky,
- porovnávací nástroj (comparison tool) – grafické užívateľské rozhranie na porovnávanie textových súboroch po riadku.



Obrázok 6: MATLAB R2016a

(Zdroj: Vlastné spracovanie podľa: 10)

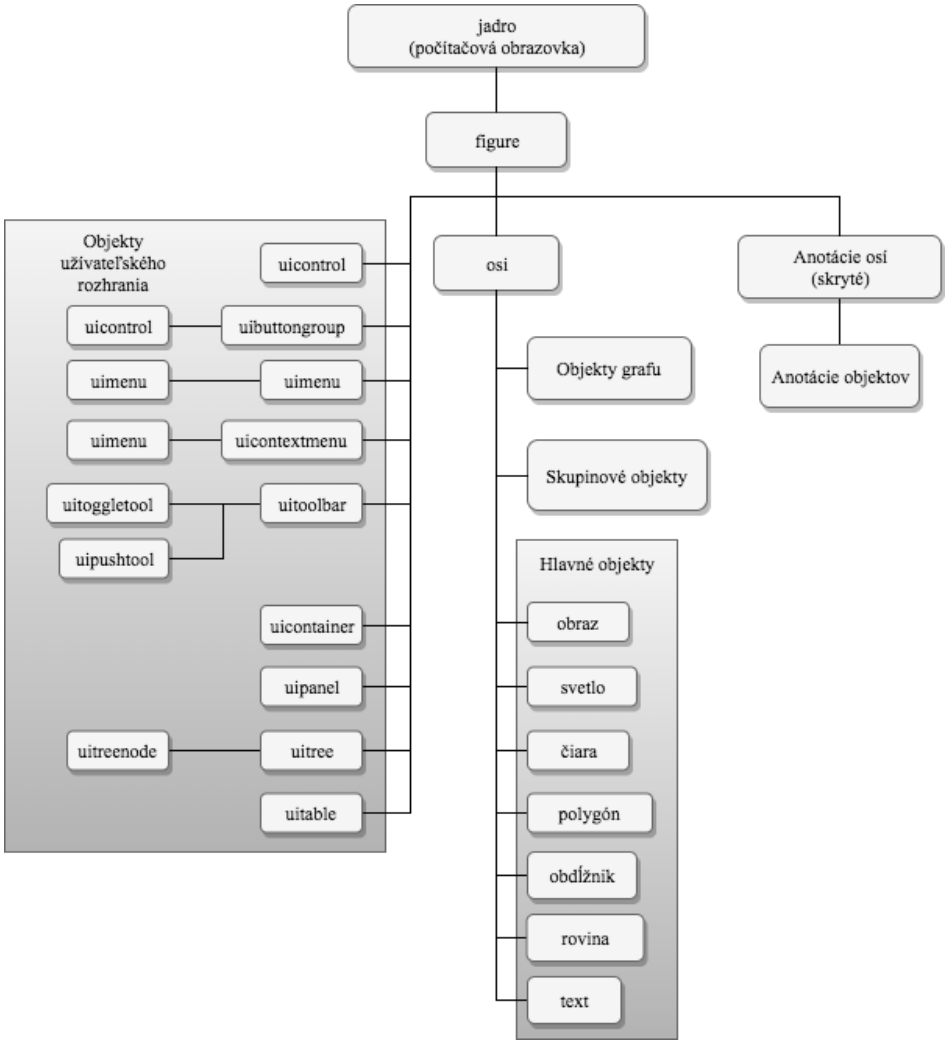
Na obrázku 6 môžeme vidieť príklad pracovného prostredia, konkrétne pre verziu 2016a pre operačný systém macOS. Užívateľ môže okná pracovného prostredia rozmiestniť na základe svojich potrieb, prípadne ich odstrániť alebo pridať [9].

1.6.2 GUI

GUI je skratka pre graphical user interface, inak povedané grafické užívateľské rozhranie. Je to zbierka grafických prvkov, ktoré špecifikujú ako sa majú grafické krivky správať a sú zobrazené v MATLABe. Tieto špecifiká sú následne uložené v dvoch súboroch. Jeden zo súborov, označený koncovkou '.m', tzv. m-file, obsahuje zdrojový kód. Druhý súbor, ktorý obsahuje schému grafických zložiek prepojených so zdrojovým kódom, označený koncovkou '.fig', tzv. figure.

Každý komponent MATLAB grafiky sa nazýva objekt. Objekt je definovaný ako úzko súvisiaca kolekcia dát a funkcií, ktoré tvoria jedinečný celok. Každý objekt má jedinečný identifikátor, ktorý sa nazýva handle, a vlastnosti, ktoré môžu byť upravené podľa potreby. Objekty sú usporiadané v hierarchii od nadradených objektov a pridružených podriadených objektov, ako je znázornené na obrázku 7.

Vždy, keď je vytvorený nový objekt, MATLAB preň automaticky vytvorí handle, ktorý je číslom s dvojitou presnosťou. Číslo s dvojitou presnosťou je typ čísla s pohyblivou rádovou čiarkou, má väčšiu presnosť, používa dvakrát toľko bitov ako jednoduché číslo s pohyblivou čiarkou. Napríklad, keď číslo s jednoduchou presnosťou vyžaduje 32 bitov, tak jeho dvojitá presnosť bude mať 64 bitov.



Obrázok 7: Hierarchia objektov

(Zdroj: Vlastné spracovanie podľa: 9)

Všetky objekty majú súbor vlastností, ktoré ich charakterizujú. Medzi tieto vlastnosti patrí pozícia objektu, farba, typ, nadradený objekt, podradený objekt a pod. Tieto vlastnosti sa dajú zobraziť pomocou dialógového okna, ktoré zahŕňa názov vlastnosti a jej typ hodnoty [9].

2 Analýza problému a súčasnej situácie

V tejto kapitole si priblížime teoretický základ súčasnej situácie analyzovaného problému. Dozvieme sa kto je Ben Tamari, aké sú jeho hlavné diela a aké problémy sú v nich rozoberané.

Predovšetkým sa zameriame na *Tamariho atraktor*, ktorý neskôr využijeme na riešenie konkrétneho problému. Uvedieme jeho matematickú interpretáciu a jednotlivé exogénne veličiny, z ktorých sa skladá. Taktiež opíšeme jednotlivé ekonomické podpriestory, z ktorých sa skladá Tamariho priestor. Nakoniec priblížime jeho ideu, postupnosť a dôvody jeho vzniku.

2.1 Ben Tamari

Ben Tamari sa narodil v roku 1942 v Jeruzaleme. Študoval ekonómiu a filozofiu na Hebrejskej univerzite v Jeruzaleme. Pracoval ako ekonóm pre vládu.

Jeho prvá kniha, s názvom *Ecometry – Foundations of Economics*, bola prezentovaná ako dizertačná práca. V tejto knihe rozoberá nový spôsob predpovede ekonomických procesov. Predpoveď konzervatívnych systémov, ako sú energetické systémy alebo počasie, sa dá vytvoriť v prípade uzavretejších systémov celkom presne. Ben vo svojej knihe rozoberá autonómne hospodárske konzervatívne systémy v Keynesovom priestore, ako sú napríklad systémy rôznych krajín.

V jeho druhej knihe, *Dynamic Economy*, rozoberá teóriu ekonomických predpovedí pomocou formulácie pohybových rovníc, odvodených z Langrangeovej funkcie, a ich riešení, za pomoci Lorentzovej transformácie. Tieto rovnice sú založené na princípoch symetrie, ktoré vo svojich knihách opisuje.

V poslednej knihe, *Conservation and Symmetry Laws and Stabilization Programs in Economics*, opisuje tendenciu autonómnych ekonomických systémov byť konzervatívnymi a symetrickými v Keynesovom priestore, z toho dôvodu môžu byť reprezentované ako komplexné číselné systémy. To umožňuje zlúčiť alebo rozdeliť systémy na všetkých úrovniach, od individuálneho po najobecnejší súhrn, a naopak. Taktiež ponúka súčasné riešenie problému rozdeľovania užitočných zdrojov na trhu [11].

2.2 Tamariho atraktor

Tamariho atraktor, nazvaný po Benovi Tamari, vytvára priestor tretej dimenzie, ktorý sa nazýva ekonomický kozmos alebo Tamariho prostredie. Vyvíja sa z dynamického systému spojeného s teóriou ekonomiky krajiny. Hlavným ťažiskom tejto teórie je predstavenie závislosti *HDP – hotovosť – cena* alebo *HDP – hotovosť – bohatstvo* [11].

2.2.1 Matematická interpretácia

Daná závislosť sa vyjadruje pomocou exogénnych veličín, ktoré sú:

- a – zotrvačnosť inflácie,
- b – produktivita práce,
- c – miera zmeny v peňažnej zásobe,
- d – adaptácia ekonomiky na zmeny,
- e – výmena (kolísanie kurzu vlastnej meny),
- f – valorizácia dôchodkov,
- g – cenová pružnosť/očakávania,
- u – nezamestnanosť,
- i – úroková miera.

Pre simuláciu sú dané exogénne veličiny využité v diferenciálnych rovniciach, pre zobrazenie závislosti HDP – hotovosť – cena, nasledovne:

$$x' = (x - a \cdot y) \cdot \cos(z) - b \cdot y \cdot \sin(z),$$

$$y' = (x + c \cdot y) \cdot \sin(z) + d \cdot y \cdot \cos(z),$$

$$z' = e + f \cdot z + g \cdot \operatorname{atan}\left(\frac{(1-u)\cdot y}{(1-i)\cdot x}\right),$$

kde x' predstavuje HDP, y' predstavuje hotovosť a z' cenu. Výsledkom je verzia špirály.

Verzia atraktoru, ktorá predstavuje pomer HDP – hotovosť – bohatstvo, má zmenu v diferenciálnej rovnici z':

$$z' = e + f \cdot z + g \cdot \operatorname{atan}\left(\frac{(1-u)}{(1-i)\cdot x\cdot y}\right),$$

v tomto prípade z' predstavuje bohatstvo. Pre premenné musí byť splnený rozsah $1 \leq x, y \leq 4$ a počiatočné hodnoty x_0, y_0, z_0 sú rovné 1. Pre analýzu v tejto práci budeme využívať verziu atraktoru [12].

2.2.2 Primárne komponenty

Tamariho priestor je založený na základe troch podpriestorov druhej dimenzie:

- HDP – hotovosť → Keynesov model,
- HDP – cena → Patinkinov model,
- hotovosť – cena → Friedmanov model [11].

Keynesova teória určovania rovnovážneho reálneho HDP, zamestnanosti a cien sa zameriava na vzťah medzi agregátnymi príjmami a výdajmi. Vytvoril model, v ktorom tvrdí, že rovnovážna úroveň produkcie alebo reálneho HDP nemusí korešpondovať s prirodzenou úrovňou reálneho HDP. Model ukazuje, že reálny HDP je na takej úrovni, ktorá je v súlade so súčasnou úrovňou agregátnych výdavkov. Ak súčasná hodnota agregátnych výdavkov nie je dostatočná na nákup celého dodaného reálneho HDP, produkcia sa zníži, kým sa hodnota reálneho HDP nebude rovnať hodnote celkových výdavkov. Z toho dôvodu, ak súčasná hodnota agregátnych výdavkov nestačí na nákup prirodzenej hladiny reálneho HDP, potom rovnovážny bod reálneho HDP bude niekde pod prirodzenou hladinou [14].

Patinkinova teória, ktorú nazval klasická dichotómia, integruje monetárnu teóriu a teóriu hodnoty. Pomocou tejto teórie by sa mohli stanoviť absolútne a relatívne ceny. Táto teória vznikla na základe zahrnutia efektu reálnych peňažných zostatkov do teórie o všeobecnej ekonomickej rovnováhe, kde ekonomické subjekty vyvodzovali úžitok z prvkov, ktoré boli vytvorené na základe efektu reálnych peňažných zostatkov. Na základe zmien sa odrážajú účinky reálnych premenných, čo predstavuje špecifický prípad efektu bohatstva [15].

Friedman vytvoril teóriu dopytu po peniazoch. Rozlišuje dva typy dopytu po peniazoch. Prvý typ nám poukazuje na to, že peniaze sú požadované na transakčné účely, čiže slúžia ako prostriedok výmeny. Tento pohľad na peniaze je rovnaký ako pôvodná kvantitatívna teória. Podľa druhého typu sú peniaze požadované z dôvodu

prínosu. Predstavujú dočasnú kúpnu silu, a teda majetok alebo časť bohatstva. Friedman považuje dopyt po peniazoch za súčasť teórie bohatstva [16].

Tamari dokázal, že mechanizmus prevodu peňazí na trh nie je priamy, ale zahŕňa vplyv *Greshamovho zákona*. Tento zákon hovorí, že ak sú dva druhy mien v obehu a majú rovnakú nominálnu hodnotu ale odlišnú reálnu hodnotu, mena s vyššou reálnou hodnotou sa nahromadí a postupne vyradí menu s nižšou reálnou hodnotou z obehu. V matematickej interpretácii Tamariho atraktoru je využitý Greshamov koeficient $(1-u)\cdot y$, ktorý reprezentuje dopyt a $(1-i)\cdot x$, ktorý reprezentuje ponuku [11, 13].

2.3 Idea a vznik atraktoru

Ben Tamari pracoval ako ekonóm pre Izraelský úrad pre hospodárske plánovanie v 70-tych rokoch, kde vypracovával ročnú prognózu a 5 ročné ekonomické plány. Plány a prognózy sa zostavovali na základe metódy zotrvačnosti. Na základe tejto metódy by sme zobrali posledné miery rastu rôznych parametrov a odhadli ich do budúcnosti. Táto metóda bola dostačujúca až dokiaľ sa nezmenila politická situácia v Izraeli.

Zo sociálno-demokratickej spoločnosti, ktorá bola charakterizovaná solidaritou a plánovaním, sa po zmene vlády Izrael zmenil na kapitalisticko-liberálnu krajinu. S touto zmenou začala rásť inflácia, a z toho dôvodu začalo byť vytváranie prognóz a plánovanie problematické. V tej dobe boli pre vytváranie prognóz a plánovanie využívané ekonometrické modely, ktoré ponúkali spolu s počítačovými tabuľkami výpočtovú schopnosť, ale málo matematickej kapacity.

Tamari si uvedomil, že pre proces riadneho plánovania a vytvárania predpovedí chýbali rovnice, ktoré vyjadrovali správanie ekonomiky. Pri jeho prvom pokuse, o pridanie vhodného matematického nástroja do systému, bol využitý mechanizmus pavučiny pre opis správania trhu s obydliami v Izraeli. S nárastom inflácie sa zdalo, že najdôležitejšou premennou by bolo množstvo peňazí, ktoré malo zásadný vplyv na ostatné premenné v systéme. Očakávanú koreláciu medzi peniazmi a cenou sa nepodarilo nájsť. Preto ukázal, že mechanizmus prevodu peňazí na trh nie je priamy, ale zahŕňa vplyv Greshamovho zákona. V tých dňoch bol akceptovaný efekt reálnej rovnováhy, pomocou ktorého sa riešil vplyv peňazí na ekonomiku. Bolo prosté predpokladať, že inflácia bude neutrálna tak dlho, kým budú ekonomické premenné spojené s indexom spotrebiteľských cien.

Jeho ďalšou dokázanou myšlienkou bolo, že tlačenie peňazí zrýchľuje infláciu a tá prostredníctvom viazaných mechanizmov zrýchľuje tlač peňazí. Keď je prekročená určitá kritická suma, ich tendencia sa zvráti. Verejnosť už ďalej nezhromažďuje hotovosť, v skutočnosti sa jej zbavuje, čo spôsobuje infláciu. To ukázal na základe toho, že na samotnom rozsahu nezáleží, či je to veľká, či malá krajina, aplikujú sa rovnaké pravidlá. Záverom bolo, že dôležitou premennou nebola skutočná bilancia, ale množstvo peňazí v závislosti s výstupom.

Hlavnými otázkami jeho práce bolo, čo sa stane, keď sa peniaze zavádzajú ako premenná v spotrebných a výrobných funkciách, a čo sa stane v ekonomickom systéme, ak je množstvo peňazí zvýšené na neurčito. Rýchlo sa ukázalo, že reálnym problémom boli meradlo a merania, zatiaľ čo meradlo, v tomto prípade peniaze, je súčasťou meraného systému, ktorý bol predstavovaný výnosom, peniazmi a časom.

Na začiatku pracoval len v priestore výnos-peniaze. Vysvetlil, ako prebytok peňazí zakrivil ekonomický priestor a zapríčinil, že ekonomické jednotky sa odchyľujú od svojich optimálnych podmienok. Základnou hypotézou v jeho práci bolo, že správne matematické plochy pre vysvetlenie ekonomických systémov sú minimálne plochy, kvôli ich vlastnosti minimalizácii. Všetky jeho snahy dokázať to ekonomicky zlyhali.

V ďalšej pracovnej fáze sa pokúsil skontrolovať dynamiku systému pridaním tretej rovnice pre cenu, výsledok by mal fungovať ako rovnica spätnej väzby. Nová rovnica bola formulovaná pomocou procesu skúšania a chybenia, až kým sa štatistické údaje nezhodovali. Táto rovnica vyjadruje, že množstvo peňazí, ktoré je prinesené na trh pre výmenu, je citlivé na úrokovú mieru, nezamestnanosť, výmenný kurz a valorizáciu. Týmto spôsobom sa vytvoril trojrozmerný priestor, ktorý predstavuje výnos, peniaze a cenu. Ekonometrické výsledky ukázali, že ekonomické systémy sú symetrické.

Tieto ekonomické rovnice boli vložené do softwaru Mathematica pre lepšie pochopenie grafickej štruktúry systému. Výsledný graf nazval Nest. Nest je ekonomický povrch, ktorý predstavuje ekonomický systém, v konkrétnom rozsahu parametrov je atraktorom. Mapovanie Nest slúži ako účinný nástroj na navigáciu a riadenie národného hospodárstva. Väčšina ekonomických systémov spadá do časti, kde je povrch takmer lineárny, situácia je nechaotická a neexistuje vysoká citlivosť na počiatočné podmienky. Príležitostne, počas politicky chaotických období, napríklad počas vojenských konfliktov, sa ekonomické systémy nachádzajú v nebezpečnej zóne. V tejto zóne sa

krajina približuje ku kritickým chaotickým oblastiam, spolu s neustálym zvyšovaním citlivosti vzhľadom k počiatočným podmienkam a stráca kontrolu. Tvorba predpovedí a plánovanie je možné, s primeranou presnosťou a predpoveďou v krátkodobom horizonte, až do 3 rokov [11].

3 Vlastné návrhy riešenia, prínos návrhov riešenia

V tejto kapitole si predstavíme návrh riešenia na zlepšenie súčasnej situácie. Prvým krokom bude analýza Tamariho atraktoru. Ďalším krokom, na základe analýzy, bude zostavenie simulácie modelu Tamariho atraktoru pomocou MATLABu, v ktorom využijeme grafické užívateľské rozhranie na zostavenie simulátora. Detailne opíšeme zostavenie objektov a zostrojenie kódu, v ktorom popíšeme jednotlivé úseky pre jednoduchšie pochopenie jeho funkčnosti.

V ďalšej časti budeme rozoberať konkrétny problém. Ten bude pozostávať v porovnaní ekonomík, niektorých krajín, na základe jednotlivých parametrov. Tento problém bude skúmaný na základe štatistických údajov, ktoré boli verejne dostupné, a pomocou simulácie modelu, ktorý sme vytvorili pre tento účel.

3.1 Tamariho atraktor v MATLABe

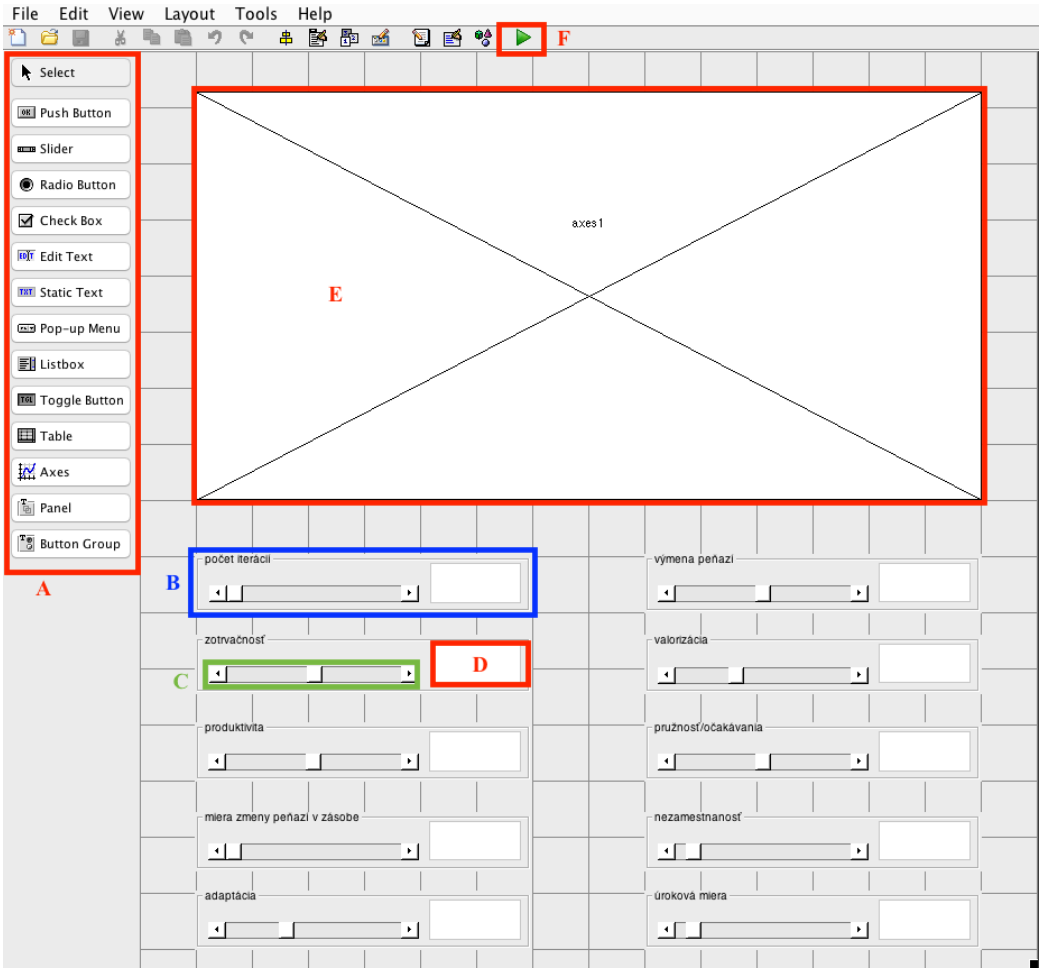
V tejto podkapitole si podrobne opíšeme postup tvorenia simulácie Tamariho atraktoru v MATLABe. Táto tvorba pozostáva z dvoch častí. Prvou časťou je vytvorenie grafického rozhrania pre Tamariho atraktor pomocou GUI. V ďalšej časti bude detailne opísaná tvorba kódu, ktorý je prepojený s GUI.

3.1.1 Tvorba pomocou GUI

Pre vytvorenie simulácie Tamariho atraktoru sme využili grafické užívateľské rozhranie, tzv. GUI. V GUI sme si vybrali niektoré objekty, pre ktoré sme následne upravili vlastnosti podľa potreby.

Na obrázku 8 v časti A môžeme vidieť ponuku objektov. Patrí tam tlačidlo, slider, prepínacie tlačidlo, políčko na začiarknutie, okno pre editovanie, statický text, tlačidlo na rozbalenie ponuky, zoznam s ponukou, tlačidlo pre vypnutie a zapnutie, tabuľka, graf, panel, panel pre zoskupovanie tlačidiel.

V časti B môžeme vidieť použitý panel. Tomuto panelu boli zmenené vlastnosti v tzv. inšpektorovi, ktorý môžeme vidieť na obrázku 9. Bola zmenená pozícia, veľkosť, štítok a popis. Podobne bolo vytvorených ďalších 9 panelov, odlišujú sa len pozíciou, štítkom a popisom.

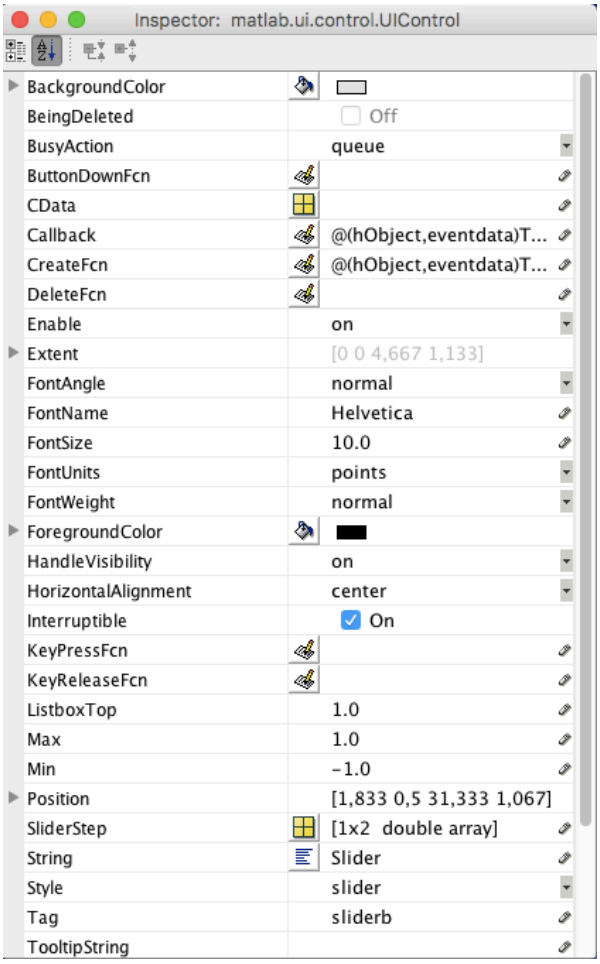


Obrázok 8: GUI

(Zdroj: Vlastné spracovanie podľa: 10)

Slider, ktorý vidíme zaradený pod písmenkom C má upravené maximum a minimum, pozíciu, štítok. To isté platí pre ostatné. Taktiež to platí pre okno pre editovanie, ktoré je označené písmenom D, pričom každé takéto okno je prepojené s určitým sliderom, ktorý sa nachádza v tom istom paneli ako konkrétne okno pre editovanie.

Pod písmenom E vidíme graf, ktorý sa následne po spustení zobrazí v 3D podobe. Tento graf je prepojený so všetkými slidermi a oknami pre editovanie. Všetky grafické úpravy v GUI sa musia uložiť pomocou tlačidla, ktoré vidíme pri písmenku F. Po jeho kliknutí sa automaticky prepíšu požadované časti kódu, vďaka ktorému sú tieto zmeny po spustení viditeľné.



Obrázok 9: Vlastnosti

(Zdroj: Vlastné spracovanie podľa: 10)

3.1.2 Zdrojový kód

Pri vytvorení GUI sa automaticky vytvoria určité časti kódu, ktoré môžu byť následne upravované podľa potreby. Iba do počiatočnej časti kódu by sa nemalo zasahovať. V tejto časti vidíme vytvorenú základnú funkciu, ktorá je nazvaná Tam a pod ňou sa nachádza počiatočný kód, ktorý sa neodporúča meniť:

```
function varargout = Tam(varargin)

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Tam_OpeningFcn, ...
                   'gui_OutputFcn',  @Tam_OutputFcn, ...
```

```
                'gui_LayoutFcn',  [] , ...
if nargin && ischar(varargin{1})
   gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
   [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
   gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

Obrázok 10: Počiatočný kód

(Zdroj: Vlastné spracovanie podľa: 10)

Nasleduje otváracia funkcia, ktorá sa vykoná hneď pred zobrazením úvodného okna obsahujúceho Tamariho atraktor, ktorý vidíme na obrázku 16.

```
% --- Executes just before Tam is made visible.
function Tam_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
```

Obrázok 11: Otváracia funkcia

(Zdroj: Vlastné spracovanie podľa: 10)

Potom vytvoríme globálne premenné. Veľké písmená boli vytvorené pre jednoduchšie značenie, ktoré je využité pri rozlišovaní zmien konkrétnych sliderov okien pre editovanie. Vďaka tomu program vie, ktoré hodnoty sa majú ako zmeniť a následne sa podľa toho mení iba určitá časť grafu. Malé písmenká predstavujú hodnoty konkrétnych premenných, ktoré sú v tomto prípade vykreslené v podobe atraktoru na obrázku 16.

```
global A B C D E F G U I N a b c d e f g u i n
a = -0.587;b = -0.011;c = 0.02; d = 0.96;e = 0;f = 0.01;g = 1;
u = 0.05;i = 0.1; n = 100;
A = 0; B = 1; C = 2; D = 3; E = 4; F = 5; G = 6; U = 7; I = 8; N = 9;
```

Obrázok 12: Počiatočný graf

(Zdroj: Vlastné spracovanie podľa: 10)

Ďalej napíšeme časť kódu, ktorá vytiahne hodnotu z konkrétneho okna pre editovanie a následne ju konvertuje z čísla na textový reťazec, tzv. string. Potom aktualizujeme dáta.

```
set(handles.edita,'String',num2str(a));
set(handles.editb,'String',num2str(b));
set(handles.editc,'String',num2str(c));
set(handles.editd,'String',num2str(d));
set(handles.edite,'String',num2str(e));
set(handles.editf,'String',num2str(f));
set(handles.editg,'String',num2str(g));
set(handles.editu,'String',num2str(u));
set(handles.editi,'String',num2str(i));
set(handles.editn,'String',num2str(n));
% Update handles structure
guidata(hObject, handles);
```

Obrázok 13: Set

(Zdroj: Vlastné spracovanie podľa: 10)

Automaticky sú vytvorené dve funkcie. Jedna, z ktorej sa výstupy vrátia do príkazového riadku. A druhá, ktorá získava výstupy z identifikátorov, tzv. handles.

```
% --- Outputs from this function are returned to the command line.
function varargout = Tam_OutputFcn(hObject, eventdata, handles)
% Get default command line output from handles structure
varargout{1} = handles.output;
```

Obrázok 14::Automatické funkcie

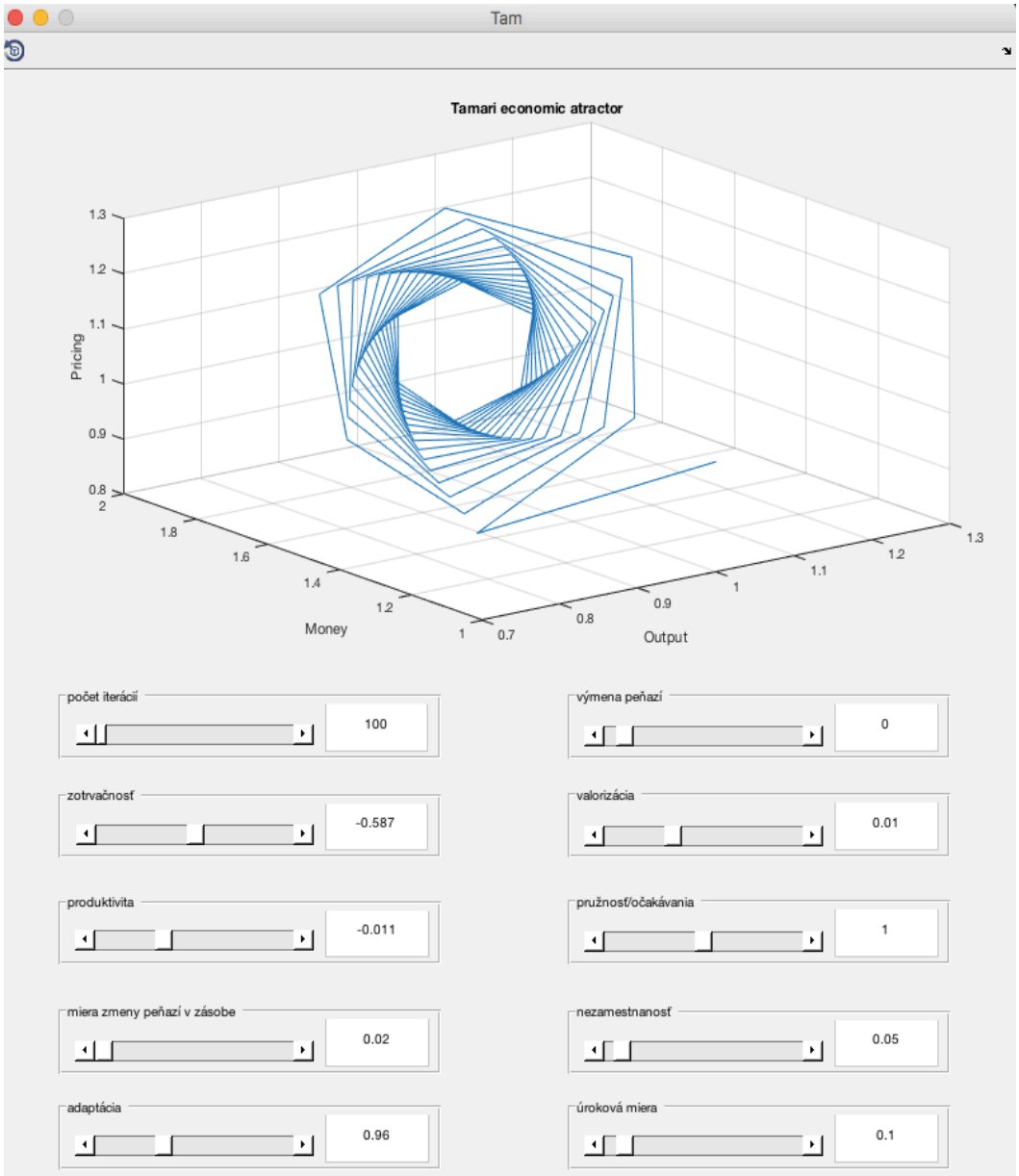
(Zdroj: Vlastné spracovanie podľa: 10)

Vytvoríme funkciu, ktorá je prepojená so zmenou dát pre slider, nazveme ju runODE.

```
% function for slider
function runODE(hObject,eventdata,handles,isWinActive,callbackType)
```

Obrázok 15: runODE

(Zdroj: Vlastné spracovanie podľa: 10)



Obrázok 16: Úvodný Tamariho atraktor pri spustení

(Zdroj: Vlastné spracovanie podľa: 10)

Táto funkcia obsahuje *S*, ktoré určuje veľkosť hodnoty posunu jednotlivých sliderov. Pomocou podmienky *if – else* skontrolujeme vykreslenie počiatočného grafu, túto podmienku nazveme *isWinActive*.

```
S = [1 1 1 1 1 1 1 1 1 0.01]; % slider strength vector

% control of the window activation (for the first graph, when opened)
if (isWinActive)
   m = get(hObject,'Value');
```

```
else
   m = 0.1;
end
```

Obrázok 17: Kontrola aktivácie okna

(Zdroj: Vlastné spracovanie podľa: 10)

Do kódu zahrnieme prepínač, tzv. switch, vďaka ktorému bude program vedieť, ktorý prípad, tzv. case, má vybrať pre zmenu hodnôt. Každý case obsahuje príkaz na prepočet premennej, kde sú využité hodnoty z *S*. Bez tejto časti kódu by nastal problém, pretože je potrebné aby sa menil len konkrétny slider, ktorý posunieme buď pomocou šípok na krajoch, alebo ho potiahneme manuálne.

```
% storing the change of values - graph change
global A B C D E F G U I N a b c d e f g u i n
switch callbackType
   case A
      a = m*S(1);
   case B
      b = m*S(2);
   case C
      c = m*S(3);
   case D
      d = m*S(4);
   case E
      e = m*S(5);
   case F
      f = m*S(6);
   case G
      g = m*S(7);
   case U
      u = m*S(8);
   case I
      i = m*S(9);
   case N
      n = m*S(10);
   otherwise

end
```

Obrázok 18: Case

(Zdroj: Vlastné spracovanie podľa: 10)

Zahrnieme najdôležitejšiu časť kódu, ktorá predstavuje Tamariho atraktor. Na začiatok priradíme hodnotu 1, potom spravíme cyklus obsahujúci tri diferenciálne rovnice, ktorými je definovaný.

```
% Tamari attractor differential equations
x(1)=1;y(1)=1;z(1)=1;
for j=1:n
   x(j+1)=(x(j)-a*y(j))*cos(z(j))-b*y(j)*sin(z(j));
   y(j+1)=(x(j)+c*y(j))*sin(z(j))+d*y(j)*cos(z(j));
   z(j+1)=e+f*z(j)+g*atan((1-u)/(1-i)*x(j)*y(j));
end
```

Obrázok 19: Rovnice Tamariho atraktoru

(Zdroj: Vlastné spracovanie podľa: 10)

Pridáme príkaz na vykreslenie 3D grafu, zobrazenie mriežky a názvy osí.

```
plot3(handles.axes1,x,y,z);
grid on; xlabel('x'); ylabel('y');ylabel('z');
title('Tamari economic atractor');xlabel('Output');ylabel('Money');zlabel('Pricing');
```

Obrázok 20: Vykreslenie grafu

(Zdroj: Vlastné spracovanie podľa: 10)

Podobne vytvoríme funkciu, ktorá rieši zmenu hodnôt v editovacích oknách, nazveme ju runODE2. Jedinou zmenou, ktorú neobsahuje runODE, je pridanie príkazu na prekonvertovanie z textového reťazca na číslo, z dôvodu funkčnosti programu.

```
% function for edit box
function runODE2(hObject,eventdata,handles,isWinActive,callbackType)
if (isWinActive)
   m = str2num(get(hObject,'String'));
else
   m = 0.1;
end
```

Obrázok 21: runODE2

(Zdroj: Vlastné spracovanie podľa: 10)

Pre každý slider boli automaticky vytvorené funkcie pre tzv. *Callback* a *Create Function*. Do funkcie callback sú zapísané príkazy, vďaka ktorým sa mení simulácia. Create Function je potrebná pre zobrazenie požadovaného objektu, zahŕňa informácie ako

sú farba objektu, pozícia a pod. Podobu tých funkcií si ukážeme pre premennú *a*, ktorá predstavuje zotrvačnosť. Vyvoláme funkciu runODE, priradíme hodnoty a následne ich aktualizujeme. Create Function je automaticky doplnená údajmi na základe zmien vlastností konkrétneho objektu.

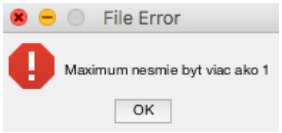
```
% slider a
function slidera_Callback(hObject, eventdata, handles)
global A
runODE(hObject,eventdata,handles,true,A);
global a
set(handles.edita,'String',num2str(a));
guidata(hObject,handles);

function slidera_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor',[.9 .9 .9]);
end
```

Obrázok 22: Slider

(Zdroj: Vlastné spracovanie podľa: 10)

Pre editovacie okno sú taktiež vytvorené dve funkcie. Create Function obsahuje iné vlastnosti. V callback funkcii si vytvoríme premennú s názvom *Theta*, pomocou ktorej vytiahneme hodnotu z editovacieho okna a zmeníme ju z dátového typu string na double, ktoré predstavuje reálne číslo. Následne, pomocou podmienky *if-elseif*, zistíme či sa hodnota, ktorá bola zadaná do editovacieho okna, nenachádza mimo intervalu, ktorý je určený minimálnou a maximálnou hodnotou. V prípade, že by sa zadaná hodnota nachádzala mimo tento interval, vyskočí varovné okno, ktoré nás upozorňuje, že sme zadali hodnotu, ktorá je mimo intervalu a taktiež nám oznámi, aké hodnoty sú prípustné. Príklad varovného okna vidíme na obrázku 23.



Obrázok 23: Varovné okno

(Zdroj: Vlastné spracovanie podľa: 10)

Ak zadaná hodnota prekročí maximálnu hodnotu, automaticky bude nastavená maximálna možná hodnota. V opačnom prípade bude automaticky nastavená minimálna

možná hodnota. Nakoniec, tak isto ako pre funkcii pre slider, bude vyvolaná funkcia, v tomto prípade runODE2, priradené a aktualizované hodnoty.

```
% edit box a
function edita_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.edita,'String'));
if Theta < -10
   Theta = -10;
   errordlg('Minimum nesmie byt mensie ako -10','File Error');
elseif Theta > 10
   Theta = 10;
   errordlg('Maximum nesmie byt viac ako 10','File Error');
end
% Set slider value.
set(handles.slidera, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.2f',Theta);
set(handles.edita, 'String', caption);
global A a
runODE2(hObject,eventdata,handles,true,A);
set(handles.slidera,'Value',a);
guidata(hObject,handles);
```

Obrázok 24: Edit box

(Zdroj: Vlastné spracovanie podľa: 10)

Nesmie chýbať create function obsahujúca vlasnosti konkrétneho editovacieho okna.

```
function edita_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end
```

Obrázok 25: Edit box create function

(Zdroj: Vlastné spracovanie podľa: 10)

Podobne budú vytvorené funkcie pre ostatné premenné. Líšiť sa budú vo vlastnostiach a štítkoch, ktorý má každý objekt jedinečný, kvôli rozoznaniu a priradeniu jednotlivých objektov v kóde.

Pre lepšie sledovanie zmien v atraktore obsahuje grafické zobrazenie tlačidlo, ktoré vidíme na obrázku 26, pomocou ktorého dokážeme obrázok rotovať. Pre jednoduchšie

skúmanie môžeme otočiť obrázok na podobu 2D grafu, kde budeme sledovať zmenu bohatstva krajiny voči HDP a následne voči hotovosti v krajine.

Obrázok 26: Tlačidlo pre rotáciu

(Zdroj: Vlastné spracovanie podľa: 10)

3.2 Riešená problematika

Problematikou, ktorou sa budeme zaoberať, je využitie Tamariho atraktoru pre skúmanie predpovede budúceho ekonomického diania určitej krajiny. Využijeme simulátor pre zobrazenie atraktoru, ktorého tvorbu sme si opísali v predošlej podkapitole.

Na predpoveď budeme potrebovať údaje jednotlivých premenných pre danú krajinu, ktorou bude v našom prípade Česká republika. V tomto príklade si ukážeme jednotlivé prípady zmien atraktoru, ktoré môžu nastať na základe zmien určitých parametrov.

Ako druhý typ príkladu, ktorý si predvedieme, bude porovnanie viacerých krajín. Zvolili sme si Českú republiku, Dánsko, Nemecko, Španielsko.

3.2.1 Prípadové štúdie Českej republiky

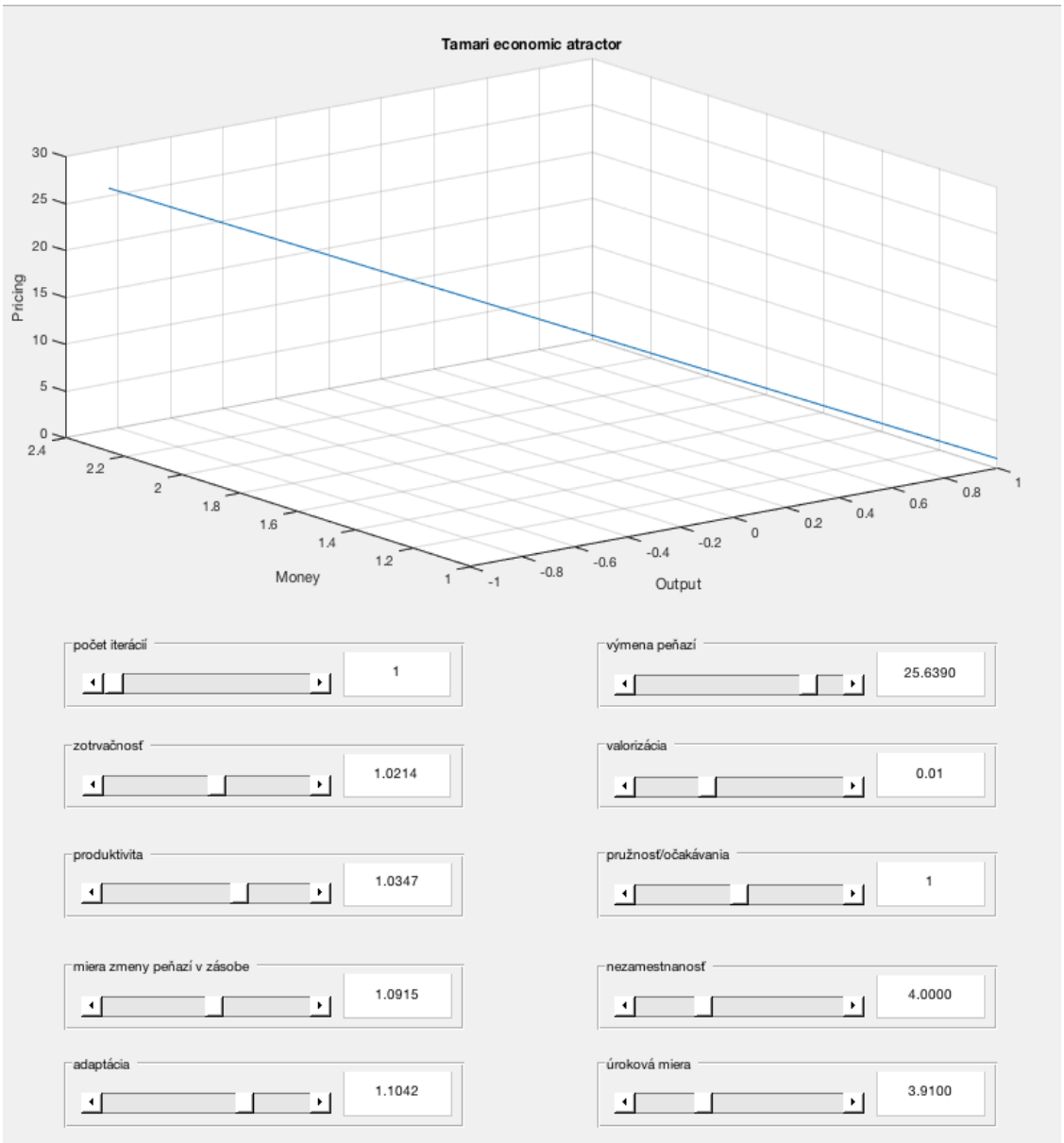
V tejto štúdii využijeme Tamariho atraktor pre skúmanie ekonomiky Českej republiky. Pre premenné, potrebné pre Tamariho atraktor, využijeme údaje uvedené v tabuľke 1, ktoré sú za rok 2016.

Výmenný kurz (e), nezamestnanosť (u) a úrokovú mieru (i) sme získali ako priame údaje zo štatistických tabuliek (IFS – International Financial Statistics Yearbook 2017). Zotrvačnosť inflácie (a), produktivitu práce (b), mieru zmeny v peňažnej zásobe (c) a adaptáciu ekonomiky na zmeny (d) sme získali ako percentuálny rast alebo pokles oproti predchádzajúcemu roku. Valorizáciu dôchodkov (f) a cenová pružnosť/očakávania (g) sme určili ako konštantu, pretože ich zmena vyvoláva minimálnu zmenu v atraktore.

Tabuľka 1 Česká republika premenné

(Zdroj: Vlastné spracovanie podľa: 17)

	a	b	c	d	e	f	g	u	i
Česko	1.0214	1.0347	1.0915	1.1042	25.639	0.01	1	4.0	3.91



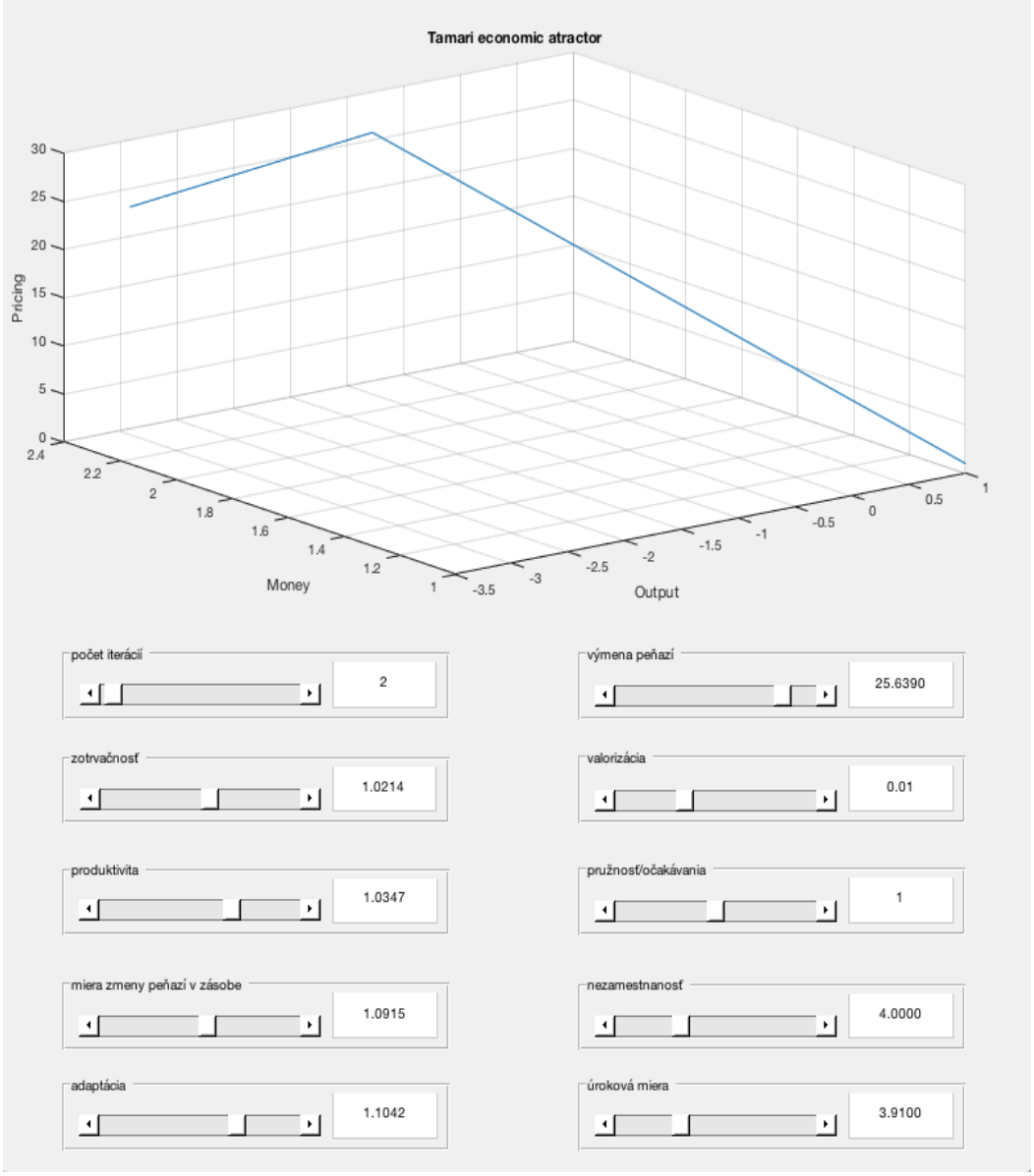
Obrázok 27: Česko rok 2016

(Zdroj: Vlastné spracovanie podľa: 10)

Na základe údajov z tabuľky 1 a kódu, pre Tamariho atraktor, vytvoríme jeho simuláciu. Na obrázku 27 vidíme Tamariho atraktor pre rok 2016, ktorý bude

východiskový rok. Os *x*, z obrázku 27, predstavuje HDP, os *y* predstavuje hotovosť a os *z* predstavuje bohatstvo.

Na základe roku 2016, ktorý je počiatočný, budeme sledovať zmeny v krivke pre nasledujúce roky. V roku 2016 môžeme vidieť, že hotovosť a bohatstvo krajiny s rastúcim HDP klesajú, ale z pohľadu rastúcej hotovosti bohatstvo rastie.



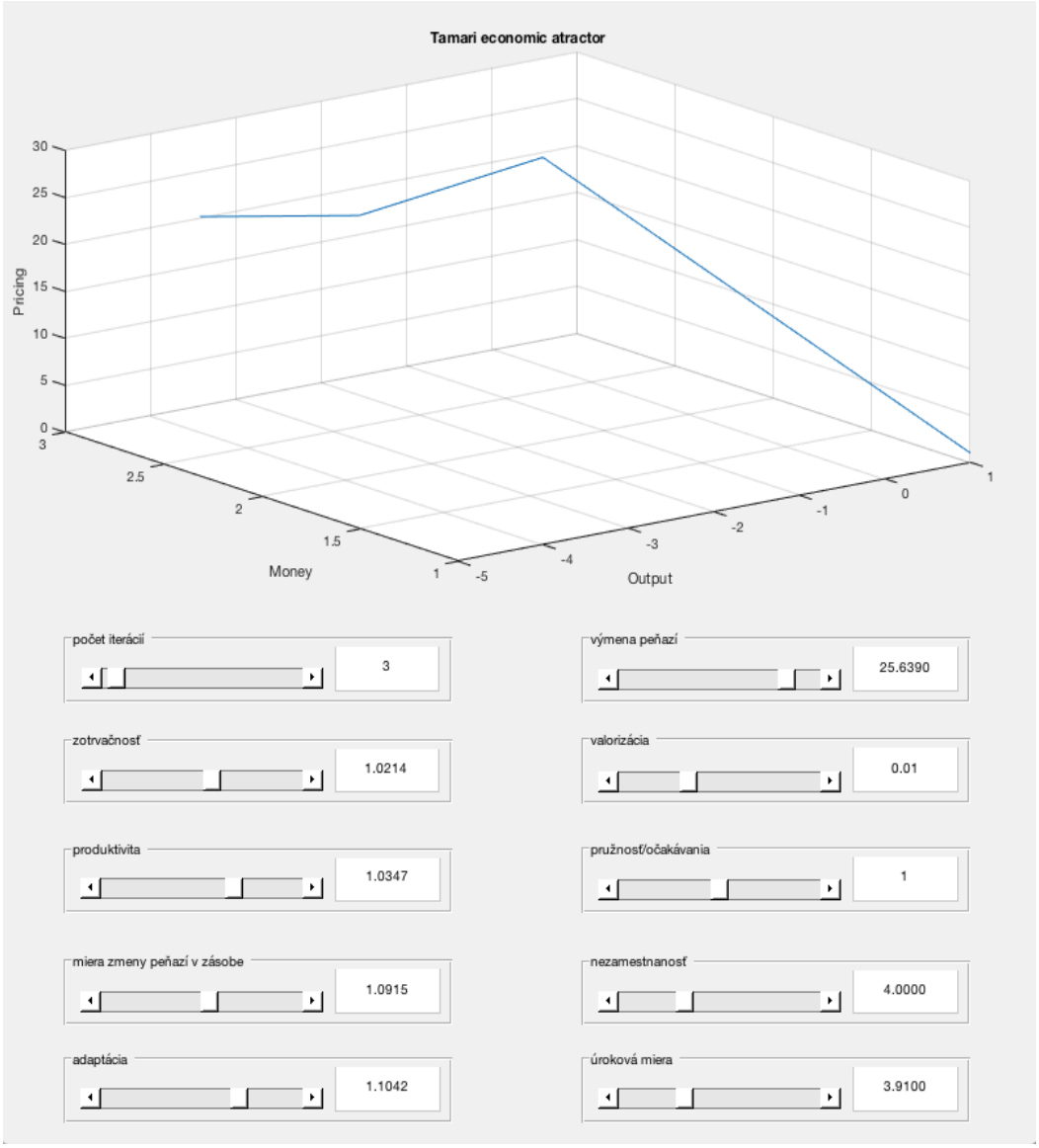
Obrázok 28: Česko rok 2017+

(Zdroj: Vlastné spracovanie podľa: 10)

Zvýšime počet iterácií o 1, čím vytvoríme predpoveď na nasledujúci rok. Tým vytvoríme predikciu pre rok 2017, ktorého údaje v štatistických tabuľkách IFS nie sú zatiaľ dostupné. Znakom + budeme označovať predikciu, čo znamená, že na obrázku 28

vidíme rok 2017^{+}. Podobne budeme označovať ďalšie roky, kde zvýšenie počtu iterácií o 1 znamená zvýšenie predpovede o 1 rok.

Pre rok 2017^{+} vidíme miernu zmenu v krivke. Táto zmena poukazuje na rast hotovosti a bohatstva s rastúcim HDP až do bodu, ktorý sa nachádza v intervale (-1; -0,5). V tomto bode hotovosť a bohatstvo, s rastúcim HDP, začnú klesať. Taktiež bohatstvo rastie spolu s hotovosťou.



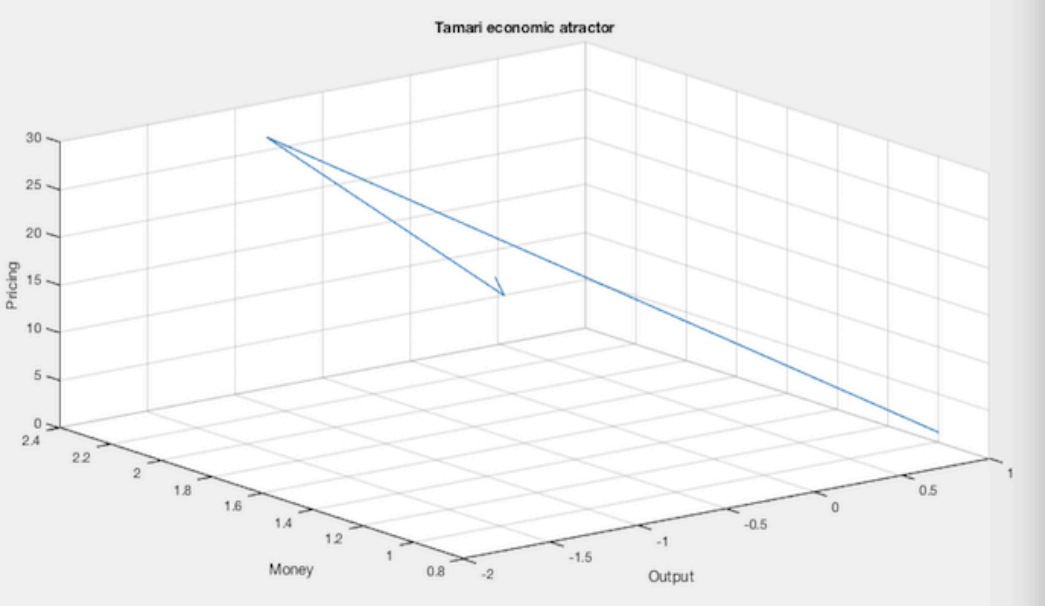
Obrázok 29: Česko rok 2018^{+}

(Zdroj: Vlastné spracovanie podľa: 10)

Na obrázku 29 je zobrazený rok 2018^{+}. Zmena vzhľadom k roku 2017^{+} je predovšetkým v hotovosti. Hotovosť, s rastúcim HDP, klesá do bodu nachádzajúceho sa

v intervale (-2; -1). Následne mierne rastie, až kým HDP nenadobudne bod, ktorý sa nachádza v intervale (0; 1), odkiaľ začne hotovosť rapídne klesať. Bohatstvo má voči HDP podobný charakter, zmena je len v intenzite rastu. V porovnaní zmeny krivky z pohľadu bohatstva voči hotovosti začína bohatstvo nadobúdať konštantný charakter.

Na obrázku 30 sme si zobrazili Tamariho atraktor pre Českú republiku za rok 2018[+], v ktorom sme zmenili hodnotu nezamestnanosti zo 4% na 12%. Naopak, na obrázku 31, hodnota nezamestnanosti poklesla na 1%. Nárast hodnoty nezamestnanosti spôsobil viac chaotický výkyv atraktoru, ako v prípade, kde nezamestnanosť klesla. Zvýšenie nezamestnanosti na obrázku 30 zmenila atraktor tak, že je v podobe prudkého skoku, pretože jej zvýšenie bolo veľké. Rôzne zmeny v parametroch môžu vyvolávať rôzne zmeny v atraktore.

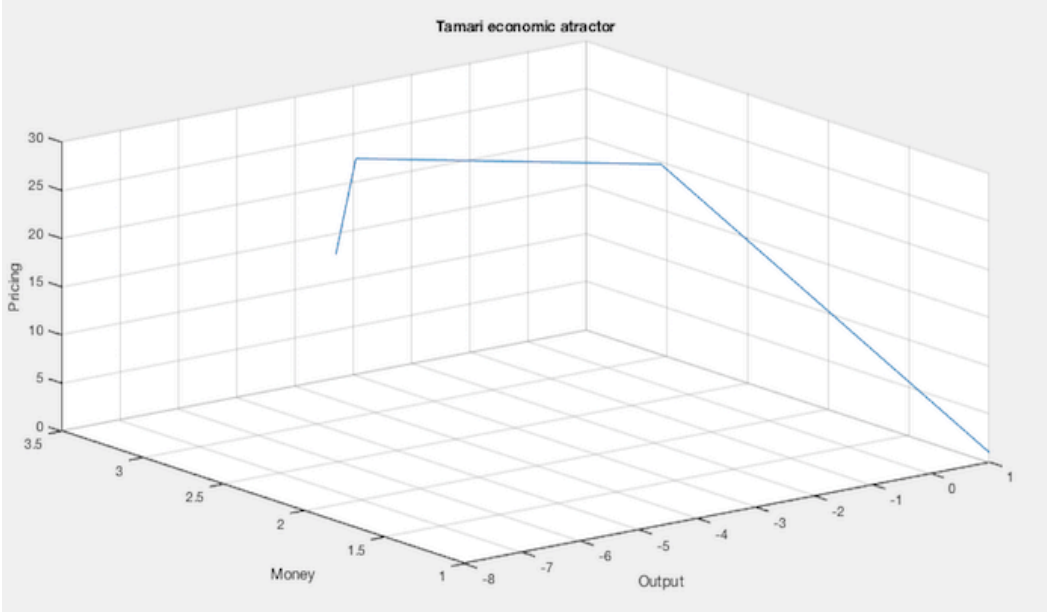


Obrázok 30: Nárast hodnoty nezamestnanosti

(Zdroj: Vlastné spracovanie podľa: 10)

Pri zvýšenej nezamestnanosti sa dá spozorovať, že bohatstvo voči HDP rastie do určitej hodnoty a potom začne prudko klesať. Z pohľadu bohatstvo – hotovosť, obe rastú a následne od určitej hodnoty začne klesať aj bohatstvo, aj hotovosť krajiny.

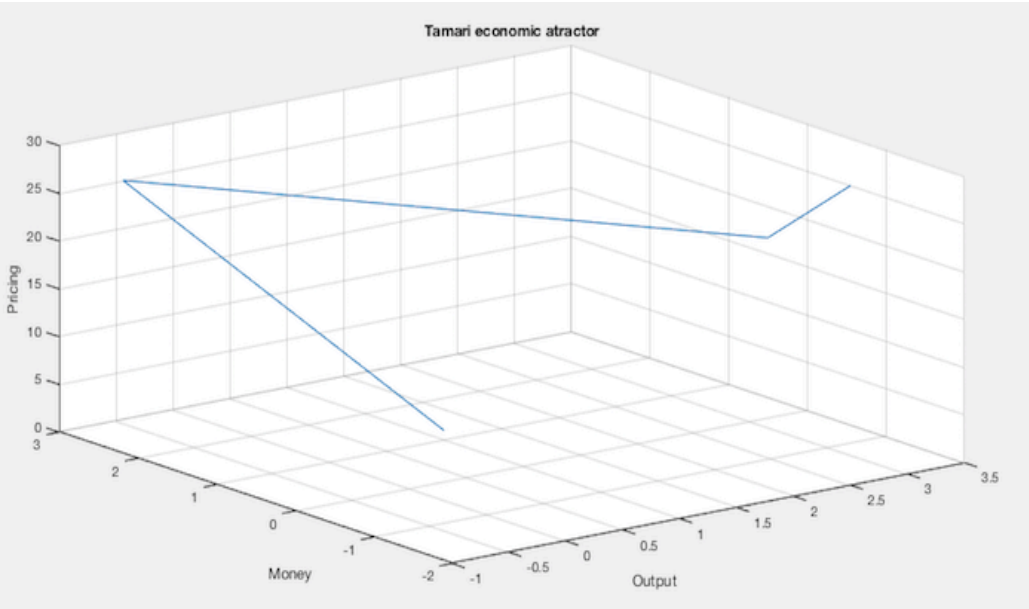
V prípade nízkej nezamestnanosti je bohatstvo konštantné a následne začne prudko klesať s nárastom HDP. S hotovosťou najskôr súmerne rastie, neskôr krivka predstavuje konštantu s nárastom hotovosti a nakoniec, aj pri poklese hotovosti, je bohatstvo prevažne konštantné.



Obrázok 31: Pokles hodnoty nezamestnanosti

(Zdroj: Vlastné spracovanie podľa: 10)

Z tohto pozorovania nám vyplýva, že 12% je vysoká nezamestnanosť, ktorá nie je prospešná pre krajinu, pretože bohatstvo krajiny v konečnom dôsledku klesá. Naopak, pri zníženej nezamestnanosti je bohatstvo krajiny prevažne konštanta.



Obrázok 32: Nárast hodnoty výmenného kurzu

(Zdroj: Vlastné spracovanie podľa: 10)

Na obrázku 32 sme zvýšili hodnotu výmenného kurzu o necelé 2 české koruny. Aj takáto malá zmena môže vyvolať veľké výkyvy atraktoru. Pre porovnanie využijeme

obrázok 29, na ktorom je zobrazený rok 2018^{+} s pôvodnou hodnotou výmenného kurzu. Táto zmena nám ukazuje, že pri vyšších hodnotách HDP narastá hotovosť aj bohatstvo. Pri nižšom kurze takáto situácia nenastáva. Naopak, bohatstvo aj hotovosť prudko klesajú.

3.2.2 Prípadové štúdie porovnania viacerých vybraných krajín

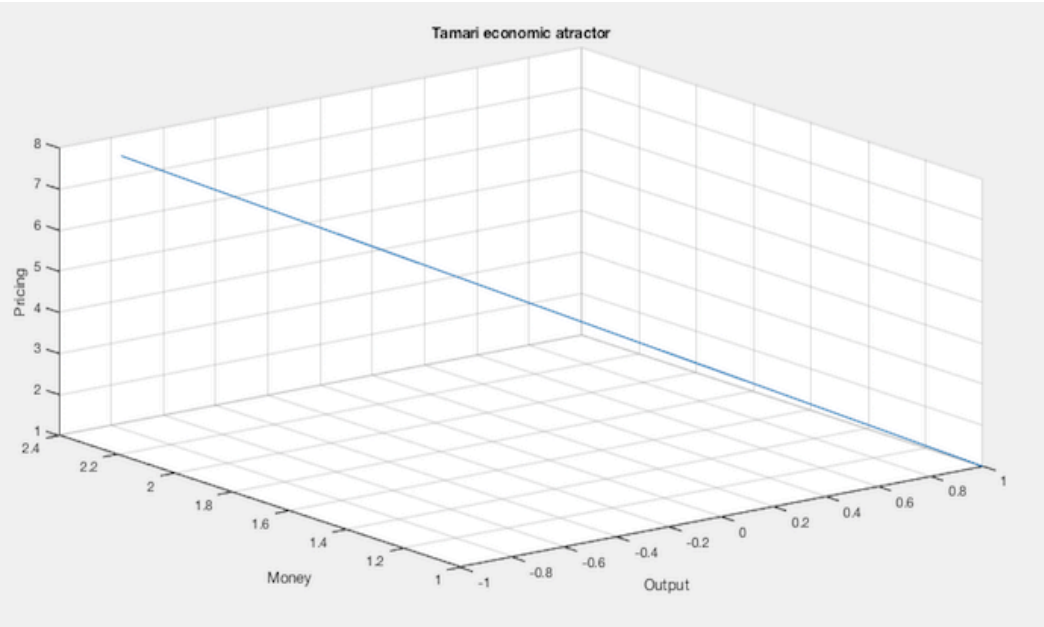
V tejto podkapitole porovnáme jednotlivé zmeny atraktoru pre ďalšie tri krajiny, pre nasledujúce roky. Tieto krajiny sú Dánsko, Nemecko a Španielsko. Dôvodom výberu týchto konkrétnych krajín je poukázanie na malé rozdiely v atraktore pre krajiny s podobnou ekonomikou, ktorými sú Dánsko a Nemecko. A naopak, výrazný rozdiel v atraktore, ktorý je vykreslený pre ekonomiku, ktorá je v krízovom stave, ako napríklad Španielsko. Nato porovnáme jednotlivé roky medzi krajinami. Hodnoty jednotlivých parametrov pre sledované krajiny sú zapísané v tabuľke 2. Tieto dáta sme získali rovnakým spôsobom ako dáta v tabuľke 1, taktiež pomocou štatistických tabuliek IFS.

Tabuľka 2 Premenné skúmaných krajín

(Zdroj: Vlastné spracovanie podľa: 17)

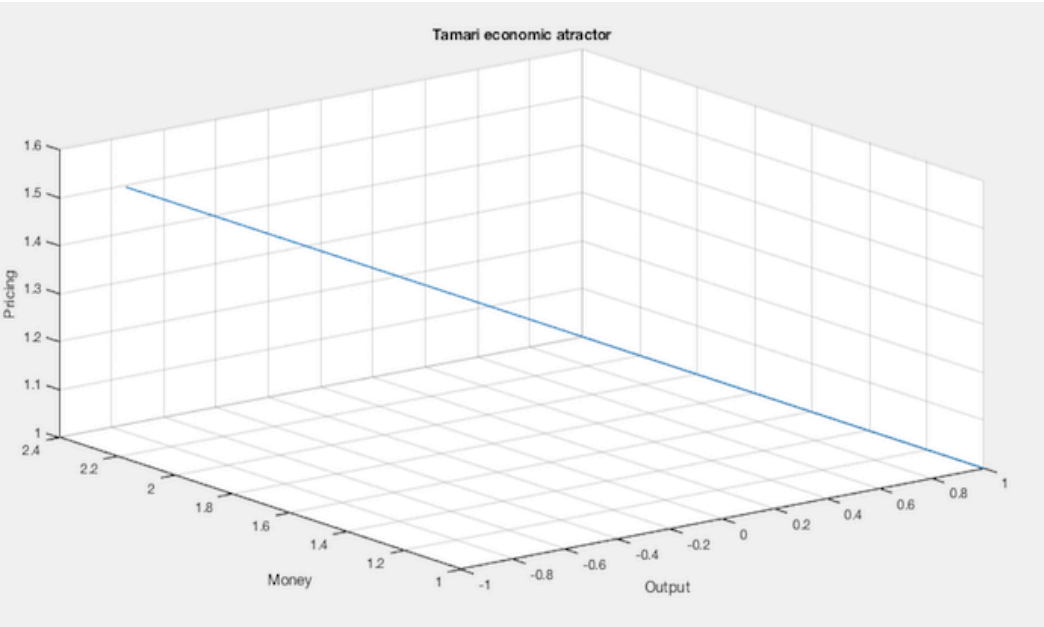
Krajina / Premenná	**Česko**	**Dánsko**	**Nemecko**	**Španielsko**
a	1.0214	1.0116	1.0128	1.0239
b	1.0347	1.0344	1.0109	1.0168
c	1.0915	1.0431	1.0396	1.0397
d	1.1042	1.0644	1.0742	1.1652
e	25.639	7.053	0.9487	0.9487
f	0.01	0.01	0.01	0.01
g	1	1	1	1
u	4.0	6.2	4.1	19.6
i	3.91	5.74	5.85	4.2

Ako prvé si ukážeme počiatočný rok, čiže rok 2016, pre jednotlivé krajiny. Rok 2016 vidíme na obrázku 33 pre Dánsko, na obrázku 34 pre Nemecko a na obrázku 35 pre Španielsko. Vo všetkých prípadoch bohatstvo klesá s rastúcim HDP a rastie s rastúcou hotovosťou. Rozdiel je len vo výške hodnôt bohatstva.



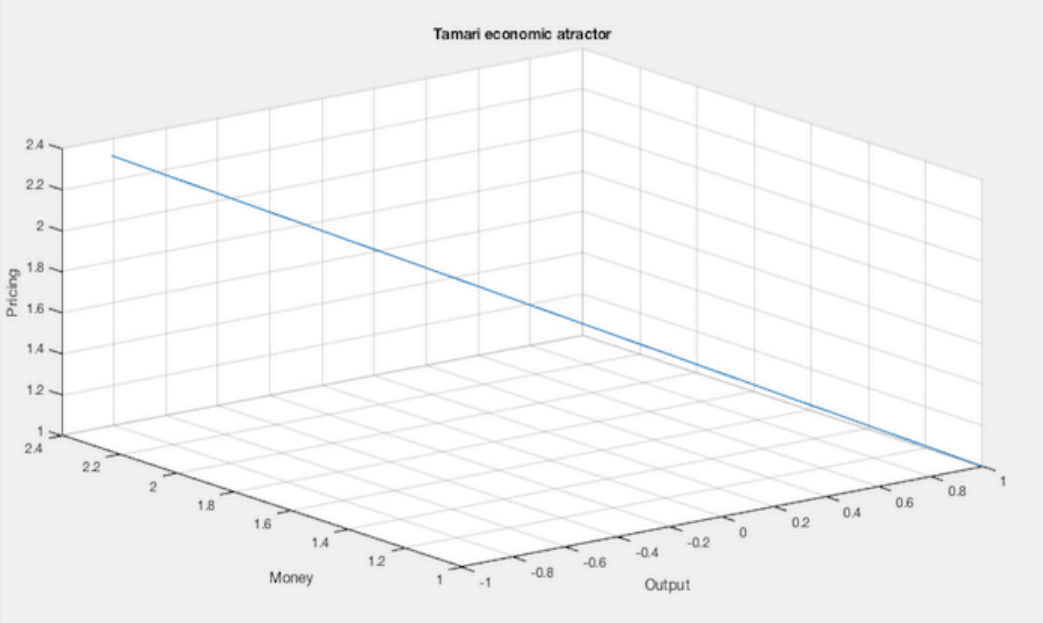
Obrázok 33: Dánsko rok 2016

(Zdroj: Vlastné spracovanie podľa: 10)



Obrázok 34: Nemecko rok 2016

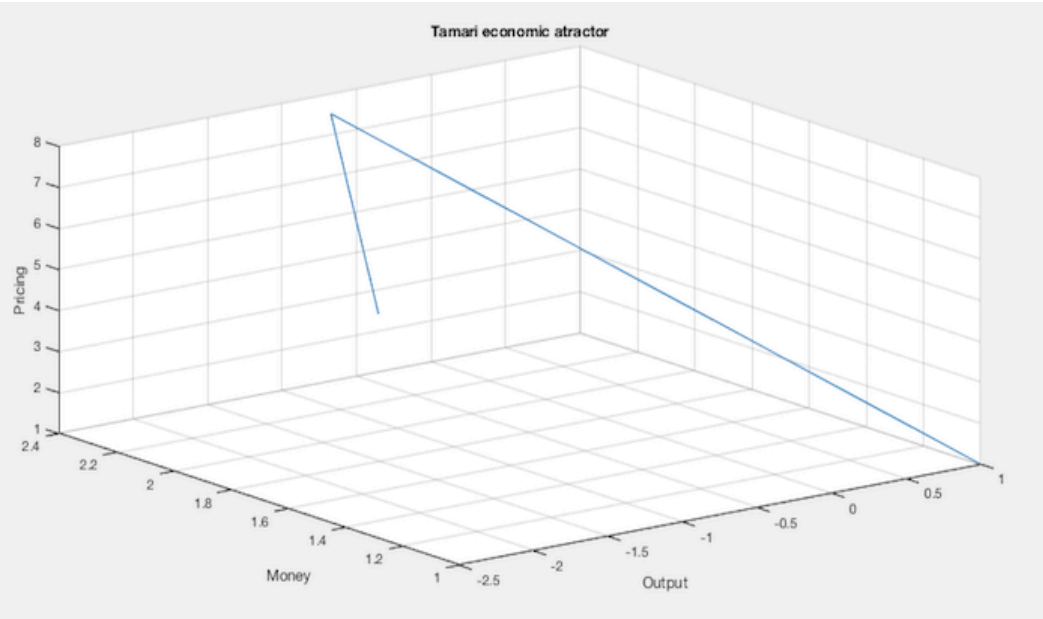
(Zdroj: Vlastné spracovanie podľa: 10)



Obrázok 35: Španielsko rok 2016

(Zdroj: Vlastné spracovanie podľa: 10)

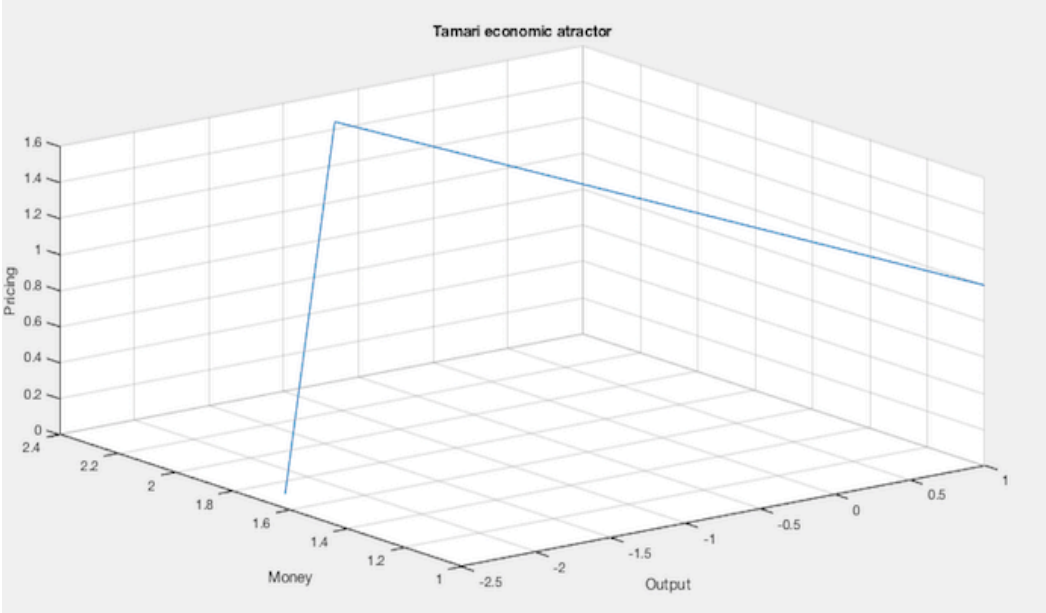
Pridáme jednu iteráciu a dostaneme predikciu pre rok 2017^{+}. Táto predikcia je viditeľná na obrázku 36 pre Dánsko, 37 pre Nemecko a 38 pre Španielsko. Sú viditeľné jasné zmeny a rozdiely medzi krajinami.



Obrázok 36: Dánsko rok 2017^{+}

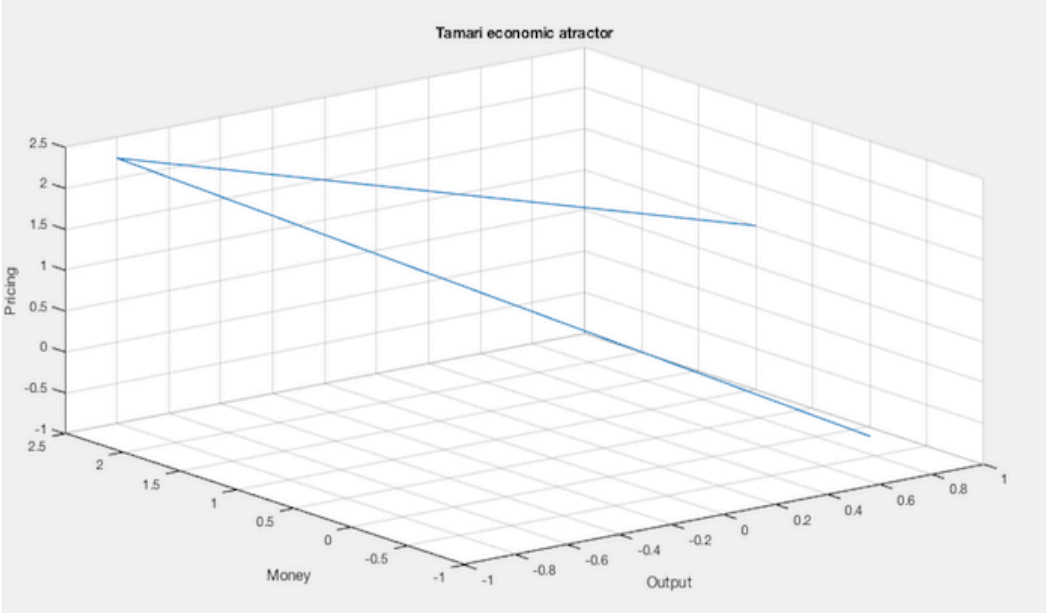
(Zdroj: Vlastné spracovanie podľa: 10)

Pre Dánsko aj Nemecko platí, že bohatstvo v roku 2017^{+} do určitej hodnoty rastie a následne začne klesať s rastúcim HDP. Taktiež rastie spolu s rastúcou hotovosťou a potom začnú klesať aj hodnoty bohatstva, aj hodnoty hotovosti. Rozdiel je znova v konkrétnych hodnotách, ale chovanie ataktoru je veľmi podobné.



Obrázok 37: Nemecko rok 2017^{+}

(Zdroj: Vlastné spracovanie podľa: 10)



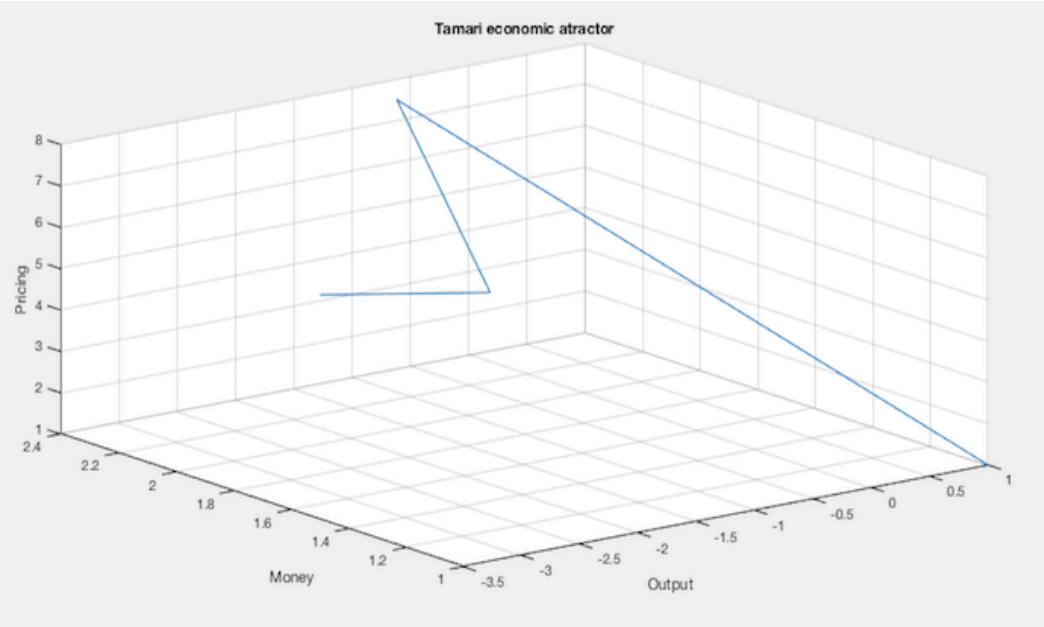
Obrázok 38: Španielsko rok 2017^{+}

(Zdroj: Vlastné spracovanie podľa: 10)

Španielsko sa odlišuje, ako môžeme vidieť na obrázku 38. Bohatstvo Španielska v roku 2017^{+} najskôr prudko rastie s klesajúcim HDP a zrazu začne prudko klesať, pričom HDP rastie. Z pohľadu bohatstvo – hotovosť je to ako pri Dánsku a Nemecku. Bohatstvo rastie spolu s hotovosťou, neskôr začnú obe prudko klesať.

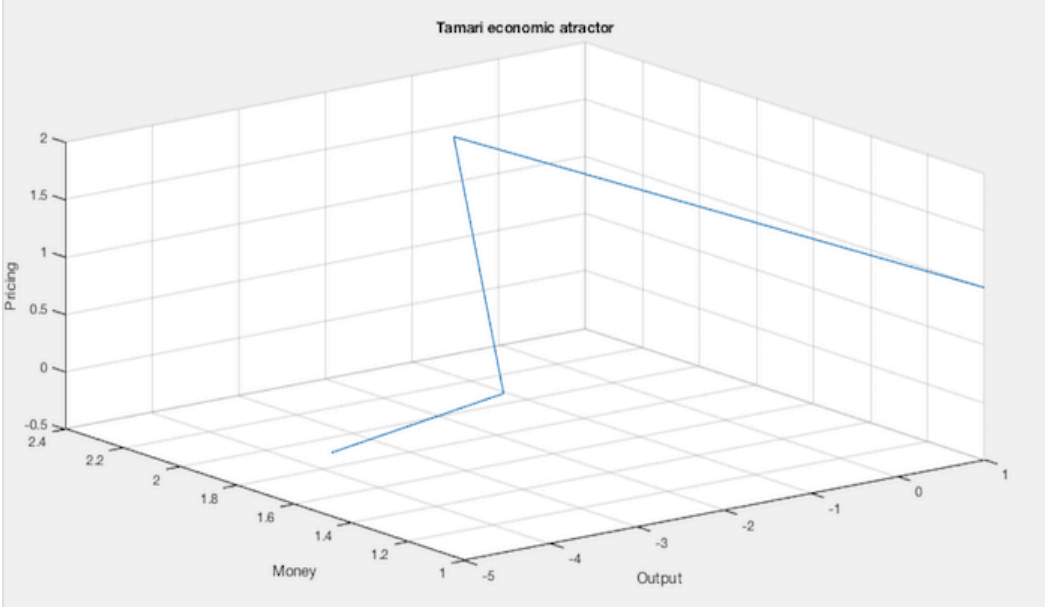
Nakoniec spravíme predikciu pre rok 2018^{+}. Zmeny v atraktoroch sú zobrazené na obrázkoch 39 pre Dánsko, 40 pre Nemecko a 41 pre Španielsko. Dánsko a Nemecko majú opäť podobné výsledky. Pri oboch bohatstvo mierne rastie s rastúcim HDP, potom začne krivka náhle rásť a nakoniec začne klesať. V tomto roku sa už z pohľadu bohatstvo – hotovosť odlišujú. Pre Dánsko platí, že bohatstvo mierne rastie zatiaľ čo hotovosť klesá, nasledovne začnú obe prudko rásť a nakoniec obe klesať. V Nemecku bohatstvo aj hotovosť rastú, potom budú obe klesať a na záver bohatstvo stále klesá zatiaľ čo hotovosť rastie.

Španielsko je stále výrazne odlišné. Bohatstvo a HDP rovnomerne klesajú, ďalej začne bohatstvo rásť a napokon krivká obráti smer, čiže bohatstvo klesá a HDP rastie. S klesajúcou hotovosťou bohatstvo tiež klesá, nato začnú hodnoty oboch rásť a napokon znova klesať.



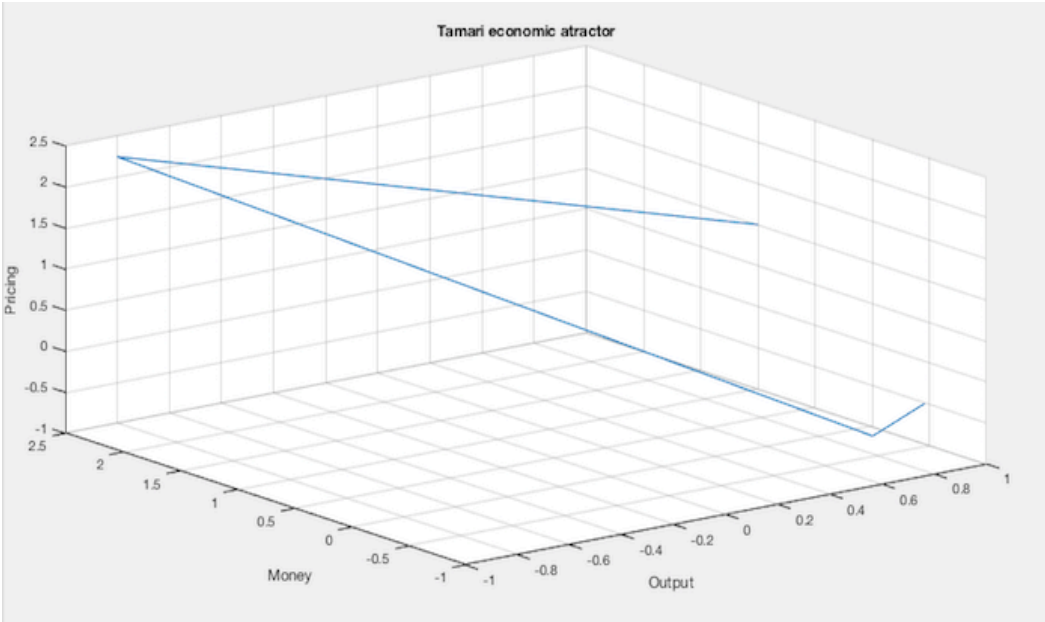
Obrázok 39: Dánsko rok 2018^{+}

(Zdroj: Vlastné spracovanie podľa: 10)



Obrázok 40: Nemecko rok 2018^{+}

(Zdroj: Vlastné spracovanie podľa: 10)



Obrázok 41: Španielsko rok 2018^{+}

(Zdroj: Vlastné spracovanie podľa: 10)

Tieto krajiny porovnáme aj s Českou republikou, ktorej zmeny v atraktore sme opísali na obrázku 28 pre rok 2017^{+} a obrázku 29 pre rok 2018^{+}, v predošlej podkapitole. Tamariho atraktor Českej republiky za rok 2017^{+} má približne rovnaký tvar ako pre Dánsko a Nemecko z pohľadu bohatstvo – HDP. Z pohľadu bohatstvo – hotovosť,

na rozdiel od Dánska a Nemecka, hodnoty oboch rastú a krivka sa nezmení. Pre rok 2018^{+} sa odlišuje od všetkých porovnávajúcich krajín.

3.2.3 Prínos riešenia

Prínos Tamariho atraktoru do ekonomiky pozostáva z možnosti predikcie zmien v krajine. Presnosť modelu sa bude dať posúdiť, keď budú verejnosti prístupné dáta pre roky, pre ktoré bola vytvorená predikcia.

Hlavným prínosom tejto bakalárskej práce je vytvorený model, ktorý nám opisuje simuláciu analýzy tzv. *what –if*, inak povedané *čo-keď* analýza. Náš model je využívaný na modelovanie a simuláciu procesov v krajine. Pre tento model boli využité dáta, ktoré sú dostupné v tabuľke 2, ktoré nebolo jednoduché získať, pretože väčšina týchto dát buď nie je verejnosti voľne dostupná, alebo sa musia hľadať medzi veľkým množstvom štatistických tabuliek. Z týchto tabuliek je potrebné vypočítať percentuálnu zmena oproti predchádzajúcemu roku, prípadne je možné hodnoty získať pomocou regresie alebo iných štatistických metód.

Záver

V bakalárskej práci sme sa zaoberali predikciou správania sa ekonomík skúmaných krajín za pomoci Tamariho atraktoru. Hlavným cieľom bolo zostavenie modelu a jeho využitie na simuláciu a analýzu. Čiastkovým cieľom bolo oboznámiť čitateľa s existenciou a problematikou chaotického Tamariho atraktoru a jeho využiteľnosťou na predikciu.

Práca je rozdelená na tri časti, ktorými sú teoretická časť, popis súčasnej situácie a vlastné návrhy riešenia. V teoretickej časti sme zhrnuli základné poznatky potrebné pre pochopenie skúmaného komplexného problému. Patrí tam priblíženie umelej inteligencie, konkrétne teórie chaosu, ktorá zahŕňa chaotické atraktory. Bližšie boli popísané aj niektoré ekonomické veličiny, ktorých znalosť bola potrebná pre jednoduchšie oboznámenie sa s matematicko-ekonomickou súvislosťou skúmaného modelu.

Jedným s cieľov práce bolo rozšírenie povedomia o Tamariho atraktore, pretože to ešte nie je dostatočne preskúmaný predikčný model. Na tento fakt nám poukazuje stručný opis časti analýzy problému a súčasnej situácie. Pre splnenie cieľa sme vytvorili pre Tamariho atraktor simuláciu pomocou vývojového prostredia MATLAB. Tvorba kódu programu je popísaná v časti návrhov riešení. Obsahuje jednotlivé časti kódu a ich detailný opis pre lepšie pochopenie jeho fungovania.

Medzi ďalšie riešenia patrí využitie programu pre vytvorenie predikcie pre Českú republiku a popísanie situácií, ktoré by mohli nastať, ak by sa náhle zmenila hodnota niektorého z ekonomických parametrov. Medzi popísané situácie patrí napríklad prudký nárast alebo pokles nezamestnanosti alebo nepatrná zmena vo výmennom kurze. Týmto sme chceli poukázať na fakt, že aj malá zmena môže vyvolať veľké výkyvy, a naopak.

Na záver môžeme konštatovať, že Tamariho atraktor stále vyžaduje podrobnejšie preskúmanie, predovšetkým preverenie stability modelu. Taktiež sa dá využiť na ďalšie prípadové štúdie, pretože je možné vytvoriť veľké množstvo kombinácií endogénnych veličín, či už konkrétnych hodnôt pre dané krajiny, alebo vymyslených hodnôt pre vytvorenie analýzy čo-keď.

Zoznam použitých zdrojov

[1] MAŘÍK, Vladimír, Jiří LAŽANSKÝ a Olga ŠTĚPÁNKOVÁ. *Umělá inteligence. [Díl] 1*. Praha: Academia, 1993. ISBN 80-200-0496-3.

[2] DOSTÁL, Petr. *Pokročilé metody rozhodování v podnikatelství a veřejné správě*. Brno: Akademické nakladatelství CERM, 2012. ISBN 978-80-7204-798-7.

[3] DOSTÁL, Petr. *Soft computing v podnikatelství a veřejné správě*. Brno: Akademické nakladatelství CERM, 2015. ISBN 978-80-7204-896-0.

[4] ČERMÁK, Jan a Luděk NECHVÁTAL. *Matematika III*. Brno: Akademické nakladatelství CERM, s.r.o. Brno, 2016 ISBN 978-80-214-5400-2.

[5] DOSTÁL, Petr. *Advanced decision making in business and public services*. Brno: Akademické nakladatelství CERM, 2011. ISBN 978-80-7204-747-5.

[6] KUČEROVÁ, Vladimíra. *Makroekonomie 1: Studijní text pro denní a kombinovanou formu studia bakalářských studijních programů*. Brno: Akademické nakladatelství CERM, 2013. ISBN 978-80-214-4798-1.

[7] KOCMANOVÁ, Alena. *Ekonomické řízení podniku*. Praha: Linde Praha, 2013. ISBN 978-80-7201-932-8.

[8] CIPRA, Tomáš. *Praktický průvodce finanční a pojistnou matematikou*. Vydání III., v Ekopressu II. Praha: Ekopress, 2015. ISBN 978-80-87865-18-7.

[9] HANSELMAN, Duane C a Bruce LITTLEFIELD. *Mastering MATLAB*. Upper Saddle River: Pearson, 2012. ISBN 9780136013303.

[10] The MathWorks, Inc. *MATLAB* [software]. © 1984 – 2016 [prístup 2017-10-01]. Dostupné z: https://www.mathworks.com/products/matlab.html

[11] Ben Tamari. *Ben Tamari - Chaos and Economics* [online]. © 2000. Posledná aktualizácia 01.04.2017 [cit. 13.01.2018]. Dostupné z: http://www.bentamari.com/

[12] DOSTÁL, Petr. *Soft computing v podnikatelství a veřejné správě: doplněk*. Brno: Akademické nakladatelství CERM, 2017. ISBN 978-80-7204-958-5.

[13] Gresham's law – definition of Gresham's law by The Free Dictionary. *Dictionary, Encyclopedia and Thesaurus – The Free Dictionary* [online]. Copyright © 2003 [cit. 19.02.2018]. Dostupné z: https://www.thefreedictionary.com/Gresham%27s+law

[14] The Keynesian Theory. *CliffsNotes Study Guides | Book Summaries, Test Preparation & Homework Help | Written by Teachers* [online]. Copyright © 2016 [cit. 23.02.2018]. Dostupné z: https://www.cliffsnotes.com/study-guides/economics/classical-and-keynesian-theories-output-employment/the-keynesian-theory

[15] Patinkin Don. *Fakulta ekonomická ZČU v Plzni* [online]. Copyright © 2007 [cit. 26.02.2018]. Dostupné z: https://www.dfek.zcu.cz/kalendarium/EKONOM/Patinkin_d.pdf

[16] Quantity Theory of Money by Friedman. *Share Your Knowledge on Economics | Economics Discussion* [online]. Dostupné z: http://www.economicsdiscussion.net/economic-theories/quantity-theory-of-money-by-friedman/21120

[17] IMF Statistics Department *International Financial Statistics* [online]. Washington, D.C.: International Monetary Fund, 2017 [cit. 2018-04-05]. ISBN 978-1-47559-536-9. Dostupné z: http://data.imf.org/

Zoznam použitých obrázkov

Obrázok 1: Bodový atraktor (Zdroj: vlastný) 14
Obrázok 2: Cyklický atraktor (Zdroj: vlastný) 15
Obrázok 3: Chaotický atraktor (Zdroj: Vlastný) 15
Obrázok 4: Chaotický atraktor v ekonomickom systéme (Zdroj: Vlastný) 16
Obrázok 5: Nezamestnanosť (Zdroj: Vlastné spracovanie podľa: 6) 20
Obrázok 6: MATLAB R2016a (Zdroj: Vlastné spracovanie podľa: 10) 24
Obrázok 7: Hierarchia objektov (Zdroj: Vlastné spracovanie podľa: 9) 26
Obrázok 8: GUI (Zdroj: Vlastné spracovanie podľa: 10) 34
Obrázok 9: Vlastnosti (Zdroj: Vlastné spracovanie podľa: 10) 35
Obrázok 10: Počiatočný kód (Zdroj: Vlastné spracovanie podľa: 10) 36
Obrázok 11: Otváracia funkcia (Zdroj: Vlastné spracovanie podľa: 10) 36
Obrázok 12: Počiatočný graf (Zdroj: Vlastné spracovanie podľa: 10) 36
Obrázok 13: Set (Zdroj: Vlastné spracovanie podľa: 10) 37
Obrázok 14::Automatické funkcie (Zdroj: Vlastné spracovanie podľa: 10) 37
Obrázok 15: runODE (Zdroj: Vlastné spracovanie podľa: 10) 37
Obrázok 16: Úvodný Tamariho atraktor pri spustení (Zdroj: Vlastné spracovanie podľa: 10) 38
Obrázok 17: Kontrola aktivácie okna (Zdroj: Vlastné spracovanie podľa: 10) 39
Obrázok 18: Case (Zdroj: Vlastné spracovanie podľa: 10) 39
Obrázok 19: Rovnice Tamariho atraktoru (Zdroj: Vlastné spracovanie podľa: 10) . 40
Obrázok 20: Vykreslenie grafu (Zdroj: Vlastné spracovanie podľa: 10) 40
Obrázok 21: runODE2 (Zdroj: Vlastné spracovanie podľa: 10) 40
Obrázok 22: Slider (Zdroj: Vlastné spracovanie podľa: 10) 41
Obrázok 23: Varovné okno (Zdroj: Vlastné spracovanie podľa: 10) 41
Obrázok 24: Edit box (Zdroj: Vlastné spracovanie podľa: 10) 42
Obrázok 25: Edit box create function (Zdroj: Vlastné spracovanie podľa: 10) 42
Obrázok 26: Tlačidlo pre rotáciu (Zdroj: Vlastné spracovanie podľa: 10) 43
Obrázok 27: Česko rok 2016 (Zdroj: Vlastné spracovanie podľa: 10) 44
Obrázok 28: Česko rok 2017$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 45
Obrázok 29: Česko rok 2018$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 46

Obrázok 30: Nárast hodnoty nezamestnanosti (Zdroj: Vlastné spracovanie podľa: 10) 47
Obrázok 31: Pokles hodnoty nezamestnanosti (Zdroj: Vlastné spracovanie podľa: 10) 48
Obrázok 32: Nárast hodnoty výmenného kurzu (Zdroj: Vlastné spracovanie podľa: 10) 48
Obrázok 33: Dánsko rok 2016 (Zdroj: Vlastné spracovanie podľa: 10) 50
Obrázok 34: Nemecko rok 2016 (Zdroj: Vlastné spracovanie podľa: 10) 50
Obrázok 35: Španielsko rok 2016 (Zdroj: Vlastné spracovanie podľa: 10) 51
Obrázok 36: Dánsko rok 2017$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 51
Obrázok 37: Nemecko rok 2017$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 52
Obrázok 38: Španielsko rok 2017$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 52
Obrázok 39: Dánsko rok 2018$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 53
Obrázok 40: Nemecko rok 2018$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 54
Obrázok 41: Španielsko rok 2018$^+$ (Zdroj: Vlastné spracovanie podľa: 10) 54

Zoznam použitých tabuliek

Tabuľka 1 Česká republika premenné (Zdroj: Vlastné spracovanie podľa: 17)..... 44

Tabuľka 2 Premenné skúmaných krajín (Zdroj: Vlastné spracovanie podľa: 17) 49

Zoznam príloh

Príloha 1: Kód pre Tamariho atraktor v MATLAB I

Prílohy

Príloha 1: Kód pre Tamariho atraktor v MATLAB

```
function varargout = Tam(varargin)

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Tam_OpeningFcn, ...
                   'gui_OutputFcn',  @Tam_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Tam is made visible.
function Tam_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
global A B C D E F G U I N a b c d e f g u i n
a = -0.587;b = -0.011;c = 0.02; d = 0.96;e = 0;f = 0.01;g = 1;
u = 0.05;i = 0.1; n = 100;
A = 0; B = 1; C = 2; D = 3; E = 4; F = 5; G = 6; U = 7; I = 8; N = 9;
runODE(hObject,eventdata,handles,false,50);
set(handles.edita,'String',num2str(a));
set(handles.editb,'String',num2str(b));
set(handles.editc,'String',num2str(c));
set(handles.editd,'String',num2str(d));
set(handles.edite,'String',num2str(e));
set(handles.editf,'String',num2str(f));
set(handles.editg,'String',num2str(g));
set(handles.editu,'String',num2str(u));
set(handles.editi,'String',num2str(i));
set(handles.editn,'String',num2str(n));
% Update handles structure
guidata(hObject, handles);


% --- Outputs from this function are returned to the command line.
function varargout = Tam_OutputFcn(hObject, eventdata, handles)
```

```
% Get default command line output from handles structure
varargout{1} = handles.output;

% function for edit box
function runODE(hObject,eventdata,handles,isWinActive,callbackType)

S = [1 1 1 1 1 1 1 1 1 1]; % slider strength vector

% control of the window activation (for the first graph, when opened)
if (isWinActive)
    m = get(hObject,'Value');
else
    m = 0.1;
end

% storing the change of values - graph change
global A B C D E F G U I N a b c d e f g u i n
switch callbackType
    case A
        a = m*S(1);
    case B
        b = m*S(2);
    case C
        c = m*S(3);
    case D
        d = m*S(4);
    case E
        e = m*S(5);
    case F
        f = m*S(6);
    case G
        g = m*S(7);
    case U
        u = m*S(8);
    case I
        i = m*S(9);
    case N
        n = m*S(10);
    otherwise

end

% Tamari attractor differential equations
x(1)=1;y(1)=1;z(1)=1;
for j=1:n
    x(j+1)=(x(j)-a*y(j))*cos(z(j))-b*y(j)*sin(z(j));
    y(j+1)=(x(j)+c*y(j))*sin(z(j))+d*y(j)*cos(z(j));
    z(j+1)=e+f*z(j)+g*atan((1-u)/(1-i)*x(j)*y(j));
end
plot3(handles.axes1,x,y,z);
grid on; xlabel('x'); ylabel('y');ylabel('z');
```

```
title('Tamari economic
atractor');xlabel('Output');ylabel('Money');zlabel('Pricing');

% function for edit box
function runODE2(hObject,eventdata,handles,isWinActive,callbackType)

S = [1 1 1 1 1 1 1 1 1 1]; % slider strength vector

% control of the window activation (for the first graph, when opened)
if (isWinActive)
    m = str2num(get(hObject,'String'));
else
    m = 0.1;
end

% storing the change of values - graph change
global A B C D E F G U I N a b c d e f g u i n
switch callbackType
    case A
        a = m*S(1);
    case B
        b = m*S(2);
    case C
        c = m*S(3);
    case D
        d = m*S(4);
    case E
        e = m*S(5);
    case F
        f = m*S(6);
    case G
        g = m*S(7);
    case U
        u = m*S(8);
    case I
        i = m*S(9);
    case N
        n = m*S(10);
    otherwise

end

% Tamari attractor differential equations
x(1)=1;y(1)=1;z(1)=1;
for j=1:n
    x(j+1)=(x(j)-a*y(j))*cos(z(j))-b*y(j)*sin(z(j));
    y(j+1)=(x(j)+c*y(j))*sin(z(j))+d*y(j)*cos(z(j));
    z(j+1)=e+f*z(j)+g*atan((1-u)/(1-i)*x(j)*y(j));
end
plot3(handles.axes1,x,y,z);
grid on; xlabel('x'); ylabel('y');ylabel('z');
```

```
title('Tamari economic
atractor');xlabel('Output');ylabel('Money');zlabel('Pricing');

% slider a
function slidera_Callback(hObject, eventdata, handles)
global A
runODE(hObject,eventdata,handles,true,A);
global a
set(handles.edita,'String',num2str(a));
guidata(hObject,handles);

function slidera_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% edit box a
function edita_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.edita,'String'));
if Theta < -10
    Theta = -10;
    errordlg('Minimum nesmie byt mensie ako -10','File Error');
elseif Theta > 10
    Theta = 10;
    errordlg('Maximum nesmie byt viac ako 10','File Error');
end
% Set slider value.
set(handles.slidera, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.edita, 'String', caption);
global A a
runODE2(hObject,eventdata,handles,true,A);
set(handles.slidera,'Value',a);
guidata(hObject,handles);

function edita_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box b
function sliderb_Callback(hObject, eventdata, handles)
global B
runODE(hObject,eventdata,handles,true,B);
global b
set(handles.editb,'String',num2str(b));
guidata(hObject,handles);
```

```
function sliderb_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editb_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editb,'String'));
if Theta < -1
    Theta = -1;
    errordlg('Minimum nesmie byt mensie ako -1','File Error');
elseif Theta > 2
    Theta = 2;
    errordlg('Maximum nesmie byt viac ako 2','File Error');
end
% Set slider value.
set(handles.sliderb, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editb, 'String', caption);
global B b
runODE2(hObject,eventdata,handles,true,B);
set(handles.sliderb,'Value',b);
guidata(hObject,handles);

function editb_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box c
function sliderc_Callback(hObject, eventdata, handles)
global C
runODE(hObject,eventdata,handles,true,C);
global c
set(handles.editc,'String',num2str(c));
guidata(hObject,handles);

function sliderc_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editc_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editc,'String'));
if Theta < 0
    Theta = 0;
    errordlg('Minimum nesmie byt mensie ako 0','File Error');
elseif Theta > 2
```

```
    Theta = 2;
    errordlg('Maximum nesmie byt viac ako 2','File Error');
end
% Set slider value.
set(handles.sliderc, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editc, 'String', caption);
global C c
runODE2(hObject,eventdata,handles,true,C);
set(handles.sliderc,'Value',c);
guidata(hObject,handles);

function editc_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box d
function sliderd_Callback(hObject, eventdata, handles)
global D
runODE(hObject,eventdata,handles,true,D);
global d
set(handles.editd,'String',num2str(d));
guidata(hObject,handles);

function sliderd_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editd_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editd,'String'));
if Theta < -1
    Theta = -1;
    errordlg('Minimum nesmie byt mensie ako -1','File Error');
elseif Theta > 2
    Theta = 2;
    errordlg('Maximum nesmie byt viac ako 2','File Error');
end
% Set slider value.
set(handles.sliderd, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editd, 'String', caption);
global D d
runODE2(hObject,eventdata,handles,true,D);
set(handles.sliderd,'Value',d);
guidata(hObject,handles);
```

```
function editd_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box e
function slidere_Callback(hObject, eventdata, handles)
global E
runODE(hObject,eventdata,handles,true,E);
global e
set(handles.edite,'String',num2str(e));
guidata(hObject,handles);

function slidere_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function edite_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.edite,'String'));
if Theta < -2
    Theta = -2;
    errordlg('Minimum nesmie byt mensie ako -2','File Error');
elseif Theta > 30
    Theta = 30;
    errordlg('Maximum nesmie byt viac ako 2','File Error');
end
% Set slider value.
set(handles.slidere, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.edite, 'String', caption);
global E e
runODE2(hObject,eventdata,handles,true,E);
set(handles.slidere,'Value',e);
guidata(hObject,handles);

function edite_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box f
function sliderf_Callback(hObject, eventdata, handles)
global f
set(handles.editf,'String',num2str(f));
```

```
guidata(hObject,handles);
global F
runODE(hObject,eventdata,handles,true,F);

function sliderf_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editf_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editf,'String'));
if Theta < -0.5
    Theta = -0.5;
    errordlg('Minimum nesmie byt mensie ako -0.5','File Error');
elseif Theta > 1
    Theta = 1;
    errordlg('Maximum nesmie byt viac ako 1','File Error');
end
% Set slider value.
set(handles.sliderf, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editf, 'String', caption);
global F f
runODE2(hObject,eventdata,handles,true,F);
set(handles.sliderf,'Value',f);
guidata(hObject,handles);

function editf_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box g
function sliderg_Callback(hObject, eventdata, handles)
global G
runODE(hObject,eventdata,handles,true,G);
global g
set(handles.editg,'String',num2str(g));
guidata(hObject,handles);

function sliderg_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editg_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editg,'String'));
```

```
if Theta < -2
    Theta = -2;
    errordlg('Minimum nesmie byt mensie ako -2','File Error');
elseif Theta > 2
    Theta = 2;
    errordlg('Maximum nesmie byt viac ako 2','File Error');
end
% Set slider value.
set(handles.sliderg, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editg, 'String', caption);
global G g
runODE2(hObject,eventdata,handles,true,G);
set(handles.sliderg,'Value',g);
guidata(hObject,handles);

function editg_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box u
function slideru_Callback(hObject, eventdata, handles)
global U
runODE(hObject,eventdata,handles,true,U);
global u
set(handles.editu,'String',num2str(u));
guidata(hObject,handles);

function slideru_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editu_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editu,'String'));
if Theta < -1
    Theta = -1;
    errordlg('Minimum nesmie byt mensie ako -1','File Error');
elseif Theta > 20
    Theta = 20;
    errordlg('Maximum nesmie byt viac ako 20','File Error');
end
% Set slider value.
set(handles.slideru, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editu, 'String', caption);
```

```
global U u
runODE2(hObject,eventdata,handles,true,U);
set(handles.slideru,'Value',u);
guidata(hObject,handles);

function editu_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% slider and edit box i
function slideri_Callback(hObject, eventdata, handles)
global I
runODE(hObject,eventdata,handles,true,I);
global i
set(handles.editi,'String',num2str(i));
guidata(hObject,handles);

function slideri_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function editi_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editi,'String'));
if Theta < -1
    Theta = -1;
    errordlg('Minimum nesmie byt mensie ako -1','File Error');
elseif Theta > 15
    Theta = 15;
    errordlg('Maximum nesmie byt viac ako 15','File Error');
end
% Set slider value.
set(handles.slideri, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.4f',Theta);
set(handles.editi, 'String', caption);
global I i
runODE2(hObject,eventdata,handles,true,I);
set(handles.slideri,'Value',i);
guidata(hObject,handles);

function editi_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
% slider and edit box n
function slidern_Callback(hObject, eventdata, handles)
global N
runODE(hObject,eventdata,handles,true,N);
global n
set(handles.editn,'String',num2str(n));
guidata(hObject,handles);

function slidern_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end


function editn_Callback(hObject, eventdata, handles)
Theta=str2double(get(handles.editn,'String'));
if Theta < 0
    Theta = 0;
    errordlg('Minimum nesmie byt mensie ako 0','File Error');
elseif Theta > 100
    Theta = 100;
    errordlg('Maximum nesmie byt viac ako 100','File Error');
end
% Set slider value.
set(handles.slidern, 'Value', Theta);
% Set static text caption of slider.
caption = sprintf('%.0f',Theta);
set(handles.editn, 'String', caption);
global N n
runODE2(hObject,eventdata,handles,true,N);
set(handles.slidern,'Value',n);
guidata(hObject,handles);

function editn_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```