

UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE
FAKULTA PRÍRODNÝCH VIED

SIMULÁCIE MODELOV TYPU DRAVEC-KORISŤ
DIPLOMOVÁ PRÁCA

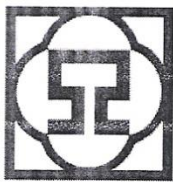
2018

Bc. Zdenko Hruška

UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE
FAKULTA PRÍRODNÝCH VIED

SIMULÁCIE MODELOV TYPU DRAVEC-KORISŤ
DIPLOMOVÁ PRÁCA

Študijný odbor: 9.2.9 Aplikovaná informatika
Študijný program: Aplikovaná informatika
Školiace pracovisko: Katedra informatiky
Školiteľ: Mgr. Martin Vozár, PhD.



Univerzita Konštantína Filozofa v Nitre
Fakulta prírodných vied

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Zdenko Hruška
Študijný program: Aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: 9.2.9 aplikovaná informatika
Typ záverečnej práce: Diplomová práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Simulácie modelov typu dravec-korist'

Anotácia: Tvorba matematických modelov spoločenstiev živočíchov je dôležitá. Pomáha napríklad pri začleňovaní jedincov do ekosystémov alebo ich reguláciu. Pomocou nich tak vieme simulovať vývin jedincov v ich prostredí.
Cieľ diplomovej práce: vo zvolenom systéme simulovať rôzne druhy matematických modelov typu dravec-korist'.
Požiadavky na vedomosti a zručnosti študenta: pokročilé vedomosti a zručnosti z programovania v systéme MATLAB (Simulink), dobrá znalosť angličtiny a matematiky.

Školiteľ: Mgr. Martin Vozár, PhD.
Oponent: doc. RNDr. Michal Munk, PhD.
Katedra: KI - Katedra informatiky

Dátum zadania: 30.09.2016

Dátum schválenia: 10.10.2016

prof. Ing. Milan Turčáni, CSc.
schválil/a

Pod'akovanie

V prvom rade by som chcel pod'akovať Mgr. Martinovi Vozárovi, PhD. za pomoc, odborné vedenie, cenné rady a pripomienky. Tiež ďakujem za váš čas a ochotu.

Pod'akovanie patrí aj mojej rodine a blízkym, ktorí mi boli oporou počas celého štúdia.

ABSTRAKT

HRUŠKA, Zdenko: Simulácie modelov typu Dravec-korist'. [Diplomová práca]. Univerzita Konštantína Filozofa v Nitre. Fakulta prírodných vied. Školiteľ: Mgr. Martin Vozár, PhD. Stupeň odbornej kvalifikácie: Magister Aplikovanej informatiky. Nitra: FPV, 2018. 63 s.

V ekosystéme majú dravce nenahraditeľnú úlohu, regulujú totiž populáciu koristi, ktorá by sa premnožila. V predkladanej diplomovej práci sa zaoberáme simuláciou modelov typu Dravec-korist'. Cieľom práce je vo zvolenom programe simulovať rôzne druhy matematických modelov typu dravec-korist'. Teoretická časť práce sa venuje pojmom týkajúcich sa modelovania, tvorby modelov a ich simulácie. V praktickej časti sa zaoberáme konkrétne matematickými modelmi, ktoré sme si podrobne rozobrali a následne pozorovali, ako sa správa populácia dravca a koristi. Chceme čo najlepšie predpovedať budúci vývoj či zmenu veľkosti populácii dravca a koristi. Modely sme vytvorili pomocou matematického výpočtového systému Matlab, ktorému sa venuje v teoretickej časti diplomovej práce.

Kľúčové slová: Dravec. Korist'. Modelovanie. Simulácia. Matlab.

ABSTRACT

HRUŠKA, Zdenko: Models of Type Predator-prey Simulation. [Diploma thesis]. Constantine the Philosopher University in Nitra. Faculty of Natural Sciences. Supervisor: Mgr. Martin Vozár, PhD. Degree of academic qualification: Master of Applied Informatics. Nitra: FPV, 2018. 63 p.

Predators have an irreplaceable role in the ecosystem. They regulate the population of prey that could overgrow. In this diploma thesis we deal with the simulation of the predator-prey models. The aim of the thesis is to simulate various kinds of mathematical predator-prey models in the selected programme. The theoretical part of the thesis defines the concepts of modeling, the creation of models and their simulation. The empirical part focuses specifically on mathematical models that we have analyzed in detail and later monitored how does the population of predators and preys behave. In the thesis we want to give the best prediction of the future development or the changes in size of the predator and prey population. The models were created with the help of mathematical computing system called Matlab. The description of the system is provided in the theoretical part of the thesis.

Key words: Predator. Prey. Modelling. Simulation. Matlab.

Obsah

Úvod	7
1 Modelovanie a simulácia	10
1.1 Typy modelov	11
1.1.1 Modelovanie	11
1.2 Matematické modelovanie pomocou rovníc	12
1.3 Diskrétny čas a rekurentné rovnice	13
1.4 Spojitý čas a diferenciálne rovnice	15
1.5 Systémová dynamika	17
1.5.1 Základné prvky grafického znázornenia	18
1.5.2 Vyjadrenie pomocou rovníc	18
1.6 Základné vzory správania	19
2 Ciele diplomovej práce	21
3 Matlab	22
4 Funkčné odpovede Hollingovho typu	28
5 Modely rastu populácie	31
5.1 Zmena parametrov	34
6 Model dravec-Korist'	38
6.1 Zmena parametrov	42
7 Model typu jeden dravec, dve koristi	46
7.1 Zmena parametrov	47
8 Rosenzweig-MacArthur model s funkčnou odpoveďou Hollingovho II. typu	49
8.1 Zmena parametrov	50
9 Model Nicholson-Bailey	52
10 Allee efekt	55
10.1 Silný a slabý allee efekt	55
Záver	57
Zoznam bibliografických odkazov	59
Zoznam príloh	62

ÚVOD

Nielen v prírode, ale aj v ľudskom živote platí od nepamäti zákon prežitia. Kto chce prežiť musí byť silný, vytrvalý, odvážny, sebaistý. Kým ľudia môžu rozhodnúť o tom či budú „šelmou“, alebo „úlovkom“ v zvieracej ríši je to už zariadené dopredu. Tam máme presne určené, ktoré zvieratá sú dravcami a naopak tie, ktoré sú korisťou týchto dravcov. Nikdy sa nemôže stať, že by zajac bol dravcom a liška jeho korisťou. Takto to totiž nefunguje. Zajac bude stále korisťou a liška dravcom. Sú však aj situácie, kedy je zviera nielen dravcom, ale aj korisťou. Zoberme si takého hada. Hada je dravcom, ktorý je nebezpečný pre mnoho zvierat. Avšak aj had cíti ohrozenie od iného zvieraťa, napríklad od operených dravcov. Je teda zjavné, že situácií a možností, kto je dravcom, a kto je naopak korisťou, je v zvieracej ríši mnoho. Jedno je však isté, dravec nedokáže žiť bez svojej koristi. Čo sa stane, keď je koristi málo? A čo ak je dravcov viac, ako by malo byť? Tak aj na tieto otázky poskytujeme odpovede v predkladanej diplomovej práci. Všetky tieto biologické faktory prekladáme do jazyka informatiky a prinášame ich v podobe modelov. Základná štruktúra našej práce pozostáva z desiatich hlavných kapitol, ktoré ďalej rozvíjame a dopĺňame v jednotlivých podkapitolách.

Prvá kapitola s názvom *Modelovanie a simulácia* obsahuje základné informácie o modelovaní a simulácii, ktoré sú východiskové v našej práci. Pri tvorbe tejto kapitoly pracujeme s knihou *Modelování a simulace komplexních systémů*, ktorej autorom je *Radek Pelánek*. Táto kapitola obsahuje šesť podkapitol. V prvej predstavujeme niekoľko typov modelov a následne sa venujeme samotnému procesu modelovania, kde opisujeme jednotlivé fázy tohto procesu, ako prebieha, s čím je späté, z čoho vychádza, čo je potrebné na jeho realizáciu, a aký je jeho výsledok. V ostatných predstavujeme matematické modelovanie pomocou rovníc, diskrétny čas a rekurentné rovnice, spojitý čas a diferenciálne rovnice, systémovú dynamiku a v neposlednom rade predstavujeme aj základné vzory správania.

Druhá kapitola približuje ciele a podciele, ktoré sa snažíme naplniť. Hlavným cieľom je v zvolenom programe simulovať rôzne druhy matematických modelov typu dravec-korisť a opísať tieto modely.

Program, v ktorom pracujeme a vytvárame tieto modely sa nazýva *Matlab*. V tretej kapitole približujeme tento program. Okrem historických údajov o jeho vzniku a zakladateľoch, prinášame informácie aj o programovacích jazykoch, ktoré program používa a tiež predstavujeme jeho vlastnosti. Práca v *Matlabe* je mnohotvárna. Pomocou

tohto programu dokážeme okrem iného vytvoriť monitorovacie systémy v zdravotníctve, robotické sondy vesmírneho programu a tiež sme schopný vytvoriť matematický model typu dravec-korist'. V tejto kapitole systematicky predstavujeme jednotlivé kroky, ktoré sú späté s prácou v spomínanom programe a bezprostredne súvisia s tvorením modelov typu dravec-korist'. Tu už naznačujeme postup našej praktickej práce, ktoré súvisí s vytváraním jednotlivých modelov.

Hollingove typy funkčných odpovedí, ktoré popisujú mieru konzumácie jedného dravca vzhľadom k hustote koristi, predstavujeme v štvrtej kapitole. Rozoberieme tri typy funkčných odpovedí, ktoré zaviedol v roku 1959 kanadský ekológ a profesor Crawford Stanley Holling. Priblížime, prečo funkčné odpovede dosahujú určitú hornú hranicu a vďaka čomu prišiel Holling na tvar rovnice.

Modelom rastu populácie sa budeme zaoberať v piatej kapitole našej práce. Pod týmto názvom si môžeme predstaviť tvorbu matematického modelu, ktorým chceme predpovedať budúci vývoj či zmenu veľkosti populácii dravca a koristi.

V nasledujúcej kapitole našej diplomovej práce sa zaoberáme základným modelom Lotka-Voltera dravec-korist'. Na úvod predstavujeme zakladateľov tohto modelu, po ktorých je pomenovaný a tiež opíšeme proces, ktorý podmienil jeho vznik. Ďalej uvádzame matematické rovnice modelu dravec-korist'. V podkapitole zmena parametrov upravujeme základný model, s ktorým pracujeme. Postupne si na štyroch prípadoch ukážeme, ako sa menia jednotlivé parametre A, B, C, D, ktoré nám opisujú vzájomne pôsobenie dravcov a koristi, a snažíme sa predpovedať budúci vývoj populácii dravec-korist'.

Model s jedným dravcom a dvoma korisťami predstavíme v siedmej kapitole. Pri tvorbe tejto kapitoly sme čerpali z materiálu *Predator-Two Prey*, ktorej autorom je emeritný profesor matematiky *Robert E. White*. Pracujeme tu podobne ako pri predchádzajúcich modeloch s cieľom zistiť, ako sa bude meniť vývoj populácii týchto troch druhov, v našom prípade líšok, zajacov a sliepok.

Po Michaelovi Rosenzweigovi a Robertovi MacArthurovi je pomenovaný model, s ktorým sa stretávame v ôsmej kapitole. Rosenzweig-MacArthur model dravec-korist' použijeme s funkčnou odpoveďou Hollingovho II typu, ktoré sme predstavili v štvrtej kapitole. V tomto modeli sa stretávame s paradoxom obohatenia.

Hostiteľ je organizmus, v ktorom parazit žije. Iba v tomto hostiteľovi existujú predpoklady pre ďalšie rozmnožovanie parazita. Vývojový cyklus parazita sa

v hostiteľovi začína aj končí. Vývoj týchto dvoch druhom predstavujeme v deviatej kapitole - model Nicholson-Bailey (hostiteľ a parazit).

Posledná kapitola nesie názov Alleeho efekt, v ktorom predstavíme fenomén v oblasti biológie, ktorý nesie názov po Warderovi Clyde Alleem. Alleeho efektom nazývame jav, keď sa so znižujúcou veľkosťou alebo hustotou znižuje aj špecifická miera rastu, poprípade populácia až vymiera. Za vznik Alleeho efektu môže niekoľko mechanizmov.

Informácie a vedomosti k práci čerpáme s odbornej literatúry a internetových článkov, ktoré sa zaoberajú touto problematikou.

1 MODELOVANIE A SIMULÁCIA

V tejto kapitole sa venujeme modelovaniu a simulácii. Modelovanie je proces, ktorého výsledkom je model, ktorý predstavuje zjednodušenie reality. Nevytvárame najpresnejší model, ale zjednodušujeme ho zo zámerom, aby sme lepšie pochopili realitu. Pri simulácii sa snažíme spraviť model, čo najpresnejšie a pokúšame sa ho dostať „do pohybu“. Následne pozorujeme správanie tohto modelu v rôznych situáciách.

Vďaka modelom vieme robiť predpovede, odhadovať vplyv rôznych zásahov do systému a lepšie plánovať akcie. Okrem týchto konkrétnych výsledkov sú však modely veľmi dôležité aj na mentálnej úrovni - samotná práca s modelmi výrazne ovplyvňuje náš spôsob myslenia a nazeranie na svet. Kým modely nás nútia formulovať naše hmlisté predstavy, simulácia nás už priamo konfrontuje s dôsledkami týchto predstáv. Modelovanie a simulácia tiež umožňujú zdieľanie myšlienok a toľko potrebné odovzdávanie informácií medzi odborníkmi z rôznych odborov.

Ľudia už dlhšie používajú modelovanie a simuláciu. Kým v počiatkoch to bolo väčšinou v technických odboroch, príchod výpočtovej techniky umožnil rozšírenie skúmaných obzorov dvoma spôsobmi: k dispozícii máme oveľa viac dát a tiež vieme tieto dáta lepšie spracovávať a modelovať.

Opísať správanie celku je však veľmi komplikované, nestačia nám k tomu porovnania a rovnice s niekoľkými premennými. Na skúmanie týchto systémov potrebujeme počítače, a to z dvoch dôvodov. Prvým je, že pomocou počítačov môžeme vo veľkom zbierať a spracovávať dáta, napríklad: získať dáta o počte populácií líšok a zajacov. Druhý dôvod je, že môžeme vytvárať rozsiahle modely týchto systémov.

Simulácie výpočtových modelov sú nielen dôležité, ale aj zaujímavé, a to predovšetkým svojimi vizuálnymi výstupmi. Získame nimi totiž grafy, ktoré nám znázornia správanie systému. Často však vidíme modelovanie systému aj na vlastné oči, ako napríklad pozorovanie hurikánu či dopravná zápcha na ceste. Vizuálne výstupy nie sú zložité a pochopia ich aj laici. Keď je simulácie dobre spracovaná a dáva zmysel, potom prináša pochopenie aj bez detailných znalostí. (Pelánek, 2011)

1.1 TYPY MODELOV

Máme veľké množstvo typov modelov. Niektoré z nich sú úzko späté s konkrétnym cieľom, ale väčšinu modelov môžeme použiť rôznymi spôsobmi.

- Mentálne modely – naše myslenie je založené na tomto modeli. Realitu nevnímame priamo, ale prostredníctvom našich zmyslov.
- Fyzické modely – tieto modely poznáme od detstva. Veľa hračiek je vlastne modelmi. Fyzické modely sa používajú napríklad aj v dopravných simulátoroch.
- Matematické modely – tento model tvoria matematické rovnice. Môžeme ich rozdeliť na dve podskupiny – popisné a dynamické modely. (Pelánek, 2011)

1.1.1 MODELOVANIE

Modelovanie môžeme rozdeliť do šiestich fáz: formulácia problému, návrh modelu, implementácia, verifikácia a validácia, simulácia a analýza, sumarizácia výsledkov.

V prvej fáze modelovania formulujeme, aký problém sa snažíme s pomocou modelu riešiť a na aké otázky sa pokúšame odpovedať. Snažíme sa problém zachytiť čo najkonkrétnejšie. Určíme si časový interval, v ktorom problém študujeme. Pri formulovaní problému berieme do úvahy prostriedky, ktoré máme k dispozícii.

V prvých fázach sa snažíme udržiavať model čo najjednoduchší. Používame najhrubšiu úroveň abstrakcie, ktorá je dostatočná pre daný účel a detaily pridávame, až keď je to nutné. Vyberieme si, ktoré prvky budeme modelovať. Zameriavame sa predovšetkým na to, čo bude v modeli zohľadnené, a ako detailne budú jednotlivé prvky opísané. Určujeme hlavné prvky modelu, časti a vzťahy medzi nimi. Používame kvalitatívne vzťahy (čo s čím súvisí), nie kvantitatívne (ako presne to súvisí).

Vyberieme si modelovací prístup a nástroj, v našom prípade to bude Matlab. Určíme si kľúčové prvky, doplníme ďalšie časti modelu a model uzavrieme. Použitie výpočtových modelov má tú výhodu, že nás núti doplniť všetky časti modelu a presne vyjadriť všetky vlastnosti a súvislosti. Počítačový model, aby bol spustiteľný, musí byť kompletný a musíme jasne formulovať všetky predpoklady. Posledným krokom implementácie modelu je určenie hodnôt parametrov. Tieto informácie určujeme na základe pozorovania, štatistických meraní alebo prostého odhadu.

Akonáhle máme model (program) spustiteľný, overíme, či je naozaj vhodný pre zodpovedanie vytýčených otázok - musíme model verifikovať a validovať. Validácia je

overenie, že návrh modelu naozaj reflektuje správanie reálneho systému, t. j. overenie vzťahu medzi realitou a abstraktným návrhom. Verifikácia je overenie, že model naozaj robí to, čo si myslíme, že by mal robiť, t. j. overenie vzťahu medzi abstraktným návrhom modelu a jeho konkrétnej implementácii.

Verifikácia sa dá vykonávať celkom presne pomocou presných postupov. S validáciou je to náročnejšie, pretože zo základného princípu "všetky modely sú zlé" vyplýva, že model nikdy nemôže byť úplne reálny. Validácia je teda vždy do istej miery subjektívna a skôr než správnosťou modelu sa zaoberá jeho užitočnosťou.

Keď sa nám podarí vytvoriť model, o ktorom si myslíme, že je vhodný, dôkladne ho analyzujeme. Ciele analýzy závisia na konkrétnom probléme, typicky však spadajú pod nasledujúce otázky. Akú úlohu hrajú jednotlivé prvky modelu? Ktoré prvky modelu majú najväčší vplyv na jeho správanie? Aké je správanie modelu za zmenených podmienok?

Študujeme výsledky, pokúšame sa o formuláciu odpovedí na pôvodné otázky. V prípade potreby sa vraciame k predchádzajúcim bodom. Snažíme sa zodpovedať otázky: Či splnil náš model účel, s ktorým bol vytvorený? Ako môžeme náš model ďalej využiť?

Skôr ako sa pustíme do tvorby jednotlivých modelov, pozrieme sa na základné prístupy k výpočtovým modelom. K modelovaniu môžeme pristupovať zhora alebo zdola. Ak modelujeme zhora, pozeráme sa na súhrnné veličiny, zaujíma nás celková štruktúra systému. Keď modelujeme zdola, tak sa pozeráme na jednotlivé časti, definujeme pravidlá ich správania a vývoj celku dostávame ako dôsledok.

Matematické modelovanie predstavuje modelovanie zhora. Stav systému vyjadrujeme pomocou premenných (počet zajacov) a správanie systému pomocou rovníc (napr. ako závisí počet zajacov na počte líšok). (Pelánek, 2011)

1.2 MATEMATICKÉ MODELOVANIE POMOCOU ROVNÍC

Modelovanie môžeme z pohľadu matematiky rozdeliť do dvoch kategórii. Stav systému a správanie systému. Stav vyjadrujeme pomocou premenných. Pod premennými myslíme súhrnné informácie, napríklad veľkosť populácie koristi (zajace). Správanie systému vyjadrujeme pomocou rovníc, v ktorých vystupujú stavové premenné.

Rovnice nám udávajú, ako sa mení hodnota premenných. Túto zmenu môžeme vykonávať v diskretných časových krokoch alebo spojito - podľa toho rozlišujeme základné typy matematických modelov. V oboch prípadoch môžeme správanie modelu definovaného rovnicami ďalej skúmať dvoma spôsobmi. Po prvé môžeme použiť exaktnú

matematickú analýzu, pomocou ktorej nájdeme všeobecné riešenie rovníc alebo ekvilibrium systému, t. j. hodnoty premenných, pre ktoré sa správanie modelu ustáli. Po druhé môžeme použiť simuláciu - určíme počiatočnú hodnotu premenných a ďalšie správanie modelu určíme numerickým výpočtom. (Pelánek, 2011)

1.3 DISKRÉTNY ČAS A REKURENTNÉ ROVNICE

Pri použití diskrétného času udávajú rovnice hodnotu premenných v časovom okamihu $t + 1$ v závislosti od hodnoty premenných v čase 0 až t . Takéto rovnice sa nazývajú rekurentné rovnice. Ak máme zadané počiatočné hodnoty premenných (v čase 0), hodnoty premenných môžeme vypočítať priamočiaro dosadením.

Najznámejší príklad je Fibonacciho postupnosť, ktorá popisuje idealizované množenie zajacov. Na začiatku máme jeden pár nesmrteľných zajacov. Akonáhle zajace dosiahnu vek dvoch rokov, začnú plodiť mladé (1 pár splodí za rok 1 pár mladých). Označíme veľkosť populácie v čase t ako X_t . Uvedené predpoklady môžeme vyjadriť pomocou nasledujúcej rekurentnej rovnice:

$$X_{t+1} = X_t + X_{t-1} \quad (1)$$

Na dokončenie modelu ešte musíme určiť počiatočné hodnoty: $X_0 = 1$, $X_1 = 1$. Pre tento model je možné nájsť explicitné riešenie, ktoré vyjadruje X_t priamo v závislosti na t :

$$X_t = \frac{\phi^{-(1-\phi^t)}}{\sqrt{5}} \quad (2)$$

Kde $\phi = (1+\sqrt{5})/2$

Detailné riešenie tu nebudeme rozoberať. Z pohľadu modelovania nám stačí získať hodnoty výpočtom pomocou dosadenia: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144 atď. Z uvedených analýz vyplýva, že v modeli sa zajace množia exponenciálne rýchlo a ich počet rastie nad všetky hranice. Takéto správanie samozrejme nie je realistické, žiadna populácia sa nemôže množiť nad všetky hranice. V modeli, ktorý zachytáva Fibonacciho postupnosť, máme iba pozitívnu spätnú väzbu (čím viac zajacov, tým viac mladých), ale chýba nám korigujúca negatívna spätná väzba (limit prostredie). Skonstruujme teda mierne pozmenený model, ktorý zahrnie aj kapacitu prostredia (takzvaná logistická rovnica):

$$X_{t+1} = X_t (1 - X_t / K) \quad (3)$$

Parameter r udáva základnú mieru reprodukcie populácie - čím väčšie r , tým rýchlejšie populácia rastie. Parameter K udáva kapacitu prostredia. Akonáhle sa veľkosť populácie

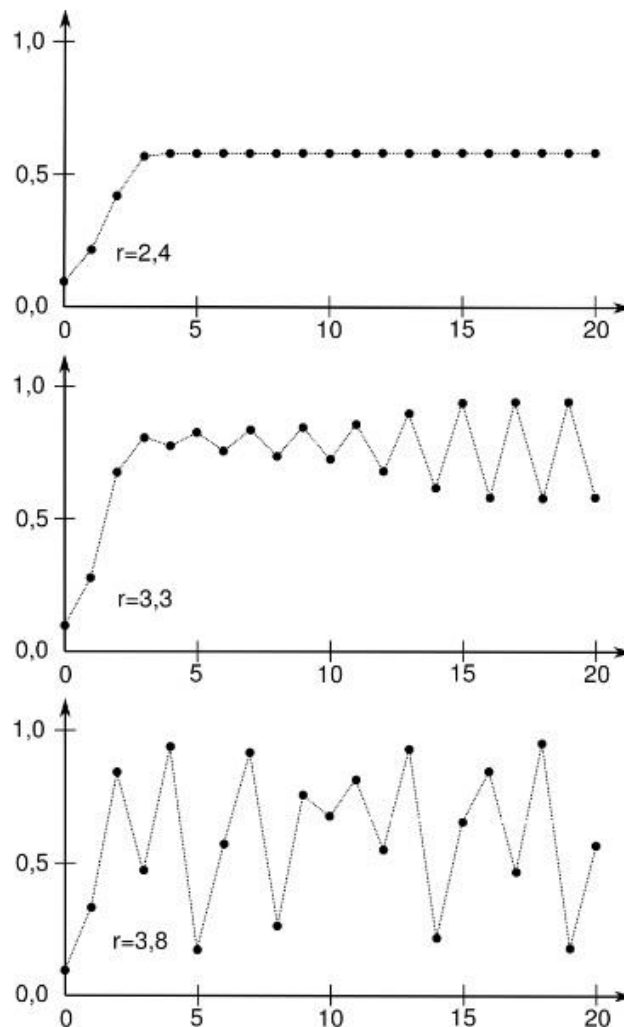
priblíži ku kapacite prostredie, člen $1 - X_t / K$ klesne a veľkosť populácie prestane rásť alebo sa začne aj znižovať.

Alternatívne môžeme rovnaký model odvodiť z predpokladu, že miera pôrodnosti je nezávislá na veľkosti populácie, keďže miera úmrtnosti závisí lineárne na veľkosti populácie (čím väčšia populácia, tým väčšia konkurencia a tým väčšia šanca, že jedinec zomrie). Tento predpoklad môžeme vyjadriť nasledovne (p je miera pôrodnosti, u_t je miera úmrtnosti):

$$X_{t+1} = pX_t - u_tX_t, u_t = uX_t \quad (4)$$

Čo je vlastne to isté ako v predchádzajúcej rovnici, keď $p=r$ a $u=r/K$.

Logistická rovnica ilustruje princíp, s ktorým sa u komplexných systémov stretávame často: jednoduché pravidlá môžu generovať zložité správanie. Ide síce o veľmi jednoduchú rovnicu, jej správanie závisí zložitým spôsobom na pomere parametrov r a K . Obrázok 1 ukazuje rôzne správanie tohto jednoduchého modelu v závislosti od hodnoty parametrov - vidíme, že správanie sa môže ustáliť, pravidelne oscilovať, ale aj skákať úplne chaoticky. (Pelánek, 2011)



Obrázok 1 – Logistický rast populácie, správanie modelu pre $X_0 = 0,1$; $K = 1$
a tri hodnoty r^1

1.4 SPOJITÝ ČAS A DIFERENCIÁLNE ROVNICE

V niektorých prípadoch nemožno čas jednoducho rozdeliť na diskkrétne kroky. V prípade modelovania populácií je rozdelenie na diskkrétne kroky problematické, pretože populačnú dynamiku možno málokedy rozdeliť na striktne oddelené generácie. Jedinci sa väčšinou rodia a umierajú priebežne. Preto používame spojitý čas a modelovanie pomocou diferenciálnych rovníc.

Ako príklad si uvedieme jednoduchý model pre rast populácie. Základnú verziu modelu môžeme slovne vyjadriť:

$$\text{zmena veľkosti populácie} = \text{počet narodení} - \text{počet úmrtí}.$$

¹ Zdroj Obrázok 1 : Modelování a simulace komplexních systémů

Pre začiatok predpokladáme, že počet narodení aj úmrtí je úmerný iba veľkosti populácie, a že miera pôrodnosti p aj miera úmrtnosti u sú nezávislé na veľkosti populácie. Dostávame diferenciálnu rovnicu:

$$\frac{dX}{dt} = pX - uX \quad (5)$$

Túto rovnicu môžeme upraviť pomocou miery reprodukcie $r = p - u$ na:

$$\frac{dX}{dt} = rX \quad (6)$$

Táto diferenciálna rovnica je jednoduchá a môžeme ju analyticky vyriešiť. Riešením je funkcia:

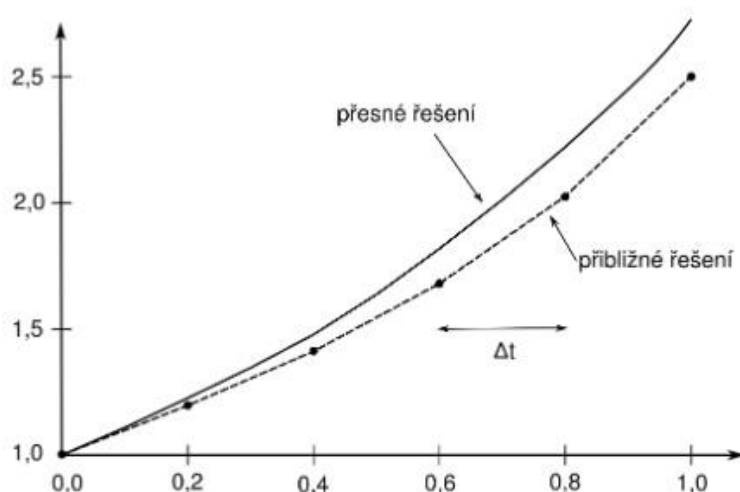
$$X(t) = X(0)e^{rt} \quad (7)$$

Veľkosť populácie rastie alebo klesá exponenciálne (podľa toho, či je r kladné, alebo záporné). Realistickejšie správanie modelu môžeme dosiahnuť rovnakým spôsobom ako pre diskretný model, t. j. pomocou miery úmrtnosti závislej na hustote, respektíve pomocou kapacity prostredia:

$$\frac{dX}{dt} = rX\left(1 - \frac{X}{K}\right) \quad (8)$$

Riešením tejto diferenciálnej rovnice je (Pelánek, 2011):

$$X(t) = K / \left(1 + \left(\frac{K}{X(0)} - 1\right)e^{-rt}\right) \quad (9)$$



Obrázok 2 – Presne a približné riešenie diferenciálnych rovníc $dx/dt = x^2$

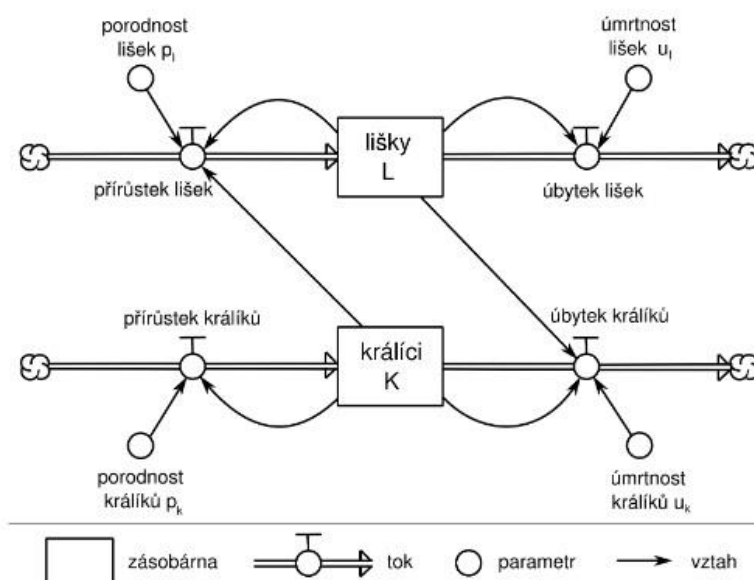
² Zdroj Obrázok 2 : Modelování a simulace komplexních systémů

1.5 SYSTÉMOVÁ DYNAMIKA

Pod pojmom systémová dynamika alebo aj systémové modelovanie si môžeme predstaviť grafickú nadstavbu nad matematickým modelovaním. Postup modelovania je nasledujúci:

- graficky vyjadríme základné vzťahy;
- na základe diagramu nástroj na modelovanie automaticky vygeneruje základné diferenciálne rovnice;
- zvyšné rovnice a hodnoty parametrov doplníme ručne;
- pomocou programu vykonávame simuláciu.

Príklad jednoduchého systémového modelu je na obrázku 3. Model zachytáva vzťahy medzi populáciami líšok a zajacov. Ide o veľmi zjednodušený model, v ktorom predpokladáme, že líšky žerú iba zajace a zajace umierajú iba vtedy, keď ich zožerie líška. Populácie líšok aj zajacov sú zachytené jednou premennou, ktorá udáva počet jedincov v populácii. V modeli je zahrnutý prírastok a pokles veľkosti populácií, pričom prírastok líšok (pokles zajacov) závisí od počtu stretnutí líšok a zajacov (t. j. na veľkosti oboch populácií). Pokles líšok (prírastok zajacov) závisí iba od veľkosti populácie líšok (zajacov). Obrázok 3 vyjadruje prítomnosť závislostí, nie ich charakter. Aby sme mohli model simulovať, musíme závislosti presne špecifikovať a určiť konkrétne hodnoty parametrov. (Pelánek, 2011)



Obrázok 3 – model v systémovej dynamike³

³ Zdroj Obrázok 3 : Modelování a simulace komplexních systémů

1.5.1 ZÁKLADNÉ PRVKY GRAFICKÉHO ZNÁZORNENIA

Prvky graficky vyjadreného modelu rozdeľujeme do niekoľkých tried, a každú z nich zakresľujeme špecifickým spôsobom. V tabuľke 1 máme uvedený príklad popísaných pojmov.

Tabuľka 1: Základné prvky ⁴

Zásobáreň	Tok	Parameter
<i>Populácia</i>	<i>Narodenie, úmrtie, migrácia</i>	<i>Pôrodnosť, úmrtnosť, miera migrácie</i>

Zásobárne, ktoré značíme obdĺžnikom, sú „podstatné mená“ v modeli. Sú to časti systému, kde sa niečo akumuluje. Obsah zásobárne možno číselne vyjadriť a v čase môže stúpať a klesať. Ak si predstavíme systém zmrazený v určitom okamihu, má zásobáreň (takmer vždy) nenulovú hodnotu.

Toky sú „slovesá“ v modeli. Sú to aktivity, ktoré určujú hodnotu rezervoárov v čase, t. j. určujú, či obsah zásobárne narastá, alebo klesá. Ak si predstavíme systém zmrazený v určitom okamihu, tak toky majú nulovú hodnotu. Toky môžu byť jednosmerné aj obojsmerné a značíme ich dvojitou šípkou, ktorá znázorňuje, odkiaľ a kam tok smeruje. Na každom toku je „ventil“, ktorý v závislosti na parametroch a stave rezervoárov určuje veľkosť toku. Obláčik využívame, ak zdroj alebo cieľ toku ležiaci mimo okraj modelu (napr. v uvedenom modeli sa nezaobráme tým, čo sa stane s líškou po tom, čo zomrie).

Parametre určujú tempo, s akým dochádza k zmene obsahu zásobárne vplyvom tokov. Sú to často vonkajšie premenné systému, t. j. prvky, ktorých správanie priamo modelujú a iba na základe pozorovania či úvahy určíme ich hodnotu. Parametre značíme kolieskom. Všetky uvedené prvky prepájame šípkami, ktoré udávajú vzťahy - čo s čím súvisí. (Pelánek, 2011)

1.5.2 VYJADRENIE POMOCOU ROVNÍC

Ako sme už uviedli, systémové modely sú v podstate systémy diferenciálnych rovníc. Takéto rozdelenie, do rôznych stavebných prvkov, slúži "len" k lepšiemu modelovaniu a ku grafickému znázorneniu modelov. Na príklade líšok a zajacov ukážeme, ako prejdeme od grafického stvárnenia k rovniciam.

Aby sme mohli model dokončiť, musíme ku grafickému vyjadreniu doplniť:

⁴ Zdroj Tabuľka 1 : Modelování a simulace komplexních systémů

- počiatočné hodnoty zásobárne, čiže hodnoty premenných K a L ,
- hodnoty parametrov - r_l (rast líšok), r_k (rast zajacov), p_l (pokles líšok), p_k (pokles zajacov)
- rovnice pre veľkosť tokov

Pri zadávaní rovníc dodržiavame konvenciu, že tok závisí práve na tých prvkoch, z ktorých do neho vedie šípka. Vzťah sa snažíme vyjadriť čo najjednoduchším matematickým vzorcom (typicky násobením ako vo vyššie uvedenom prípade). Ak nevieme vyjadriť závislosť pomocou jednoduchého matematického vzťahu, môžeme použiť závislosť vyjadrenú grafom.

Teraz nám chýba už len rovnice pre výpočet hodnoty zásobární (t. j. premenných K a L). Tieto rovnice môžeme vygenerovať automaticky, pretože všeobecne platí:

$$\text{zmena hodnoty zásobárne} = \text{vstupné toky} - \text{výstupné toky}$$

Dostaneme model:

$$\frac{dL}{dt} = r_1 KL - p_1 L \quad (10)$$

$$\frac{dK}{dt} = r_1 K - p_1 KL \quad (11)$$

1.6 ZÁKLADNÉ VZORY SPRÁVANIA

Dobry výsledok väčšinou nevznikne vďaka úžasným novým základným prvkom, ale hlavne vďaka dobrej kombinácii osvedčených dielov. Tento princíp platí aj pri modelovaní. Dobré modely nevznikajú vďaka zložitým kombináciám vzťahov medzi premennými, ale vďaka kombinácii základných vzorov správania.

Existuje päť základných vzorov správania, ktoré sa snažíme v modeloch pravidelne zachytiť. Tieto vzory správania sú zachytené na obrázku 4, na ktorom sú tiež uvedené diagramy v systémovej dynamike. Teraz si jednotlivé vzory predstavíme.

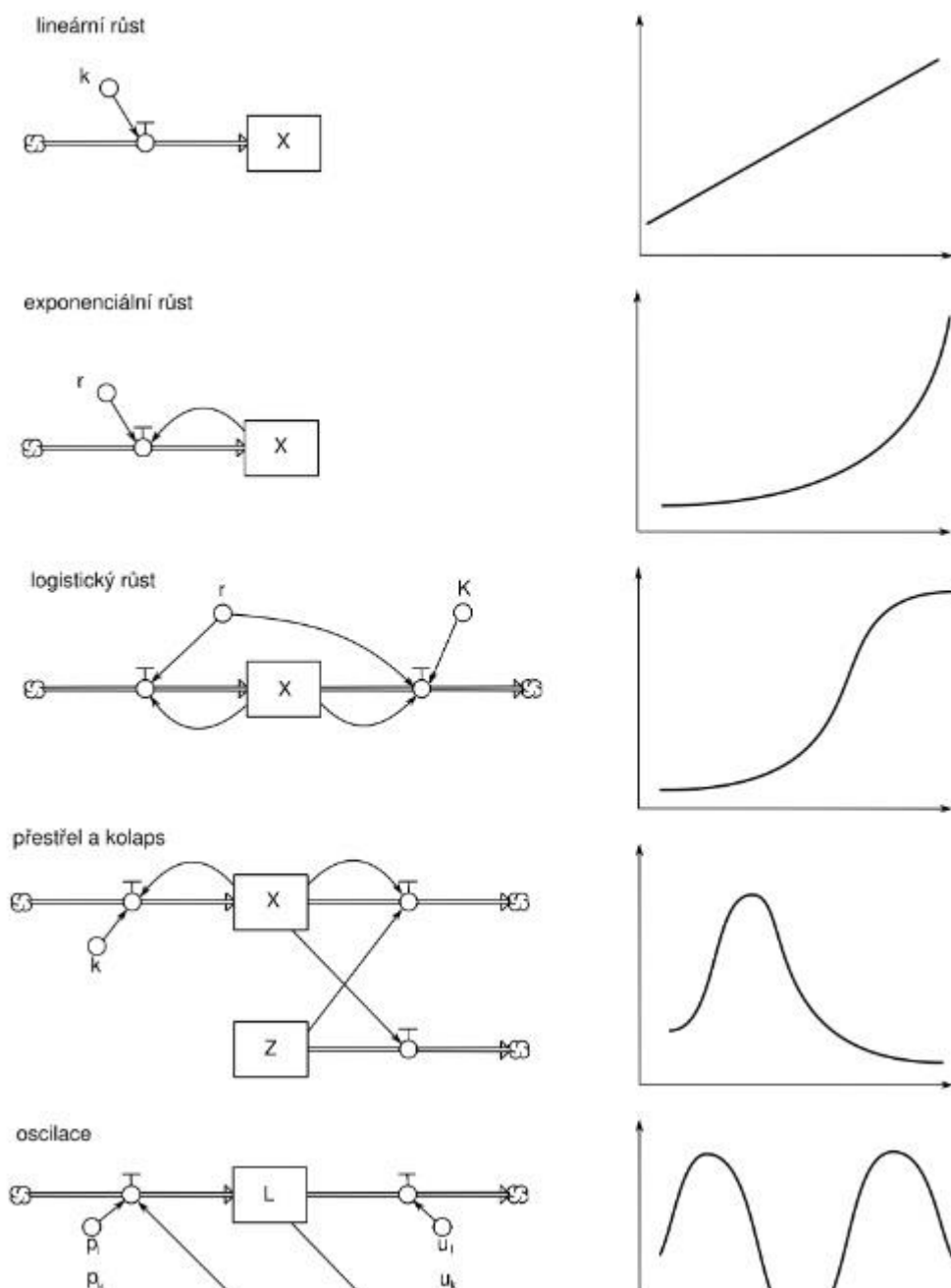
Pri lineárnom raste dochádza konštantnou rýchlosťou k zmene, nie je tu žiadna spätná väzba.

Pri exponenciálnom raste je rýchlosť zmien úmerná veľkosti zásobárne. Systém je teda riadený pozitívnou spätnou väzbou.

Logistický vývoj - v prvej fáze dochádza k exponenciálnemu rastu, ktorý je nasledovaný približovaním k rovnováhe (kapacite K). Ide o kombináciu pozitívnej a negatívnej spätnej väzby.

Prestrel a kolaps - máme dve zásobárne, jedna z nich je neobnoviteľné, druhá od nej závisí a spotrebúva ju. Ide o kombináciu pozitívnej a negatívnej spätnej väzby.

Pri oscilácii ide o dve vzájomne závislé zásobárne, ktoré sú spojené negatívne spätnou väzbou s oneskorením. Tento typ závislosti môžeme vidieť na príklade líšky a zajaca (dravec-korist'). (Pelánek, 2011)



Obrázok 4 – vzory správania⁵

⁵ Zdroj Obrázok 4 : Modelování a simulace komplexních systémů

2 CIELE DIPLOMOVEJ PRÁCE

Cieľom diplomovej práce je vo zvolenom systéme simulovať rôzne druhy matematických modelov typu dravec-korist'. Okrem toho sme si stanovili ešte menšie podciele.

Podciele:

- opísať program na vytvorenie matematických modelov,
- vytvoriť matematické modely typu dravec-korist',
- vo vybranom programe simulovať matematické modely.

3 MATLAB

V tejto kapitole predstavíme program Matlab. Program Matlab sa používa na riešenie širokého spektra problémov a jeho úplný opis je veľmi rozsiahly. Opíšeme teda hlavne tie časti Matlabu, ktoré budeme potrebovať na tvorbu našich programov.

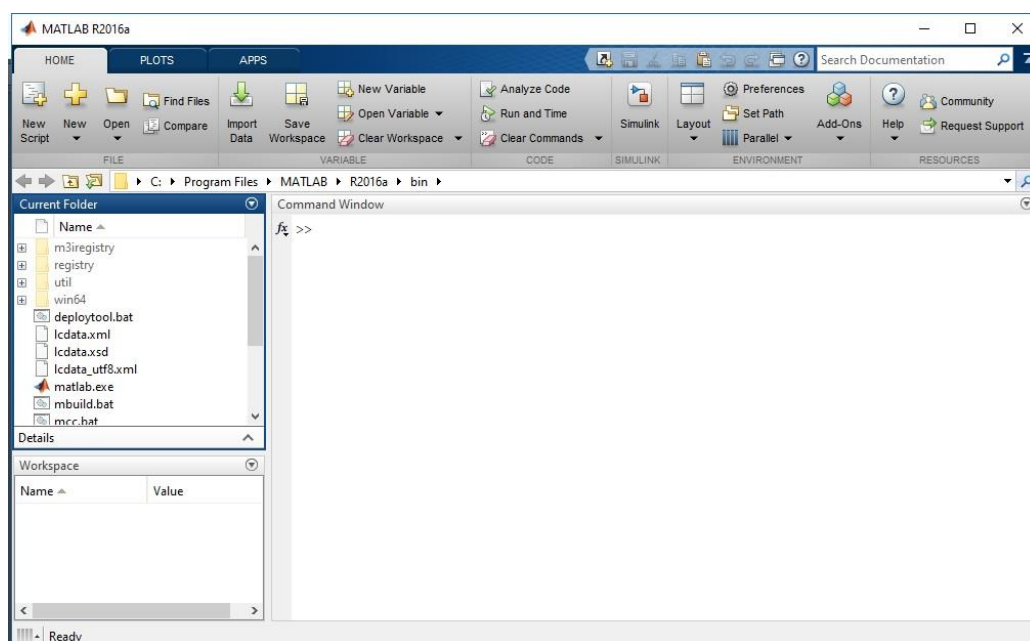
S vývojom Matlabu začal Cleve Moler v neskorých sedemdesiatych rokoch (Moler, 2004). V roku 1984 Moler spoločne so Stevom Bangertom a Jackom Littlom založili MathWorks a pokračovali vo vývoji Matlabu. Názov vznikol skrátením anglického MATrix LABoratory. Matlab je program, ktorý má vlastný programovací jazyk. Umožňuje manipuláciu s maticami, vykresľovanie funkcií a dát, implementáciu algoritmov, vytváranie používateľských rozhraní a kompatibilitu s programami napísanými v iných jazykoch, ako sú napríklad C, C++, Java a Python. Matlab bol vytvorený pre všetky významné platformy ako Windows, Linux, Solaris a Mac. Je to ideálny program pre inžinierov, vedcov, matematikov a učiteľov pri riešení problémov z mnohých oblastí. Je to ideálny inžiniersky nástroj, ktorý má interaktívne prostredie pre vedecké a technické výpočty, analýzu dát, vizualizáciu a vývoj algoritmov (Humusoft, 2018).

Kľúčové vlastnosti Matlabu sú (Karban, 2006):

- vysokoúrovňový jazyk pre technické výpočty,
- otvorený a rozšíriteľný systém,
- veľké množstvo knižníc,
- podpora viacrozmerných polí a dátových štruktúr,
- interaktívne nástroje pre tvorbu grafického užívateľského rozhrania,
- importovanie a exportovanie dát do množstva formátov,
- komunikácia s externými meracími a monitorovacími prístrojmi v reálnom čase,
- rozšíriteľnosť modulov jazykov C, C++, Java a Fortran.

Matlab má štruktúrované dátové typy. Pretože všetky premenné v Matlabe sú polia, kde každý prvok polia má rovnaké názvy polí. Okrem toho Matlab podporuje názvy dynamických polí. Matlab môžeme rozdeliť na nasledujúce časti - výpočtové jadro (najdôležitejšia časť Matlabu), grafický subsystém, pracovné nástroje a toolbox.

Po spustení Matlabe sa nám objaví pracovná plocha, ktorá pozostáva z niekoľkých okien. Current Directory nám zobrazuje aktuálny adresár. Je umiestnený v hornej lište. Môžeme si nastaviť aj iný pracovný adresár alebo otvoriť viac súborov, ktoré potom môžeme premenovať, kopírovať, presúvať popřípade zmazať. Najdôležitejšou a najpoužívanejšou časťou je Command Windows - príkazové okno vpravo. Toto okno slúži na komunikáciu medzi MATLAB a výpočtovým jadrom. Tu sa vykonáva väčšina práce, ako je napríklad zadávanie príkazov užívateľom, výpis vstupu funkcií, výpis chybových hlásení, výpis obsahu premenných alebo varovanie. Vľavo dole je okno Command History, v ktorom je zoznam použitých príkazov od posledného vyčistenia okna. Dvojklikom na vybraný príkaz vieme opätovne spustiť predtým zadaný príkaz, ktorý vieme potom upraviť pretiahnutím do Command Window. Tretie okno vľavo hore je Workspace - pracovná plocha pre sledovanie použitých premenných. Ako väčšina programov aj Matlab obsahuje nápoved' Matlab Help. Jedná sa o veľmi rozsiahlu nápoved' v anglickom jazyku, ktorá mimo všetkých dôležitých kapitol obsahuje aj veľké množstvo príkladov, ako aj návod pre začiatočníkov. (Karban, 2006)



Obrázok 5 - Matlab

Príkazový riadok používa nasledujúcu syntax:

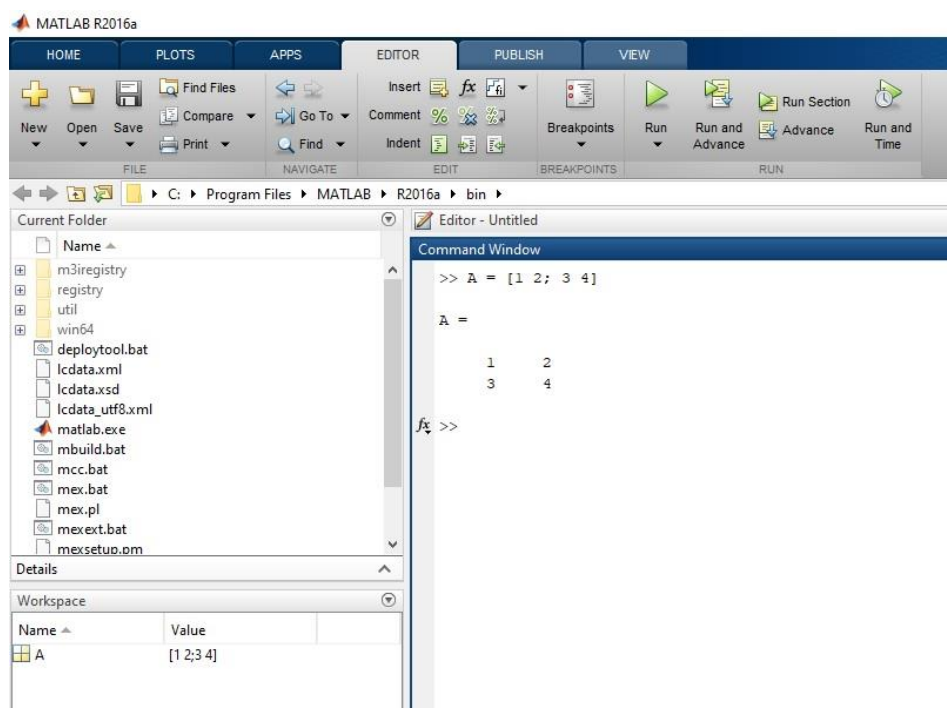
- Keď napíšeme príkaz a potvrdíme klávesom *Enter*, hneď sa vykoná.
- Na presúvanie medzi riadkami používame klávesy - (*Ctrl+p*) predchádzajúci riadok a (*Ctrl+n*) nasledujúci riadok.

- Ak chceme na začiatok slova presunúť kurzor použijeme klávesu *Enter* + \rightarrow .
- Na začiatok predchádzajúceho slova presunieme kurzor pomocou *Enter* + \leftarrow .
- Klávesom *ESC* zmažeme riadok.
- Keď máme príkaz veľmi dlhý a potrebujeme ho rozdeliť na viac riadkov, spravíme to tak, že na konci pridáme tri bodky (...).
- *Clc* slúži na vyčistenie okna v príkazovom riadku.
- Pomocníka zavoláme príkazom *Help*.
- *Who* a *whos* nám slúži na zadefinovanie premenných.
- *Clear* slúži na vyčistenie všetkých premenných, ktoré sme si zadefinovali.

S premennými pracujeme v okne *Workspace*, Zobrazuje nám všetky premenné a umožňuje nám prácu s nimi:

- dvojklikom zobrazí hodnotu premennej,
- načíta premenné zo súboru,
- edituje premenné,
- ukladá všetky premenné do súboru,
- pomocou *Delete* nám zmazáva premenné.

V Matlabe máme tri režimy práce. Prvý režim je dialógový, pod ktorým rozumieme zadávanie príkazov prostredníctvom príkazového riadku (*Command Window*). Tento režim môžeme použiť ako inteligentnú kalkulačku s viacerými funkciami. Napísané výrazy sa po stlačení klávesy *Enter* okamžite vykonajú. Druhý z režimov je programový a slúži na editovanie. Tretí režim je grafický. Jeho úlohou je vizualizácia výsledkov. Otvára samostatné okná, s ktorými môžeme ďalej pracovať v prostredí. Do tohto prostredia sa dostaneme, keď stlačíme *Dock Figure*. (Mathworks, 2018)



Obrázok 6 – Príkaz v programe Matlab

M-file sú súbory s príponou *.m a obsahujú postupnosť príkazov v Matlabe. Ukladáme ich na pevný disk alebo na iné záznamové médium. Vytvoriť m-súbor je možné v akomkoľvek textovom editore, ktorý nepridáva žiadne informácie typu formátovanie, a ktorý umožňuje zápis vlastnej prípony súboru *.m. Pre používateľov býva najjednoduchšie cestou použiť na vytváranie m-súborov vstavaný editor v Matlab. Vytváraním m-súborov sa práca s Matlab blíži programovaniu, tak ako ho poznáme z iných vývojových prostredí. (Zaplatílek a kol. 2014).

Rozlišujeme dva typy m-súborov v Matlabe – skripty a funkcie. Skripty označujú m-súbory, ktoré obsahujú príkazy Matlabu. Keď spustíme skript, Matlab spustí príkazy, ktoré našiel v súbore. Skripty sú veľmi užitočné na vykonanie rôznych analýz, riešenie problémov alebo konštruovanie dlhých príkazov, ktoré sú ťažkopádne a zdĺhavé. Skripty nevedia to, čo vedia funkcie. Funkcia tiež patrí medzi m-súbory. Na rozdiel od skriptov môžu byť volané s jedným alebo viacerými vstupnými parametrami a môžu odovzdávať jeden alebo viac výstupných parametrov. To je dôležité napríklad vtedy, keď v rámci jedného m-súboru voláte iný m-súbor a jeho činnosť sa prejaví tým, že odovzdá svoje parametre. Funkcia je teda sofistikovanejší skript, ktorý toho vie viac, a s ktorého pomocou môžeme vytvárať nové funkcie k funkciám existujúcim (Zaplatílek a kol. 2014).

```

1 function RM = enrichment(t,x,r,h,e,m,k,lambda)
2     RM = zeros(2,1);
3     RM(1) = r*x(1)*(1-x(1)/k)-lambda*x(1)*x(2)/1+h*lambda*x(1);
4     RM(2) = e*lambda*x(1)*x(2)/1+h*lambda*x(1)-m*x(2);
5 end

```

Obrázok 7 - Príklad funkcie

Funkcia musí obsahovať hlavičku: **function** [vystupná_premenná] = názov_funkcie (vstupná_promenná), ďalej nasleduje postupnosť príkazov. Môžeme vkladať aj komentáre, ale musia byť oddelené znakom %. Funkcie sa uloží ako súbor názov_funkcie.m.

V Matlabe ide všetok grafický výstup do grafického okna. Týchto grafických okien môže byť viac ako jedno. Okná možno vytvárať a tiež sa medzi nimi prepínať pomocou príkazu *figure*. Samotný príkaz *figure* vytvorí prázdny graf.

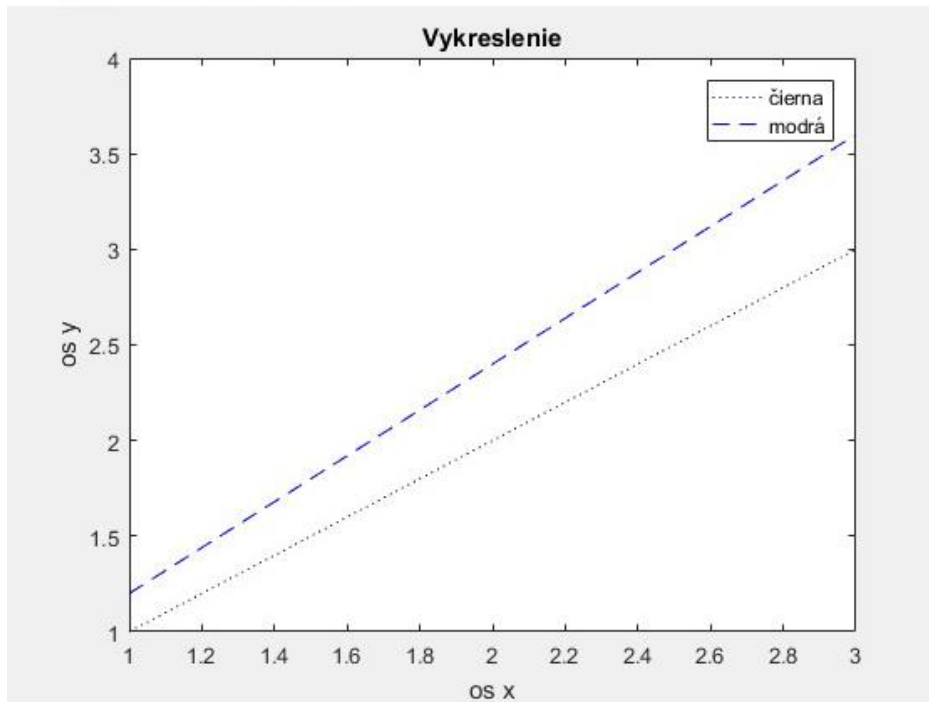
Na vykresľovanie funkcií budeme používať funkciu *plot*. Táto funkcia sa najčastejšie používa na vykresľovanie vektorov a matic do dvojrozmerného grafu. Syntax vyzerá nasledovne: *plot(a,b)*, vektory *a*, *b* musia byť rovnakej dĺžky. V tejto funkcii vieme nastaviť, akým spôsobom sa budú vektory vykresľovať. Ak chceme mať viac okien v jednom grafe, použijeme príkaz *subplot*. *Hold on/off* nám zase do jedného grafu dovoľí vykresliť viac čiar, ak nezadáme príkaz *hold on* a zavoláme viackrát *plot*, zostane nám len posledný graf. Pre vykreslenie 3D grafu k príkazu *plot* pridáme číslo tri – *plot3(x,y,z)*.

Grafy vieme ďalej upravovať pomocou funkcií v ponuke grafického okna alebo pomocou príkazov ako: *title* - nadpis grafu, *xlabel* a *ylabel* – popis horizontálnych a vertikálnych osí. *Legend* používame na vytvorenie legendy v grafe, najlepšie, keď máme v grafe viac čiar (Krejčí, a kol. 2014).

```

x=[1,2,3];
y=[1,2,3];
plot(x,y,'k:')           %vykreslenie čiernej bodkovanej čiary
hold on                  %vykreslenie viac čiar do jedneho grafu
plot(x,1.2*y,'b--')      %vykreslenie modrej čiarkovanej čiary
title('Vykreslenie')
xlabel('os x')
ylabel('os y')
legend('čierna','modrá')

```



Obrázok 8 - vykreslenia grafu

Čo sa týka ode riešiteľov, Matlab má rozsiahlu knižnicu funkcií pre riešenie obyčajných diferenciálnych rovníc. Máme na výber hneď z niekoľko možností. Ode45 je štandardný riešiteľ diferenciálnych rovníc v Matlabe. Platí všeobecne, že by mal byť prvou voľbou, ktorú by sme mali vyskúšať pri väčšine problémov. Ode23 a ode113 môžu byť efektívnejšie ako ode45, ak je problém náročný alebo vyžaduje vysokú presnosť.

V našej práci používame najčastejšie ode45. Táto funkcia implementuje metódu Runge-Kutta s premenlivým časovým krokom pre efektívny výpočet. ode45 je navrhnutý tak, aby zvládol nasledujúci všeobecný problém:

$$\frac{dx}{dt} = f(t, x), \quad x(t_0) = x_0 \quad (12)$$

kde t je nezávislá premenná, x je vektor závislých premenných a $f(t, x)$ je funkciou t a x . Matematický problém je špecifikovaný, keď je nastavený vektor funkcií na pravej strane rovnice $f(t, x)$ a sú zadané počiatočné podmienky, $x = x_0$ v čase t_0 . (Mathworks, 2018)

4 FUNKČNÉ ODPOVEDE HOLLINGOVHO TYPU

Existuje veľa príkladov v prírode, ktoré dokazujú, že dravce môžu kontrolovať počet svojej koristi. Ako príklad uvádzame ten z oblasti Yellowstone. Skôr, ako ľudia začali zabíjať dravcov (vlkov) v tejto oblasti, vlci boli práve tými, ktorý udržiavali počet losov. Vzťah popisujúci mieru konzumácie jedného dravca vzhľadom k hustote koristi bol nazvaný funkčnou odpoveďou Hollingovho typu. Tieto funkčné odpovede sú pomenované po kanadskom ekológovi a profesorovi Crawfordovi Stanleyem Hollingovi, ktorý tieto funkčné odpovede zaviedol v roku 1959. Existujú dva faktory, ktoré určujú, že funkčné odpovede dosahujú určitú hornú hranicu z prirodzených dôvodov. Po prvé, dravce sa môžu uspokojiť (ich žalúdok je dostatočne naplnený), preto ich miera lovu závisí na tom, ako sa im rýchlo podarí spracovať potravu. Druhým dôležitým faktorom je ten, že dravec neloví nepretržite. Kým začne vyhľadávať novú korisť, musí najskôr skonsumovať svoj predošlý úlovok. Čas, ktorý je v rozmedzí chytenia a skonsumovania koristi je stály, teda nemenný. Holling pomocou rovnice opísal tento vzťah medzi časom potrebným na vyhľadávanie, dobou manipulácie a časom potrebným na zjedenie koristi. Na tvar rovnice prišiel Holling vďaka experimentu. V rámci tohto experimentu mali ľudia zaviazané oči a ich úlohou bolo nájsť a zdvihnúť z rovného povrchu kolieska zo štvorcentimetrového brusného papiera. Ľudia v tomto experimente predstavovali dravcov a kolieska, ktoré museli zdvihnúť, predstavovali korisť. Pri tomto experimente Holling zaviedol nasledujúce veličiny:

N = hustota koristi (density of prey), a = efektivita hľadania (searching efficiency), T_t = celkový čas (total time), T_s = čas hľadania koristi (searching time), T_h = čas potrebný na chytenie koristi (handling time), P_e = počet ulovenej koristi počas hľadania (prey eaten)

Holling sa týmto experimentom presvedčil, že P_e rastie so zvyšujúcim sa celkovým časom hľadania koristi (T_s), s väčšou hustotou populácie koristi (N) a s vyššou efektivitou hľadania (a):

$$P_e = aT_sN \quad (13)$$

Pri väčšej populácii N klesá čas potrebný na vyhľadanie koristi (T_s). Preto T_s nie je konštantou a platí:

$$T_s = T_t - T_h P_e \quad (14)$$

Nahradením prvej rovnice dostaneme:

$$P_e = a(T_t - T_h P_e)N \quad (15)$$

Po upravení dostaneme Hollingovu rovnicu:

$$P_e = \frac{aNT_t}{1+aT_hN} \quad (16)$$

Táto rovnica je známa ako funkčná odpoveď Hollingovho II typu. Rovnica opisuje množstvo spotrebované počas určeného časového obdobia - Tt . Predpokladá sa, že hustota koristi zostáva počas tohto obdobia konštantná. V experimentoch to môže byť zaručené nahradením akejkoľvek koristi, ktorá je skonsumovaná. Holling sa neuspokojil len s experimentom na ľuďoch, ale tiež skúmal v laboratóriu tri druhy malých cicavcov. (Haefner, 2001)

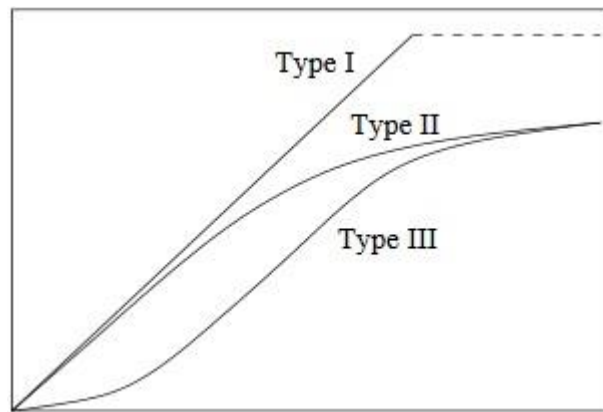
Poznáme tri typy funkčných odpovedí. Na obrázku 9 vidíme znázornené tri všeobecné typy kriviek, ktoré sa očakávajú v rôznych situáciách modelov dravec-korist'.

Typ I – $Pe=aNTt$, takúto rovnicu dostaneme, ak je veľmi krátky čas lovu ($Th \rightarrow 0$). Dravce sú v takomto prípade schopné zachovať priamy pomer lovu aj pri rastúcej hustote koristi. S touto funkčnou odpoveďou sa môžeme stretnúť v modeli Lotka-Volterra. (Baigent, 2018)

Typ II – so zvyšujúcou sa hustotou narastá aj miera lovu, nerastie do nekonečna, ustáli sa časom na hodnote, ktorú už dravec nie je schopný alebo ochotný prekročiť (je dostatočne nasýtený). S týmto typom dravca a koristi sa môžeme stretnúť v modeli Rosenzweig-MacArthur. Funkčná odpoveď typu II. je tvaru $P_e = \frac{aNT_t}{1+aT_hN}$

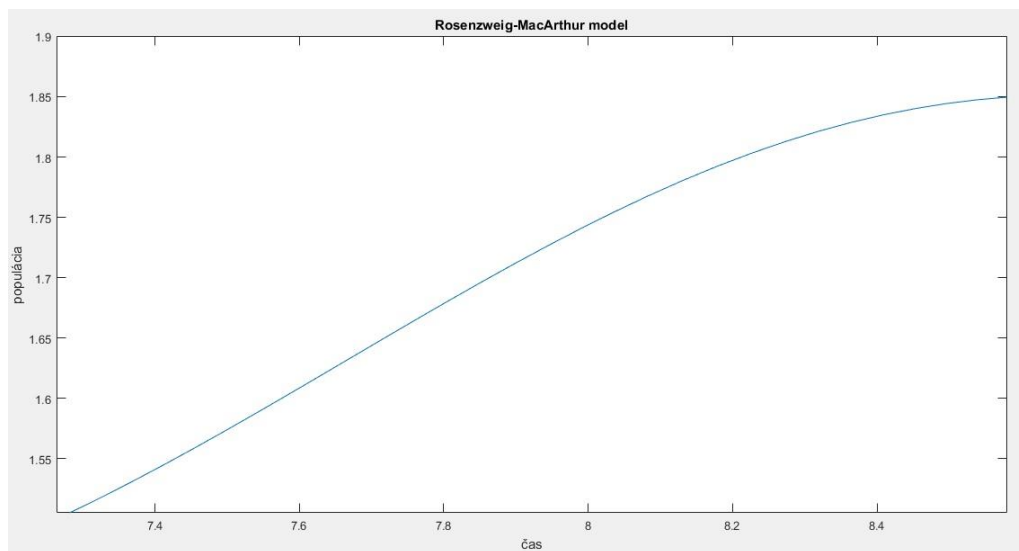
Typ III – táto funkčná odpoveď je zhora ohraničená funkčnou odpoveďou druhého typu. S malou hustotou koristi má dravec nízku priemernú funkčnú odpoveď, naopak s väčšou hustotou sa s množstvom ulovenej koristi učí lepšie loviť, a preto ulovená korisť narastá konvexne, až kým nepreváži nasýtenie. Parameter k umožňuje voliť tvar tejto funkčnej odpovede, ktorá je zovšeobecnením druhého typu. Najčastejšie sa ním modelujú dravce, ktoré majú schopnosť učiť sa. (Dawes, a kol. 2013)

$$P_e = \frac{aTN^k}{1+aT_hN^k}, k > 1 \quad (17)$$



Obrázok 9 - Funkčné odpovede Hollingovho typu⁶

Na obrázku 10 vidíme príklad funkčnej odpovede Hollingovho II typu, s ktorým sa stretneme v modeli Rosenzweig-MacArthur.



Obrázok 10 - Funkčné odpovede Hollingovho II typu v modeli Rosenzweig-MacArthur

⁶ Zdroj Obrázok 9- <http://www.indiana.edu/~oso/lessons/BioMath/hd-root.pdf>

5 MODELÝ RASTU POPULÁCIE

Najjednoduchší model rastu alebo rozpadu populácie hovorí, že rýchlosť rastu alebo miera rozpadu je úmerná veľkosti samotnej populácie. Zvýšenie alebo zníženie veľkosti populácie vedie k pomernému nárastu alebo zníženiu počtu pôrodov a úmrtí. Matematicky je to opísané pomocou diferenciálnej rovnice:

$$\dot{y} = ky \quad (18)$$

Konštanta proporcionality k zodpovedá veľkosti populácie, $y(t)$, jej rýchlosti rastu $\dot{y}(t)$. Ak je k kladné, populácia sa zvyšuje; ak k je záporné, populácia klesá.

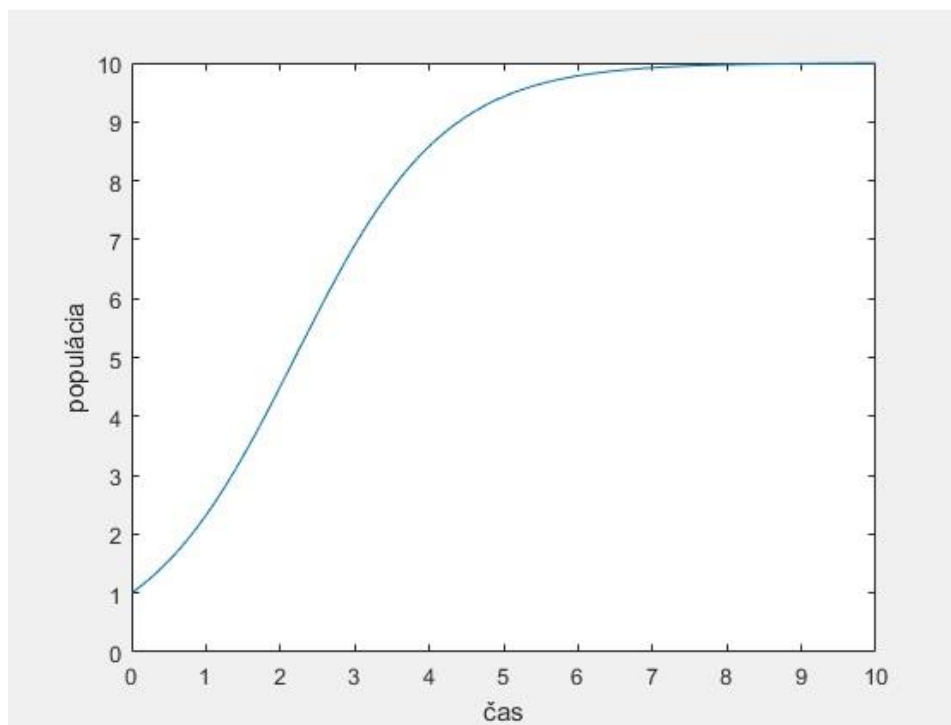
Riešením tejto rovnice je funkcia $y(t)$, ktorá je úmerná exponenciálnej funkcii:

$$y(t) = ne^{kt} \quad (19)$$

kde $\eta = y(0)$.

Tento jednoduchý model je vhodný v počiatočných štádiách rastu, keď neexistujú žiadne obmedzenia. V realistickejších situáciách existujú obmedzenia rastu, ako je napríklad konečný priestor. Viac realistický model hovorí, že obyvateľstvo si konkuruje. Keď sa populácia zvyšuje, jej miera rastu klesá lineárne. Diferenciálna rovnica sa niekedy nazýva logistická rovnica.

$$\dot{y} = k \left(1 - \frac{y}{\mu}\right) y \quad (20)$$

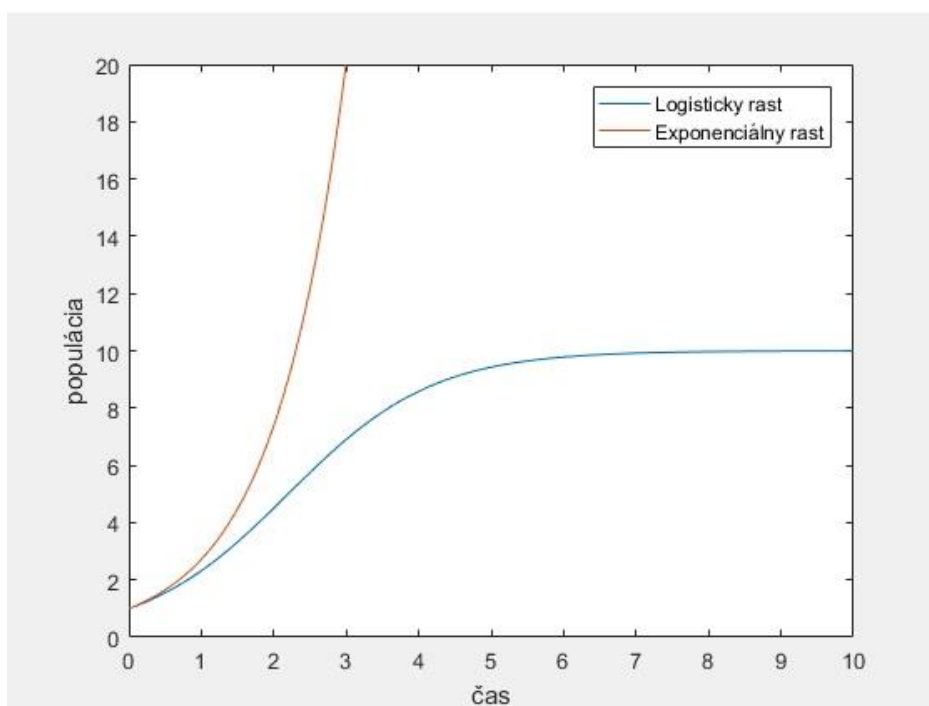


Obrázok 11 Logistická krivka

Nový parameter μ je nosnosť. Ako sa $y(t)$ blíži μ , rýchlosť rastu dosiahne nulu a rast sa nakoniec zastaví. Ukázalo sa, že riešenie je:

$$y(t) = \frac{\mu\eta e^{kt}}{\eta e^{kt} + \mu - \eta} \quad (21)$$

Môžete si ľahko overiť, že keď sa t priblíži k nule, $y(t)$ sa blíži η , a keď sa t blíži k nekonečno tak $y(t)$ sa blíži μ . Ak poznáte počet, potom môžete overiť, že $y(t)$ skutočne spĺňa logistickú rovnicu.



Obrázok 12 Exponenciálny a logistický rast

Obrázok 12 znázorňuje dve riešenia, ak obe η aj k sa rovnajú jednej. Exponenciálna funkcia:

$$y(t) = e^t \quad (22)$$

dáva rýchlo rastúcu červenú krivku. S kapacitou $\mu = 10$, logistická funkcia:

$$y(t) = \frac{10e^t}{e^t + 9} \quad (23)$$

dáva pomalšie rastúcu modrú krivku. Obe krivky majú rovnakú počiatočnú hodnotu a počiatočný sklon. Exponenciálna funkcia rastie exponenciálne. Logická funkcia taktiež rastie, ale nikdy neprekračuje hranicu kapacity. Ak nemáme vhodný vzorec na riešenie logistickej rovnice, môžeme počítať s ode45, jedným zo štandardných riešiteľov diferenciálnych rovníc v Matlabe. (Moler, 2011) Je to všestranný ODE riešiteľ a je prvým riešiteľom, ktorý by sme mali vyskúšať pri väčšine problémov. Ak je však problém náročný alebo vyžaduje vysokú presnosť, existujú aj iné riešenia ODE, ktoré by mohli byť vhodnejšie pre daný problém. (Mathworks, 2018)

Logistická rovnica a jej riešenie sa vyskytujú v mnohých rôznych oblastiach. Logistická funkcia je tiež známa ako sigmoidná funkcia a jej graf je známy ako S-krivka.

Model Lotka-Volterra predátor-korist' je najjednoduchší popis súboja medzi dvoma druhmi. Napríklad líška a zajac, lev a zebra, bocian a žaba.

Model dravec-korist' je dvojica diferenciálnych rovníc zahŕňajúcich pár konkurenčných populácií, $y_1(t)$ a $y_2(t)$. Rýchlosť rastu pre y_1 je lineárna funkcia a y_2 naopak.

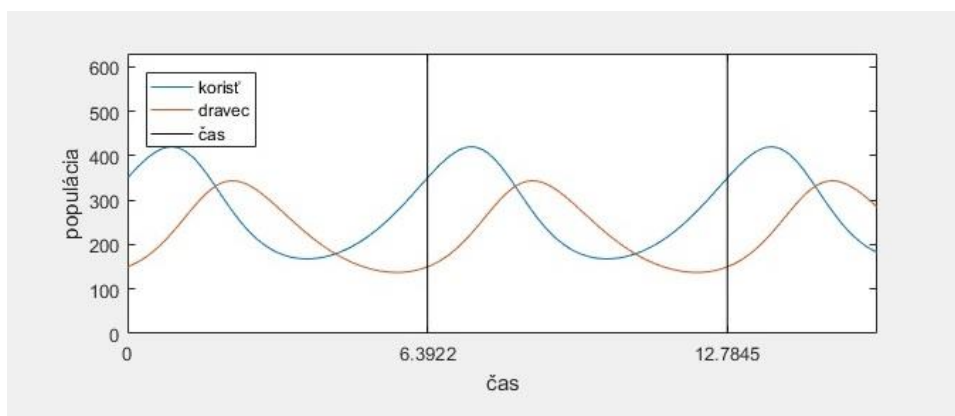
$$\dot{y}_1 = \left(1 - \frac{y_2}{\mu_2}\right) y_1 \quad (24)$$

$$\dot{y}_2 = -\left(1 - \frac{y_1}{\mu_1}\right) y_2 \quad (25)$$

Používame notáciu $y_1(t)$ a $y_2(t)$ namiesto, napríklad: $r(t)$ pre zajace a $f(t)$ pre líšky. Znamienko mínus v druhej rovnici rozlišuje dravcov od koristi. Ak sa y_1 stane nula, potom $\dot{y}_1 = y_1$ a dravci majú problémy. Ak sa však y_2 stáva nula, potom $\dot{y}_2 = -y_2$ a populácia koristi rastie exponenciálne.

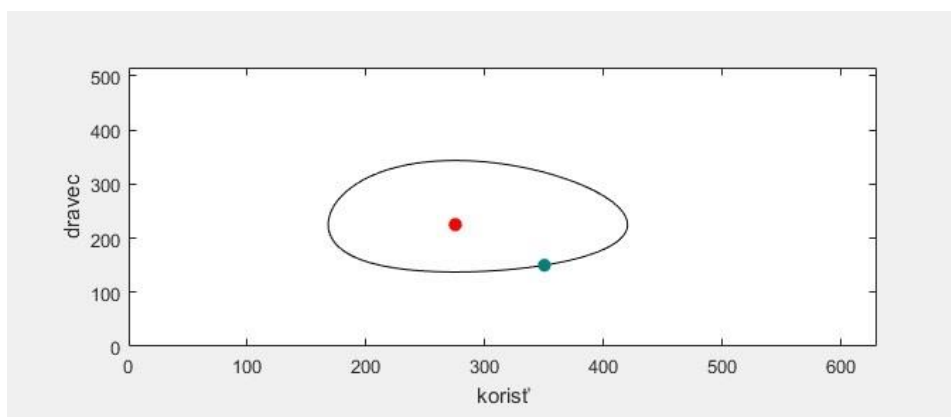
Máme vzorec pre riešenie logistického modelu s jedným druhom. Nie je však možné vyjadriť riešenie tohto modelu dravec-korist' z hľadiska exponenciálnych alebo iných základných funkcií. Je potrebné vypočítať číselné riešenia.

Existujú štyri parametre, dve konštanty μ_1 a μ_2 a dve počiatočné podmienky, $\eta = y_1(0)$, $\eta = y_2(0)$. Začínáme s $\eta_1 = \mu_1$ a $\eta_2 = \mu_2$, tak y_1 a y_2 sú nulové a populácie zostávajú konštantné pri ich počiatočných hodnotách. Inými slovami môžeme povedať, že bod (μ_1, μ_2) je rovnovážny bod. Bod $(0; 0)$ je ďalší rovnovážny bod, ale nie veľmi zaujímavý. Výpočet, ktorý vidíme na obrázku 12, zobrazuje priradenú hodnotu čas. Táto hodnota určuje hodnotu t , keď sa populácia vráti k ich pôvodným hodnotám uvedeným v n (počiatočné hodnoty populácie dravec-korist'). Kód sa integruje do troch z týchto časových intervalov, a tak sa na konci dostaneme späť k miestu, kde sme začali, v našom prípade je to hodnota 6.3922. (Moler, 2011)



Obrázok 13 model rastu populácie s pridanou hodnotou čas

Na obrázku 14 nám červený bod ukazuje model v rovnovážnom bode (μ_1, μ_2) a modrá bodka predstavuje počiatočný bod (η_1, η_2).

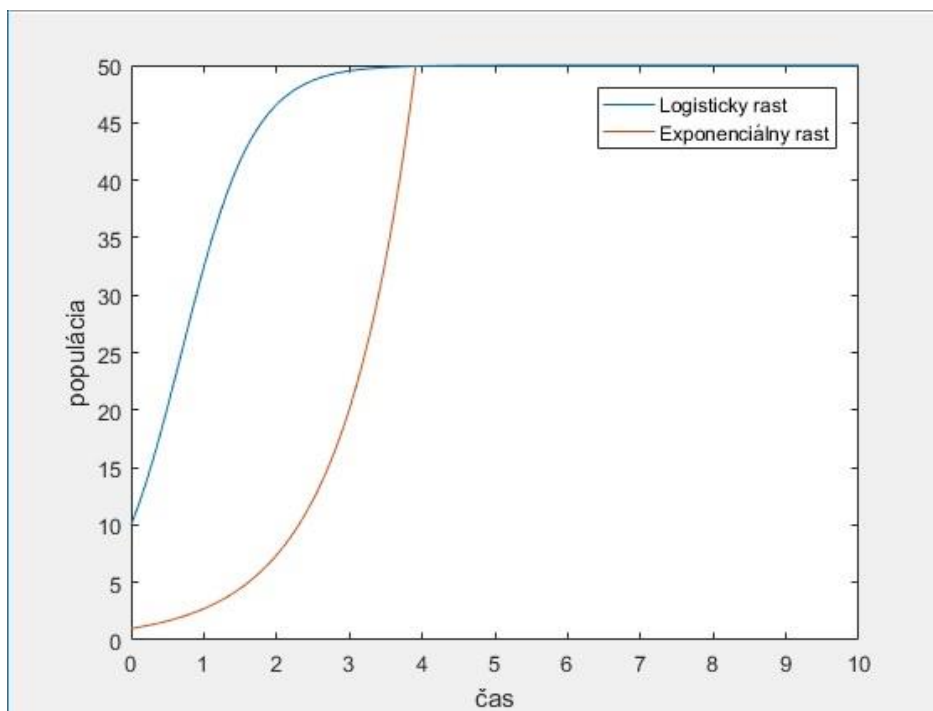


Obrázok 14 rovnovážny a počiatočný bod

Pozoruhodnou vlastnosťou modelu Lotka-Volterra je, že riešenia sú vždy periodické. Populácie sa vždy vrátia k pôvodným hodnotám a opakujú cyklus. Táto vlastnosť nie je zrejma a nie je ju ľahké dokázať. Nelineárne modely majú zriedkavé periodické riešenia. Zložitý aspekt výpočtu rovníc typu dravec-korisť určuje dĺžku obdobia. (Moler, 2011)

5.1 ZMENA PARAMETROV

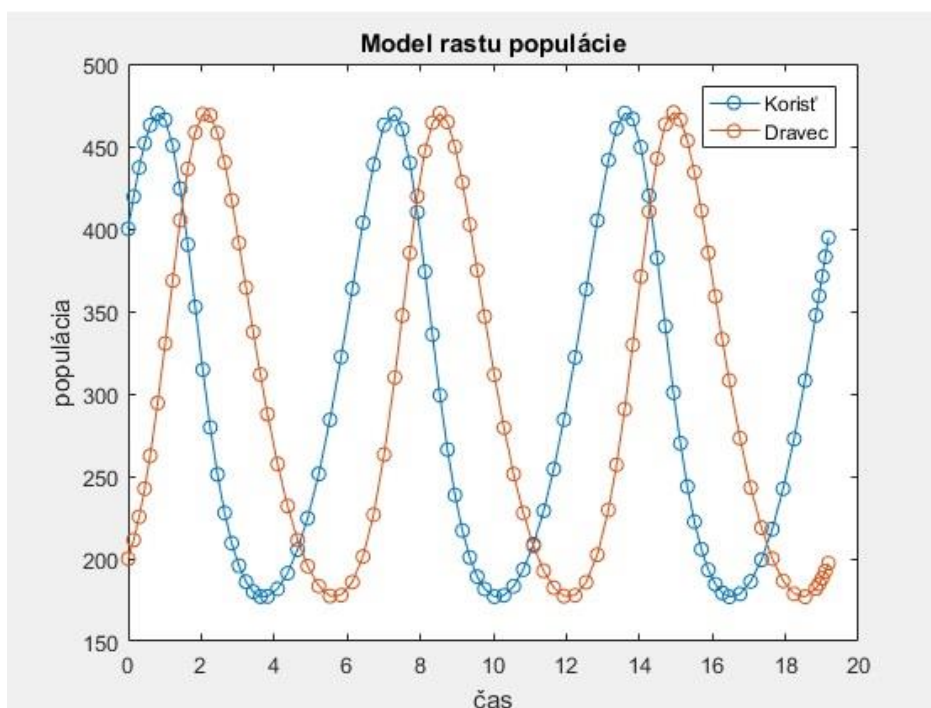
V tejto podkapitole sa budeme venovať zmene parametrov modelu rastu populácie. Upravíme model, s ktorým sme doteraz pracovali a budeme pozorovať zmeny, ktoré nastali. Zmeníme všetky tri parametre k (konštantu proporcionality zodpovedajúca veľkosti populácie), η (počiatočná veľkosť populácie), μ (nosnosť). V našom modeli mali jednotlivé parametre hodnoty : $k = 1$, $\eta = 1$, $\mu = 10$. Zmeníme parametre $k=2$, $\eta = 10$, $\mu = 50$.



Obrázok 15 zmenené parametre exponenciálneho a logistického rastu

Na obrázku 15 vidíme zmeny, ktoré sa udiali pri zmene parametrov. Obe krivky nemajú rovnakú počiatočnú hodnotu a ani počiatočný sklon nie je rovnaký, ako tomu bolo na obrázku 12 v úvode tejto kapitoly. Exponenciálna funkcia sa nemení. Logistická funkcia rastie rýchlejšie ako v prvom prípade, ale nikdy neprekračuje kapacitu prostredia, v tomto prípade 50.

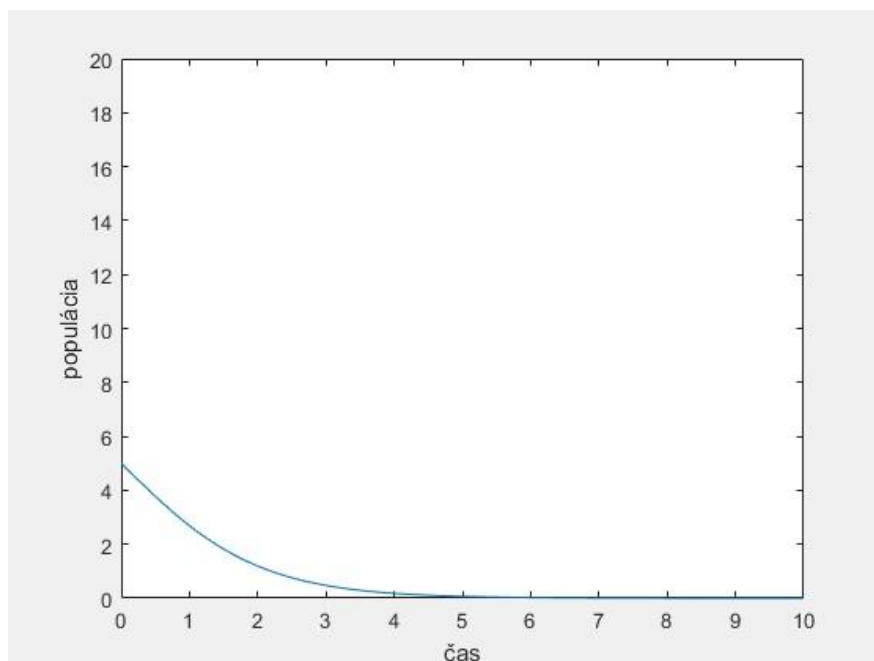
Obrázok 13 bol vytvorený na základe údajov $u = [280 \ 220]$ a počiatočných hodnôt dravcov a koristi $n = [350 \ 150]$. V tomto prípade nosnosť nastavíme na 300 pre obe populácie. Počiatočné hodnoty dravca a koristi nastavíme na 200 a 400.



Obrázok 16 Model rastu populácie

Z obrázku 16 vieme usúdiť, že obe populácie sa správajú cyklicky. Kruhy, ktoré sú umiestnené na obrázku, ukazujú body, v ktorých ode45 vypočíta riešenie. Krivky v blízkosti minimá sú širšie a vyžadujú viac krokov na výpočet ako krivky v blízkosti maxima (Moler, 2011).

Na úvod kapitoly sme hovorili, že konštanta proporcionality k zodpovedá veľkosti populácie. Doteraz sme pracovali z kladnou hodnotou parametra k , ako v úvode na obrázku tak aj v prvej zmene parametrov (obrázok 14). Teraz parametru k pridáme zápornú hodnotu a budeme sledovať, ako sa bude populácia správať.



Obrázok 17 záporný parameter k

Na obrázku 17 môžeme vidieť, že ak je k záporné, populácia nám klesá. Doposiaľ sme pracovali s kladným parametrom k a populácia sa nám zvyšovala. V tejto kapitole sme sa inšpirovali modelom, ktorý Moler predstavil v elektronickej publikácii *Experiments with MATLAB*.

6 MODEL DRAVEC-KORISTĚ

Model dravec-korist' Lotka-Volterra je jeden z najjednoduchších modelov popisujúcí interakciu dravca a koristi. Model bol pôvodne navrhnutý Alfredom Jamesom Lotkom v teórii autokatalytických chemických reakcií v roku 1910. Ide o logistickú rovnicu, ktorú pôvodne odvodil Pierre François Verhulst. V roku 1920 Lotka rozšíril model na "organické systémy" pomocou rastlinných druhov a bylinožravých živočíšnych druhov, a ako príklad v roku 1925 využil rovnice na analýzu interakcie dravec-korist' vo svojej knihe o biomatematike. Rovnaký súbor rovníc publikoval v roku 1926 Vito Volterra, matematik a fyzik, ktorý sa zaujímal o matematickú biológiu. Analýza Volterru bola inšpirovaná jeho interakciou s morským biológom Humbertom D'Anconom. D'Ancona študoval údaje o počte ulovených rýb v Jadranskom mori a všimol si, že sa zvýšilo percento ulovených rýb počas prvej svetovej vojny. To ho znepokojovalo, pretože rybolov bol počas vojnových rokov veľmi znížený. Volterra vyvinul svoj model nezávisle od Lotku a použil ho na vysvetlenie pozorovania D'Ancony. (Mohan, a kol. 2017)

Dravec-korist' je biologický systém, ktorý nám popisuje vzájomne pôsobenie dvoch druhov – dravca a koristi. Závislosť množstvo koristi na množstvo dravcov vieme vyjadriť nasledujúcim spôsobom: Čím viac rastie populácia koristi, tým viac pribúda dravcov. Väčšie množstvo dravcov zvýši tlak na korist', a tej tak začne ubúdať. Neskôr s klesajúcim množstvom koristi začne klesať aj množstvo dravcov, ktorému ubúda potrava. Následne s ubúdajúcim množstvom dravcov sa pokles množstva koristi zastaví a začne opäť stúpať. Dĺžka oscilácie kolíše medzi 5 a 10 rokmi podľa podmienok prostredia. Tento systém vieme opísať pomocou dvoch diferenciálnych rovníc.

$$\frac{dx}{dt} = Ax - Bxy \quad (26)$$

$$\frac{dy}{dt} = Dxy - Cy \quad (27)$$

X v týchto rovniciach predstavuje veľkosť populácie koristi a y naopak veľkosť populácie dravca. A , B , C , D sú parametre, ktoré nám opisujú vzájomne pôsobenie týchto dvoch druhov, t je čas.

Model Lotka-Volterra prináša množstvo predpokladov o životnom prostredí a vývoji populácie predátorov a koristi :

- korist' si nájde stále dostatok jedla,
- populácia dravca závisí od množstva populácie koristi,

- miera zmeny populácie je úmerná jej veľkosti,
- predátori majú stále chuť na korisť.

Pri použití diferenciálnych rovníc je riešenie deterministické a kontinuálne. To znamená, že generácie predátora aj koristi sa nepretržite prekrývajú.

Model pre korisť vyzerá nasledovne:

$$\frac{dx}{dt} = Ax - Bxy \quad (26)$$

Predpokladáme, že korisť má k dispozícii neobmedzené zdroje potravy a reprodukuje sa exponenciálne, ak nie je vystavená dravcom, tento exponenciálny rast je v rovnici vyššie uvedený termínom Ax . Ak nie sú prítomní dravci, potom výsledok rozmnožovania koristi bude exponenciálny nárast jej množstva

$$x(t) = x(0)e^{At} \quad (28)$$

Kde $x(0)$ znamená veľkosť populácie koristi v čase $t = 0$. Ak sú však prítomní dravci, počet koristi je zmenšený o to množstvo, ktoré dravci ulovili. Toto množstvo by malo byť priamo úmerné tomu, ako často sa dravec a korisť stretávajú, čiže súčin ich počtov xy (ak je počet dravcov alebo koristi nulový, nie je možné, že sa stretnú, a teda lov neprebíha). Ak však spravíme úmeru z počtu ulovenej koristi a počtu stretnutí xy a označíme ju ako B , zmena počtu koristi vplyvom lovu dravcom sa za jednotku času rovná $-Bxy$.

Dravec má nasledujúcu diferenciálnu rovnicu:

$$\frac{dy}{dt} = Dxy - Cy \quad (27)$$

Dravec potrebuje pre svoju existenciu loviť korisť. V tejto rovnici, Dxy predstavuje rast populácie dravcov. Cy predstavuje pokles dravcov v dôsledku smrti alebo migrácie. Toto vedie k exponenciálnemu rozkladu v neprítomnosti populácie koristi. Preto táto rovnica predstavuje zmenu veľkosti rastu populácie dravcov, mínus prirodzená smrť. (Samples, 2016)

Nemennosť veľkosti obidvoch populácií dosiahneme, ak sú zmeny ich rýchlosti nulové. Vložením nuly na ľavú stranu obidvoch rovníc získame systém dvoch rovníc, ktoré obsahujú dve neznáme x a y . Tento systém má dve riešenia:

$$x = 0, y = 0 \quad (29)$$

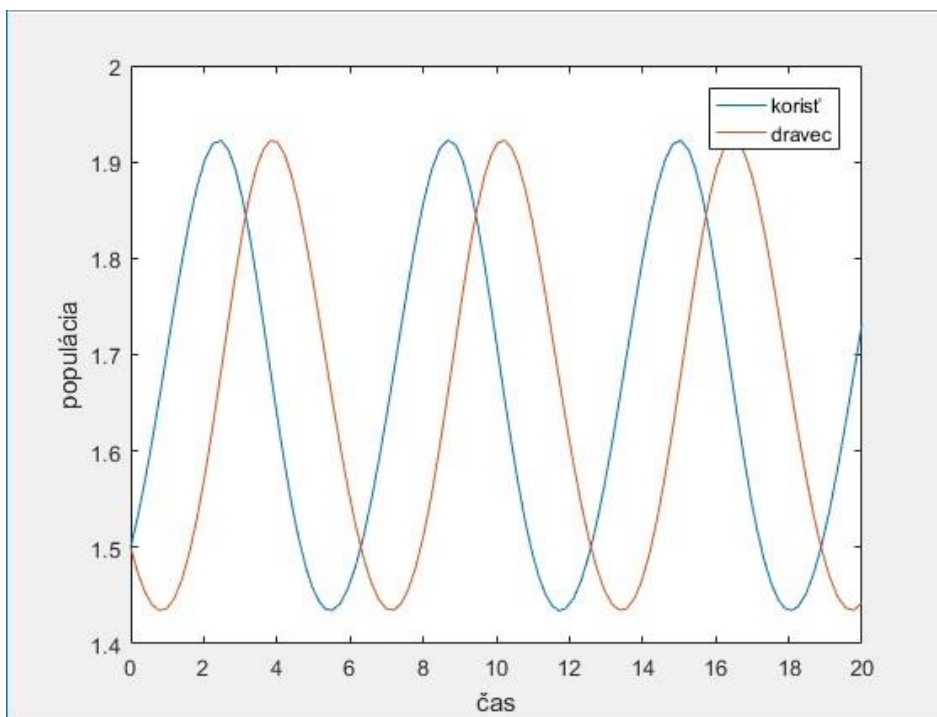
$$x = \frac{A}{B}, y = \frac{C}{D} \quad (30)$$

Prvý prípad je jednoduchý pretože počet dravcov a koristi je nulový. Druhý prípad je zaujímavý, nárast koristi je kompenzovaný lovom dravcov. Tento lov je tiež dostatočný na prirodzený pokles populácie dravcov. Prvé riešenie systému nie je prirodzené. Systém

môže získať údaje len umelým odstránením všetkej koristi ($x=0$) a dravec vymrú z dôvodu nedostatku potravy (k hodnote $y=0$ sa len priblížime, nikdy ju ale nedosiahneme).

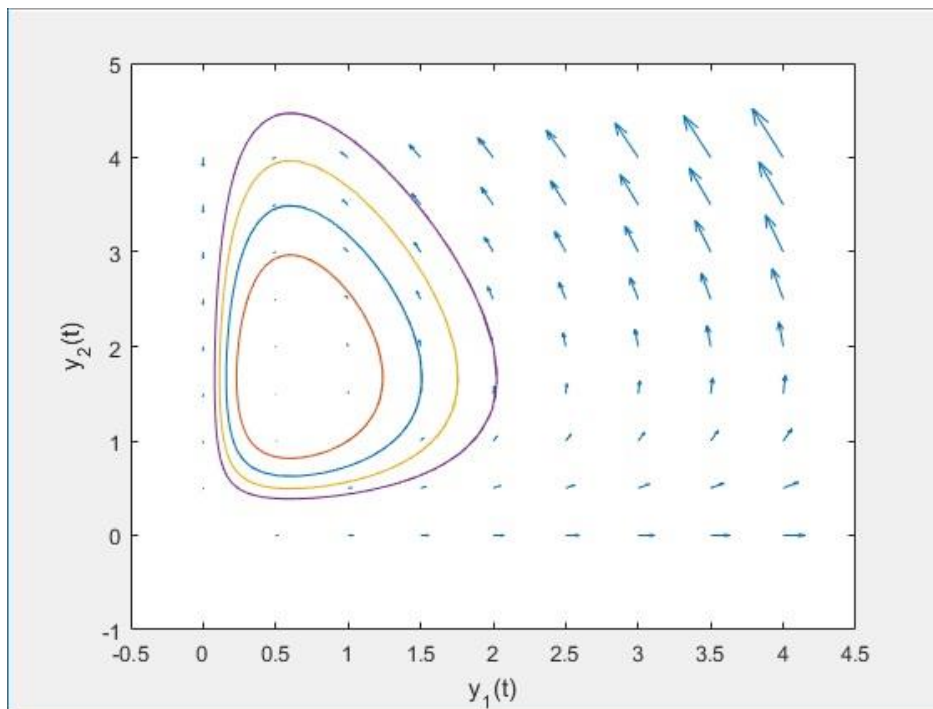
V Matlabe si vytvoríme program. Na výpočet použijeme rovnice (26) a (27), Kde x v týchto rovniciach predstavuje veľkosť populácie koristi a y naopak veľkosť populácie dravca. A = rast koristi C = miera úmrtnosti dravcov, B a D sú parametre, ktoré nám opisujú vzájomne pôsobenie týchto dvoch druhov, t je čas.

Pre simuláciu dravec-korist' sme použili nasledujúce hodnoty parametrov: $A = 1.0$, $B = 0.6$, $C = 1.0$, $D = 0.6$. Nastavíme počiatočné podmienky na 1.5 a časovú os t od 0 do 20. Na získanie riešení použijeme metódu ode45.



Obrázok 18 - Populačný graf

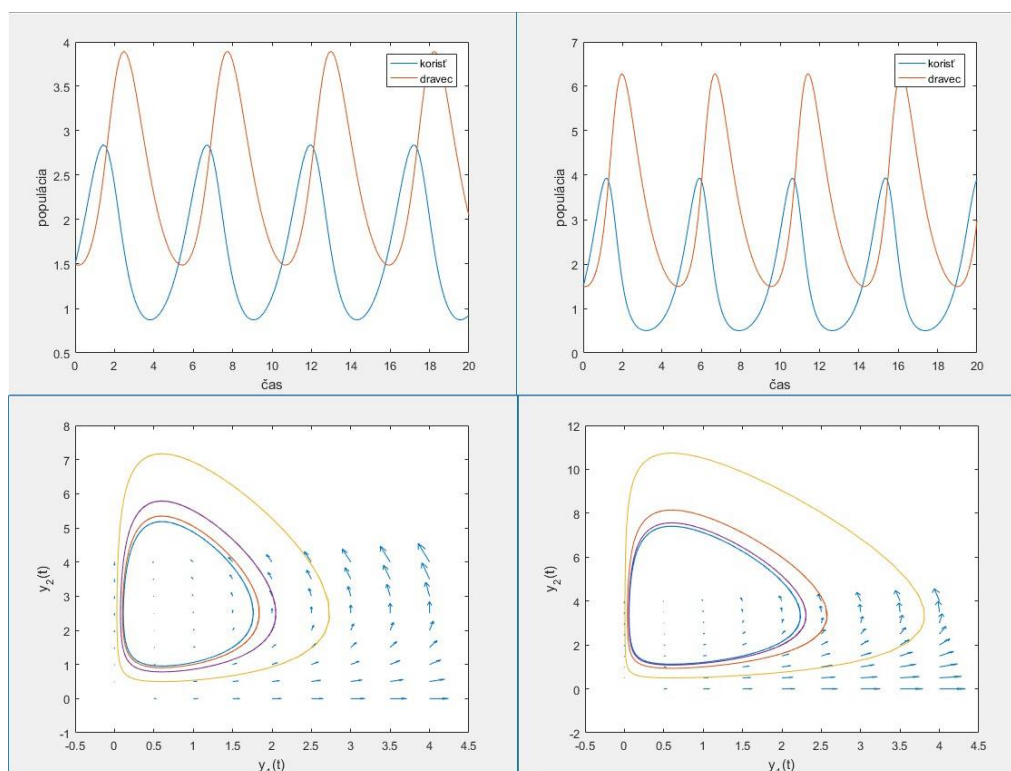
Na obrázku 18 vidíme populačný graf modelu Lotka-Voltera dravec-korist'. Červená krivka nám približuje veľkosť populácie dravcov v čase t , modrá krivka predstavuje veľkosť koristi v čase t . Z obrázka 18 vieme usúdiť, že správanie oboch populácií ukazuje opakujúce sa cykly.



Obrázok 19 - Populačné krivky

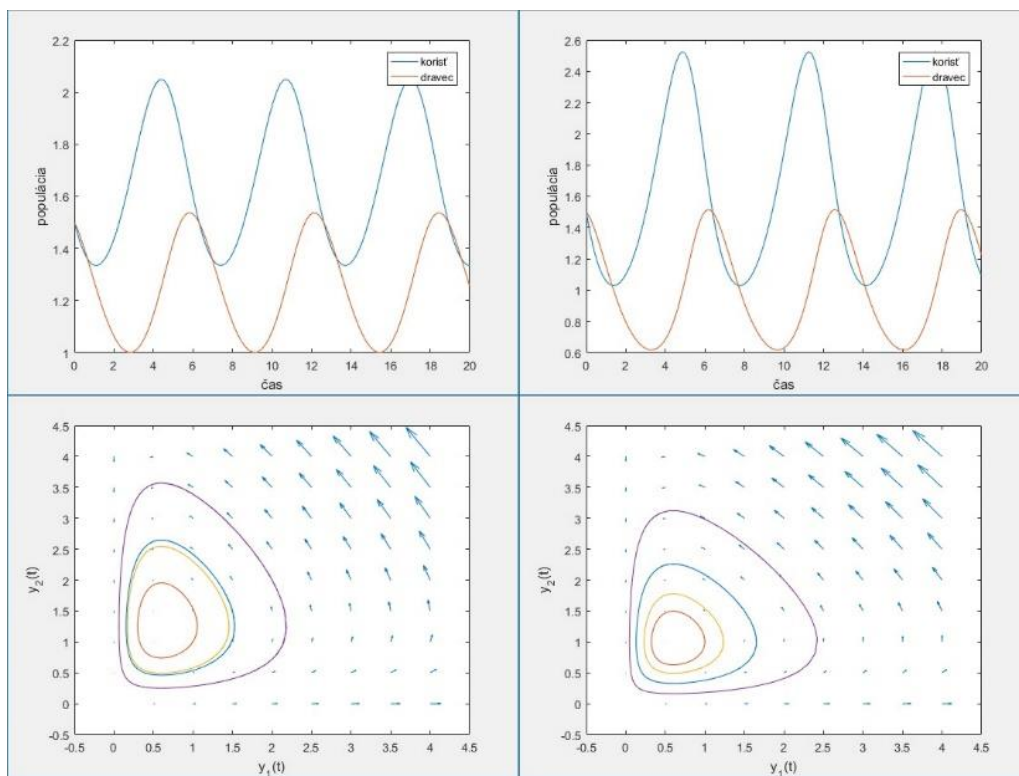
Jednotlivé kruhy, ktoré vidíme na obrázku 19, predstavujú cykly populácie grafu. Počet dravcov a koristi po čase osciluje, čo sa prejaví v uzavretí kriviek. Rôzne krivky predstavujú rôzne stavy počtov koristi a dravcov. Výstupom v Matlabe je aj rovnovážny stav, ktorý má súradnice $\frac{C}{D}, \frac{A}{B}$.

6.1 ZMENA PARAMETROV



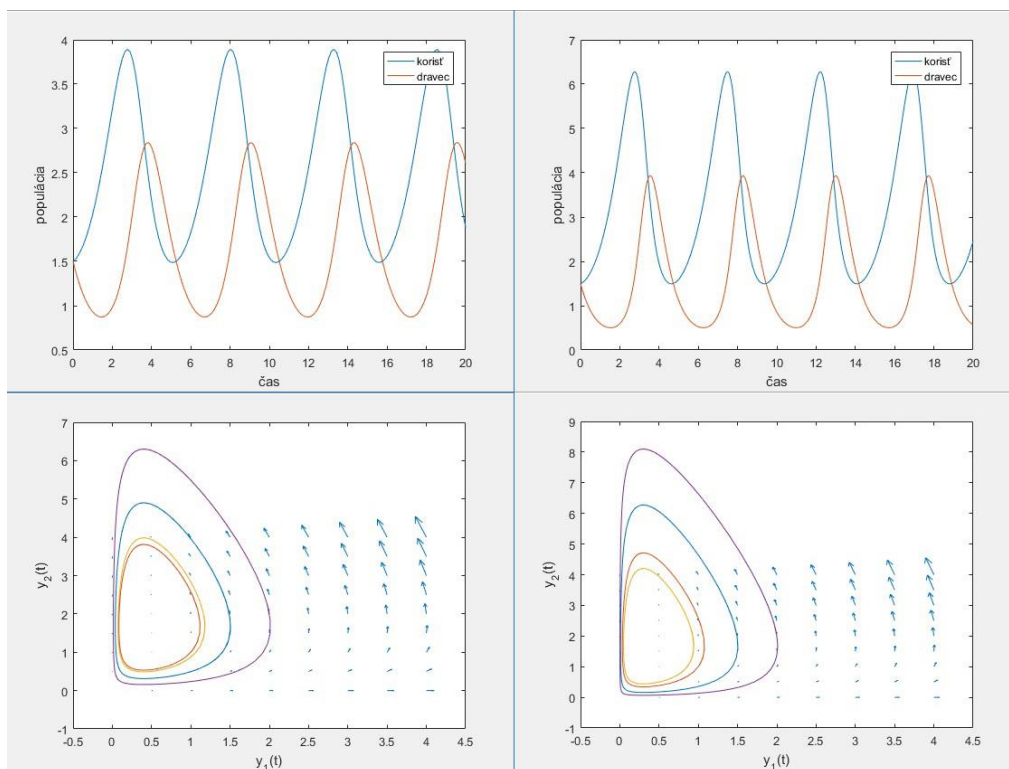
Obrázok 20 - Zmena parametra A z 1.0 na 1.5 a 2

Parameter A sme zmenili z 1 na 1.5 a 2. Ako môžeme vidieť na obrázku 20 máme na ľavej strane populačný graf so zmenou parametra na 1.5 a na pravej strane populačný graf s parametrom 2. Populačné krivky sú pod nimi. Z obrázka 20 vieme usúdiť, že pri zmene parametra A rastie populácia dravcov a súčasne s rastom populácie dravcov rastie i populácia koristi. Ak sa pozrieme na periódy, tak vidíme, že perióda medzi jednotlivých cyklov sa výrazne skracuje.



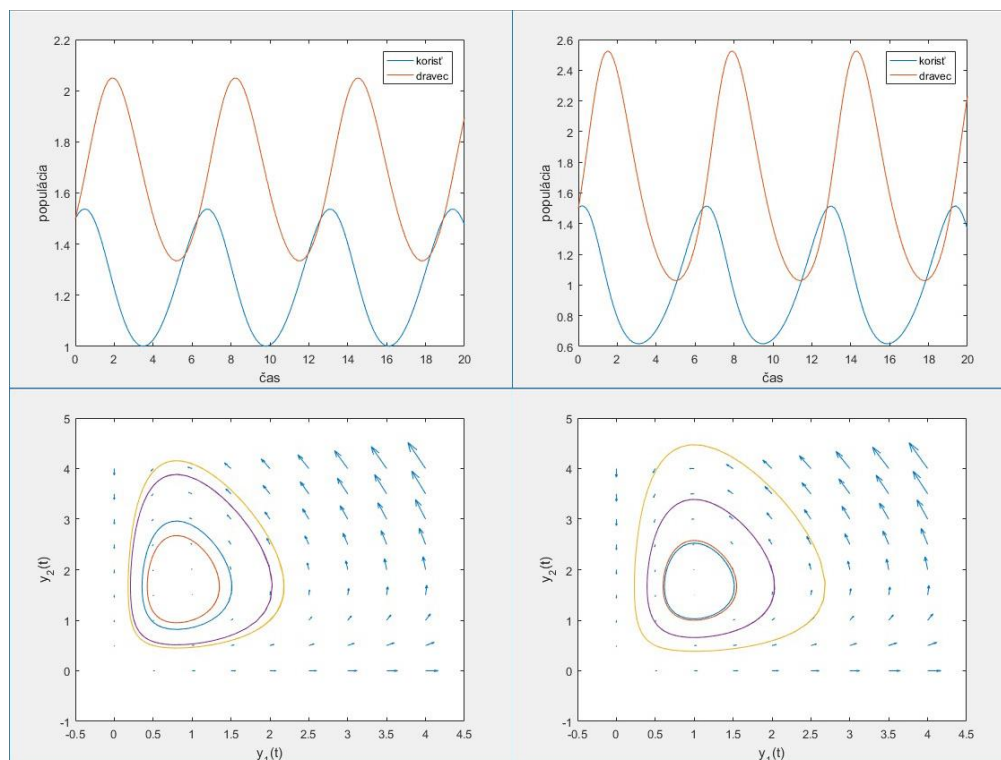
Obrázok 21 - Zmena parametra B z 0.6 na 0.8 a 1

Parameter B sme zmenili z 0.6 na 0.8 a 1. Ako môžeme vidieť na obrázku 21 máme na ľavej strane populačný graf so zmenou parametra na 0.8 a na pravej strane populačný graf s parametrom 1. Populačné krivky sú pod nimi. Z obrázka 21 môžeme usúdiť, že pri zmene parametru B pozorujeme zmeny populácie. Keď máme nedostatok koristi, tak dravec z nedostatku potravy takmer vyhynú. Z toho nám logický vyplýva, že sa nám premnoží populácia koristi. Keď sa pozeráme na periódu, môžeme usudzovať, že perióda sa medzi jednotlivými cyklami predlžuje.



Obrázok 22 - Zmena parametra C z 1 na 1.5 a 2

Parameter C sme zmenili z 1 na 1.5 a 2. Ako môžeme vidieť na obrázku 22 máme na ľavej strane populačný graf so zmenou parametra na 1.5 a na pravej strane populačný graf s parametrom 2. Populačné krivky sú pod nimi. Z obrázka 22 vieme usúdiť, že pri zmene parametru C rastie populácie koristi a súčasne s rastom populácie koristi, rastie i populácia dravcov, rovnako ako pri zmene parametra A . Ako môžeme vidieť, doba medzi jednotlivými cyklami sa skrakuje.



Obrázok 23 - Zmena parametra D z 0.6 na 0.8 a 1

Parameter D sme zmenili z 0.6 na 0.8 a 1. Ako môžeme vidieť na obrázku 23 máme na ľavej strane populačný graf so zmenou parametra na 1.5 a na pravej strane populačný graf s parametrom 2. Populačné krivky sú pod nimi. Z obrázka 23 môžeme usúdiť, že pri zmene parametru D počet dravcov výrazne prekročí populáciu koristi a perióda medzi jednotlivých cyklov sa predlžuje.

7 MODEL TYPU JEDEN DRAVEC, DVE KORISTI

V siedmej kapitole diplomovej práce si priblížime model, v ktorom budeme mať jedného dravca a dve koristi. Pod dravcom si predstavme líšku, pod korisťami zajaca a sliepku. Berieme do úvahy, že funkcie sú $x(t)$, $y(t)$ a $z(t)$, ktoré označujú populáciu líšok, zajacov a sliepok kedykoľvek v čase t

Funkcia $x(t)$ označuje populáciu líšok v čase t . Bez zajacov bude populácia líšok klesať podľa diferenciálnej rovnice $\dot{x} = -ax$, kde a je pôrodnosť mínus úmrtnosť.

Ideme modelovať populačný model, kde populácie budú mať kladné parametre. Vyššie uvedená diferenciálna rovnica pre líšky potrebuje ďalší termín na pravej strane, ktorý bude kladný, keď budú v modeli aj zajace. Zmeníme mieru pôrodnosti mínus miera úmrtnosti pri populácii líšok, ktorá má byť $-a + dy$. D je kladná konštanta, pretože zajace sú zdrojom potravy pre líšky. Preto upravíme diferenciálnu rovnicu pre líšky.

$$\dot{x} = -a + dy \quad (31)$$

Populácia zajacov bude mať logistický model, ak neexistujú žiadne líšky. To znamená, že počet pôrodov a úmrtí populácie zajacov sa bude meniť a bude sa rovnáť k $(M - y)$. Takže bez líšok model diferenciálnej rovnice pre zajace je $\dot{y} = k(M - y)$. Ak je však populácia líšok nulová, potom bude pre populáciu zajacov dodatočná úmrtnosť, takže miera pôrodnosti mínus úmrtnosť sa rovná k $(M - y) - cx$.

$$\dot{y} = k(M - y) - cx \quad (32)$$

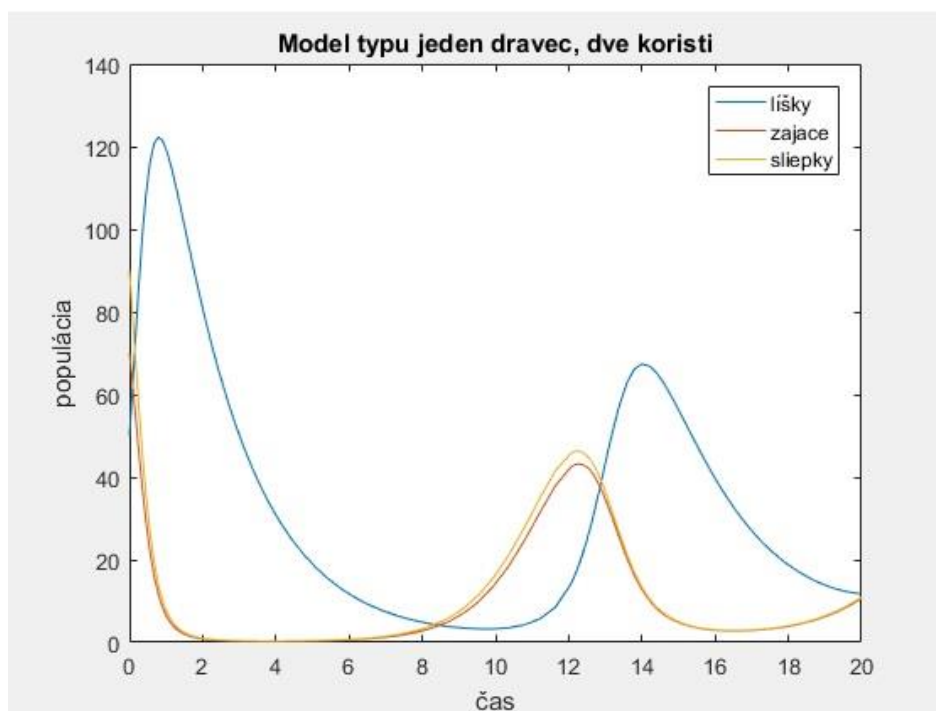
Ešte nám chýba posledná korisť a tou je sliepka. Populáciu sliepok označíme $z(t)$. Vzhľadom na to, že populácia sliepok je tiež korisťou, pôrodnosť mínus úmrtnosť sliepok musí byť podobná ako u líšok, takže pôrodnosť mínus úmrtnosť sa rovná k $(M - z) - cx$. Pôrodnosť mínus úmrtnosť obyvateľstva líšok sa zvýši na $-a + dy + dz$. Preto systém diferenciálnych rovníc bude mať tri rovnice: (White, 2008)

$$\dot{x} = -a + dy + dz \quad (33)$$

$$\dot{y} = k(M - y) - cx \quad (34)$$

$$\dot{z} = k(M - z) - cx \quad (35)$$

Na výpočet v Matlabe použijeme metódu ode45, s ktorou sme už pracovali pri predchádzajúcich modeloch. Počiatočné podmienky, ktoré sme použili $x(0) = 50$, $y(0) = 60$, $z(0) = 70$. Čiže líšky majú počiatočnú populáciu 50, zajace 60 a sliepky 70.

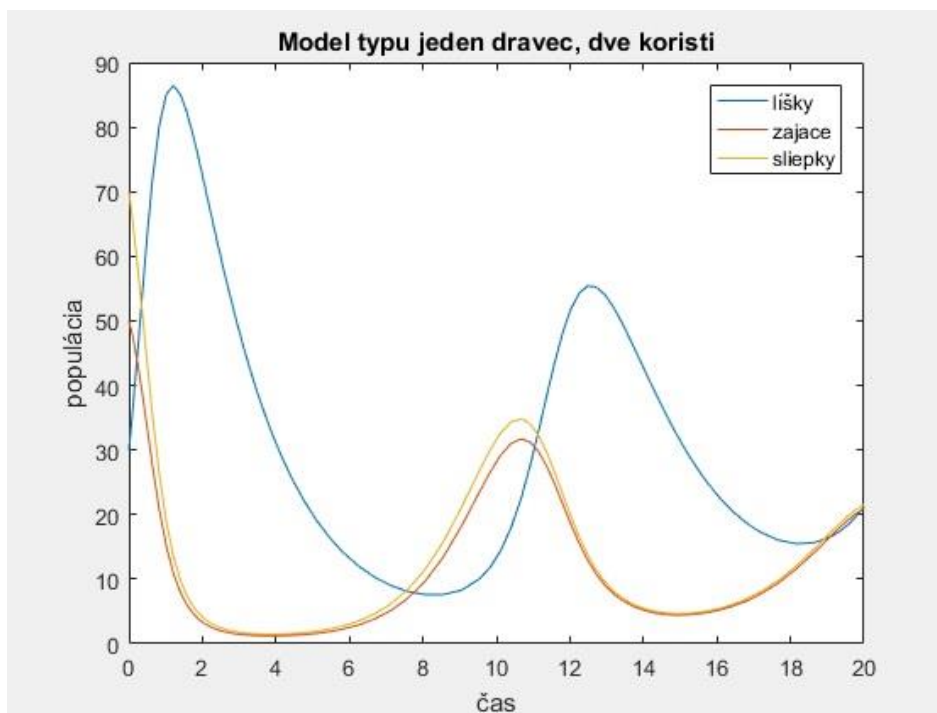


Obrázok 24 - Model typu jeden dravec, dve koristi

Na obrázku 24 vidíme ako populácia líšok prudko rastie, naopak populácia zajacov a sliepok radikálne klesá, až takmer vedie k zániku. Tento jav je však len dočasný, dravcom na vrchole ich populácie dôjde potrava a ich populácia sa začne postupne znižovať až pod úroveň koristi, ktorá sa naopak znižujúcim sa počtom dravcov začína zvyšovať. Populácia dravca sa potom opäť začne zvyšovať vďaka tomu, že bude mať dostatok koristi.

7.1 ZMENA PARAMETROV

Zmeníme počiatočné parametre v modeli s jedným dravcom a dvoma korisťami. Populáciu líšky teraz ešte viac znížime na $x(0) = 30$ a populácie zajacov zvýšime na 60 a počet sliepok ostane nezmenený. Budeme pozorovať, ako sa menia veľkosti populácii v čase t .



Obrázok 25 zmena parametrov – model typu jeden dravec, dve koristi

Na obrázku 25 môžeme vidieť, že populácie sa správajú podobne ako pri prvom modeli. Zníženie počiatočnej populácie dravca nemalo vplyv na jeho okamžitý rast, i keď svoje maximum dosiahlo trochu neskôr. Populácie koristi sa ale pomalšie znižovali. Ďalej si môžeme všimnúť, že populácia dravca klesla a populácie koristi rástli skôr ako v prvom prípade (obrázok 24). Podobným modelom sa zaoberal aj Robert White.

8 ROSENZWEIG-MACARTHUR MODEL S FUNKČNOU ODPOVEĎOU HOLLINGOVHO II. TYPU

V tomto modeli sa stretávame s paradoxom obohatenia. Michael Rosenzweig použil pojem paradox obohatenia v populačnej ekológii, ktorú vypracoval v roku 1971. Tento účinok popísal na šiestych modeloch dravec-korist', pri ktorých mala populácia koristi veľké množstvo potravy, čo spôsobilo destabilizáciu populácie dravca. Ak ma totiž korist' neobmedzení prísun potravy, jeho populácia rastie bez obmedzení a spôsobí, že aj populácia predátorov bude neúnosne rásť. Toto môže viesť až k vyhynutiu populácie dravcov. Pojem "paradox" Rosenzweig vysvetlil, ako protichodný účinok obohatenia, pretože obohacovanie je vnímané ako niečo s pozitívnym vplyvom na rast populácie koristi. (Rosenzweig, 1971)

Tento jav paradoxu slúži na zvýšenie nosnej kapacity prostredia. Predpokladáme, že keď zvýšime hodnotu nosnej kapacity prostredia, pomôžeme koristiam prežiť. Tento model sa pri zmene systému môže začať správať cyklicky, čo ma za následok kolísanie populácii. Niekedy sú veľmi nízke a niekedy naopak, veľmi vysoké. Najčastejšie používaný model je Rosenzweig-MacArthur model dravec-korist'.

Rosenzweig-MacArthur model vyzerá nasledovne:

$$\frac{dx}{dt} = rx \left(1 - \frac{x}{K}\right) - yf(x, y) \quad (36)$$

$$\frac{dy}{dt} = eyf(x, y) - my \quad (37)$$

Veľkosť populácie značí x a y . R miera rastu populácie koristi, K predstavuje kapacitu nosného prostredia, e je efektivita, s akou z konzumovanej koristi vzniknú nové dravce, m je miera úmrtnosti populácie dravcov a $f(N)$ funkčná odpoveď. Všetky parametre sú kladné.

Do modelu musíme ešte dosadiť funkčnú odpoveď Hollingovho druhého typu, $f(n) = \frac{\lambda x}{1+h\lambda x}$, ktorú sme spomínali vo štvrtej kapitole. Dostaneme nasledujúce vzorce:

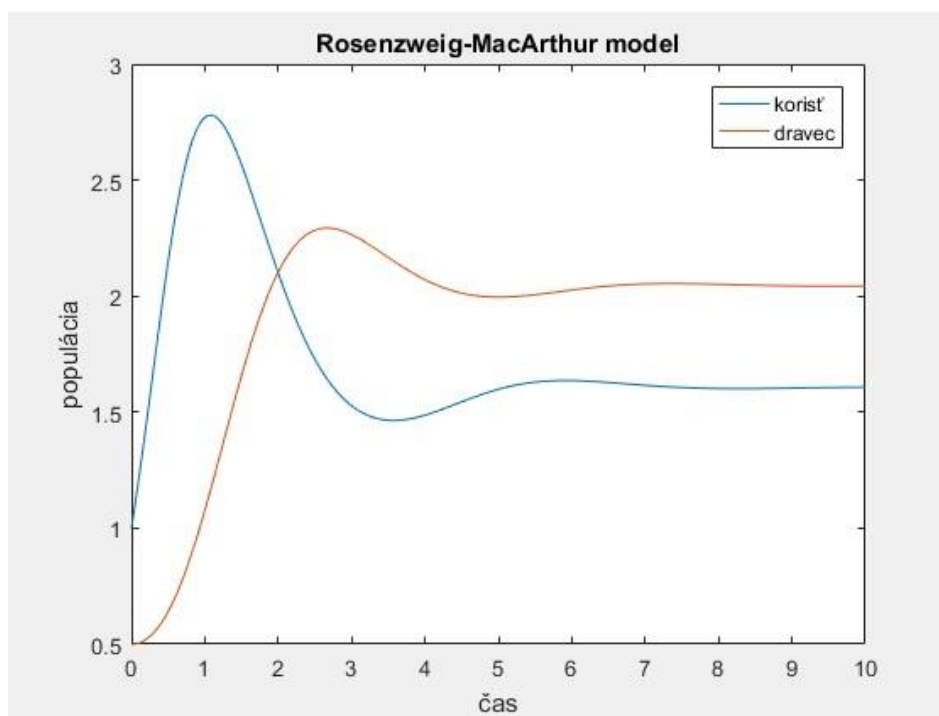
$$\frac{dx}{dt} = rx \left(1 - \frac{x}{K}\right) - \frac{\lambda xy}{1+h\lambda x} \quad (38)$$

$$\frac{dy}{dt} = \frac{e\lambda xy}{1+h\lambda x} - my \quad (39)$$

Pribudli nám nové parametre. λ predstavuje rýchlosť, s ktorou sa podarí dravcovi dolapiť korist'. H predstavuje čas na spracovanie jedinca koristi. (Rosenzweig, 1971)

Vytvorili sme si model podľa vzorca s pridanou funkčnou odozvou druhého typu. Nastavili sme si konkrétne hodnoty parametrov $e = 0.5$, $m = 1$, $h = 0.25$, $\lambda = 1$ a

$r = 3$. Kapacitu nosného prostredia sme nastavili na 4. Počiatočná populácia dravcov bola nastavená na 0.5 a populácia koristi na 1. Podľa takto zadaných parametrov sme dostali nasledujúci obrázok.

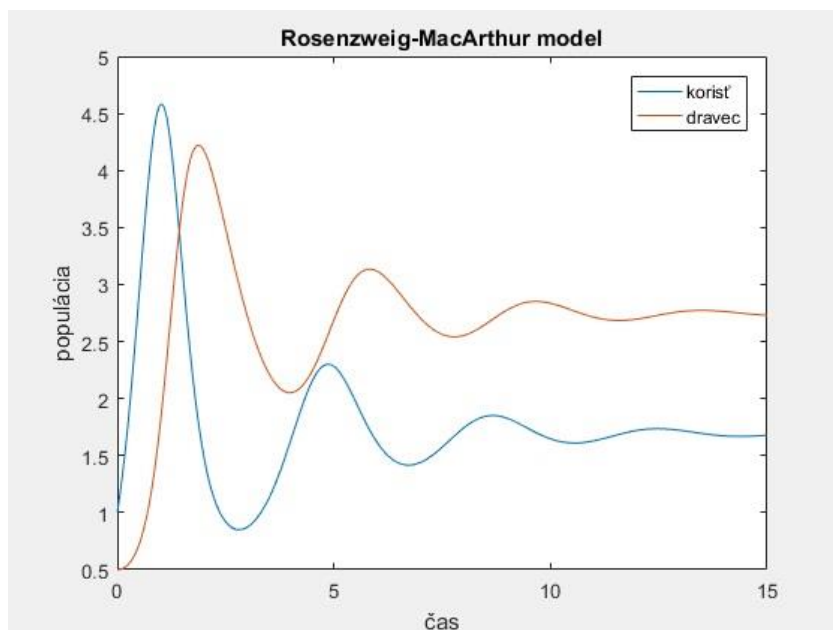


Obrázok 26 Rosenzweig-MacArthur model

Na obrázku 26 vidíme vývoj populácií dravca a koristi. Populácie koristi rastie rýchlejšie ako populácia dravca. Postupne sa so zvyšujúcim počtom dravcov znižuje počet koristi, neskôr sa počet oboch populácií ustáli.

8.1 ZMENA PARAMETROV

Zmeníme parametre modelu Rosenzweig-MacArthur. Paradoxné správanie vysvetlil Rosenzweig ako pozitívny vplyv na rast populácie koristi. Zvyšovaním kapacity prostredia pomôžeme koristiam prežiť. Preto zvýšime tento parameter zo 4 na 10. Ostatné parametre necháme nezmenené a budeme sledovať, ako sa budú vyvíjať populácie.



Obrázok 27 - Zmena parametra k modelu Rosenzweig-MacArthur

Na obrázku 27 vidíme, že zvýšením nosnej kapacity na 10 dosahujú obe populácie vyššie hodnoty, neskôr začnú kolísať veľkosti oboch populácií. Hodnoty oboch populácií sa budú opäť ušľachovať, aby sme to videli, zvýšili sme čas z 10 na 15. Detailnejší pohľad, ako sa vyvíja populácia koristi sme videli na obrázku 10, kde sme ukázali príklad Hollingovho II typu.

9 MODEL NICHOLSON-BAILEY

V tejto kapitole si predstavíme model Nicholson-Bailey. Tento model bol vyvinutý v tridsiatych rokoch minulého storočia, s cieľom popísať dynamiku populácie spojeného hostiteľsko-parazitného systému. Je pomenovaný po Alexanderovi Nicholsonovi a Victorovi Albertovi Baileymu. Hostiteľsko-parazitné systémy a systémy dravec-korist' môžu byť tiež reprezentované modelom Nicholson-Bailey. Model je úzko spätý s modelom Lotka-Volterra, ktorý opisuje dynamiku populácií dravec-korist' pomocou diferenciálnych rovníc.

Model používa diskretný čas na popísanie populačného rastu populácií hostiteľa-parazita. Model predpokladá, že parazity vyhľadávajú hostiteľov náhodne. Zvyčajne sa vyjadruje ako:

$$H_{t+1} = \lambda H_t e^{-a P_t} \quad (40)$$

$$P_{t+1} = c H_t (1 - e^{-a P_t}) \quad (41)$$

s H veľkosť populácie hostiteľa, P veľkosť populácie parazita, λ rastová rýchlosť hostiteľa, a je účinnosť vyhľadávania parazita a c je priemerný počet životaschopných vajíčok, ktoré parazit položí na jednom hostiteľovi.

Tento model možno vysvetliť na základe pravdepodobnosti. (Logan, a kol. 2009) Ustálený stav zodpovedajúci koexistencii hostiteľa a parazita v tomto modeli je daný:

$$P^* = \frac{\ln(\lambda)}{a} \quad a \quad H^* = \frac{\lambda \ln(\lambda)}{ac(\lambda-1)} \quad (42)$$

Model sleduje populáciu hostiteľského druhu H_t a druh parazita P_t . Vylepšený Nicholsonov-Baileyho model mapuje súčasnú úroveň populácie k budúcej úrovni populácie podľa vzorca:

$$H_{t+1} = f(H_t, P_t) = H_t \exp \left[r \left(1 - \frac{H_t}{K} \right) - a P_t \right], \quad (43)$$

$$P_{t+1} = g(H_t, P_t) = n H_t (1 - \exp[-a P_t]). \quad (44)$$

Pravá strana týchto rovníc môže byť považovaná za pravdepodobnosť stretnutia medzi hostiteľom a parazitom v proporcionálnej H_t , P_t . Pravdepodobnosť, že sa hostiteľ stretne s parazitmi dvakrát, je H_t , P_t^2 a pravdepodobnosť stretnutia n parazitov je H_t , P_t^n . a v našej rovnici je pravdepodobnosť parazitizmu hostiteľa, ktorý parazituje aspoň raz:

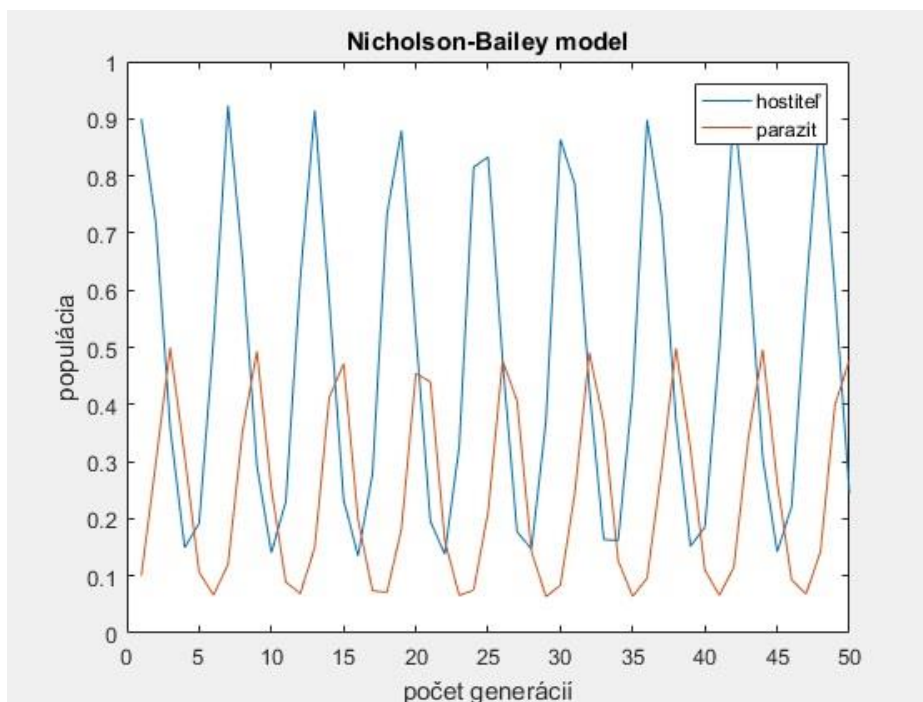
$$a H_t P_t + \frac{1}{2} a^2 H_t P_t^2 + \frac{1}{6} a^3 H_t P_t^3 + \dots = H_t [1 - e^{-a P_t}]. \quad (45)$$

Existujú tri pevné body pre Nicholsonov-Baileyho model: $(H, P) = (0, 0)$, $(K, 0)$ a zmiešaná populácia (H^*, P^*) , ktorá sa musí vypočítať pomocou nelineárnej rovnice:

$$H^* = f(H^*, P^*) \quad (46)$$

$$P^* = g(H^*, P^*) \quad (47)$$

Táto zmiešaná populácia je stabilná pre stredný rozsah plodnosti (r) a účinnosť parazita (a). Keď je a dostatočne malé, zmiešaná populácia je nestabilná a parazity zaniknú ($H_t \rightarrow K$). Ak je a dostatočne veľká, zmiešaná populácia je tiež nestabilná, avšak v tomto prípade môžu existovať stabilné limitné cykly, oscilácie vedúce k vyhynutiu alebo bifurkácie zdvojnásobujúce obdobie vedúce k chaosu. (Powell, 2009)

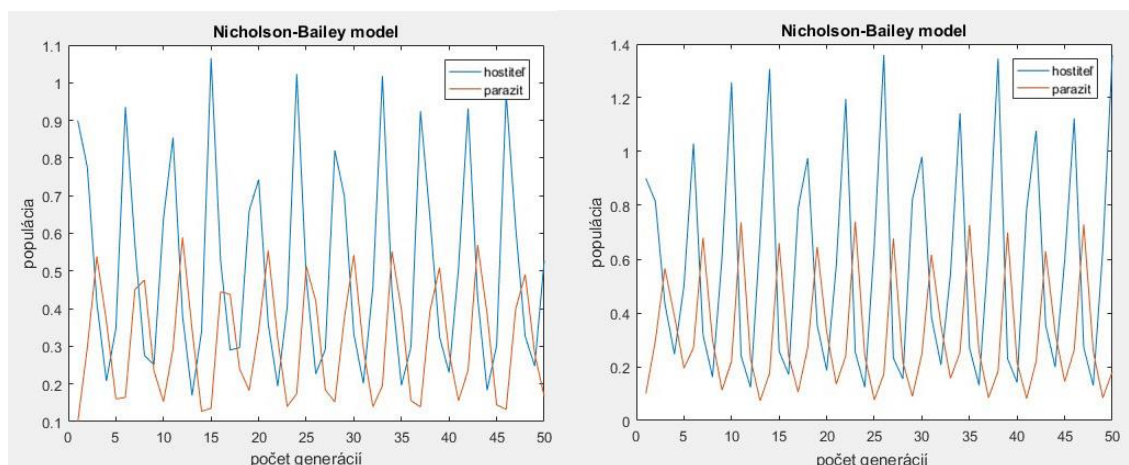


Obrázok 28 Nicholson-Bailey model

Na obrázku 28 vidíme model Nicholson-Bailey. Parametre sme nastavili nasledovne : $n = 1$ (n - parazity produkované úspešným zamorením); $a = 4$; $r = 1,75$; $k = 1$. Populácia parazitov začína na hodnote 0.1 a populácia hostiteľa začína na hodnote 0.9. Počet generácií = 50.

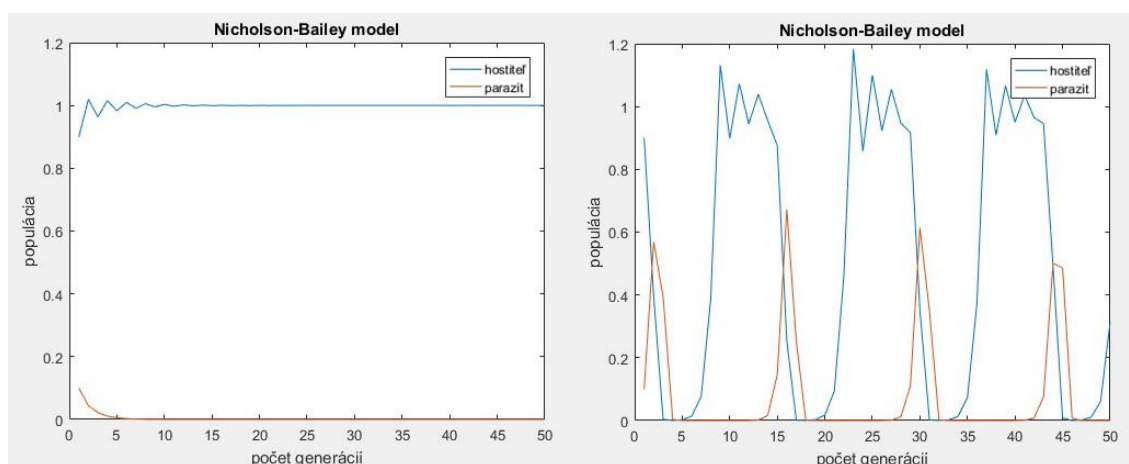
9.1 ZMENA PARAMETROV

Skúsime spustiť tento program pre čoraz väčšie hodnoty r a zistíme, ako sa bude meniť dynamika. Na obrázku 29 vidíme zmeny, ktoré nastali po zmene parametra r . Na ľavej strane obrázku je parameter r zmenený na 2.5 a pravá strana obrázku má hodnotu parametra r 3. Z obrázka 29 môžeme usúdiť, že pri zmene parametra r počet hostiteľov dosahuje stále vyššie maximálne hodnoty. Počet parazitov, ktorý v nich prežívajú sa tiež zvyšuje.



Obrázok 29 zmena parametra r v modeli Nicholson-Bailey

Teraz zameriame svoju pozornosť na parameter a (účinnosť parazita). Spustíme náš program a budeme sledovať, ako sa bude meniť populácia. Vyššie sme spomínali, že ak máme parameter a dostatočne malý, tak populácia je nestabilná a parazity nám zaniknú, avšak nestabilnou bude, aj keď bude parameter a dostatočne veľký.



Obrázok 30 zmena parametra a v modeli Nicholson-Bailey

Na obrázku 30 môžeme vidieť zmenu parametra a . Na ľavej strane obrázka 30 vidíme parameter a zmenšený oproti obrázku 24 zo 4 na 0.5. Nižšia účinnosť parazitov nám potvrdilo vyššie uvedené, že populácia sa stala nestabilná a parazity nám zanikli. Na pravej strane obrázka 30 sme naopak parameter a zvýšili oproti základným nastaveniam z obrázka 28 zo 4 na 10. Vidíme, že ak je hodnota parametra a dostatočne veľká, zmiešaná populácia je tiež nestabilná, avšak existujú stabilné limitné cykly, oscilácie vedúce k vyhynutiu. V tejto kapitole sme sa inšpirovali modelom, ktorý Powell predstavil v článku *Spatio-Temporal Models in Ecology*.

10 ALLEE EFEKT

V desiatej kapitole si predstavíme jav zvaný Allee efekt. Alleeho efekt bol opísaný profesorom Warderom Clydom Alleem v tridsiatych rokoch minulého storočia. Warder Clyde Allee bol americkým ekológom, ktorý bol uznávaný za jedného z veľkých priekopníkov americkej ekológie. Allee venoval svoj život výskumu výhod organizmov pôsobiacich v skupine. Prostredníctvom pozorovania skupín organizmov Allee zistil, že spolupráca je priaznivá a nevyhnutná. Allee efekt je fenomén v biológii charakterizovaný koreláciou medzi veľkosťou alebo hustotou obyvateľstva a priemernou individuálnou fyzickou kondíciou (často meranou ako miera rastu populácie na obyvateľa) populácie alebo druhu. Tento jav môže byť spôsobený napríklad lepšou obranou voči dravcom, alebo keď majú jedinci väčšiu šancu na prežitie v zime, alebo je už tak málo jedincov v populácii a sú tak roztrúsení v okolí, že sa nemajú pomaly šancu "stretnúť" nie to ešte nájsť partnera na párenie.

Jednoduchý matematický príklad Alleho efektu je daný modelom:

$$\frac{dN}{dt} = rN \left(\frac{N}{A} - 1 \right) \left(1 - \frac{N}{K} \right), \quad (48)$$

kde populácia má negatívnu rýchlosť rastu pre $0 < N < A$ a kladnú mieru rastu pre $A < N < K$ (za predpokladu $0 < A < K$). Toto je odklon od logistickej rovnice rastu:

$$\frac{dN}{dt} = rN \left(1 - \frac{N}{K} \right), \quad (49)$$

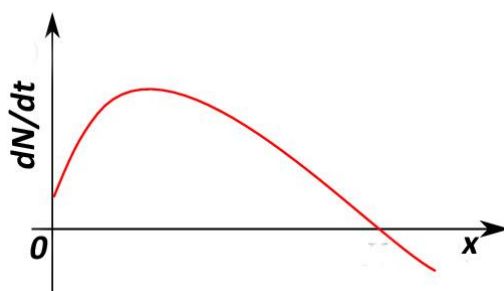
Kde: N = veľkosť populácie; r = skutočná miera rastu; K = nosnosť; A = kritický bod; dN/dt = miera rastu populácie.

Ako bolo vyššie spomenuté, Alleeho efekt môže mať za následok pri znížení veľkosti alebo hustoty populácie zníženie jej špecifickej miery rastu, v tomto prípade hovoríme o tzv. slabom Alleeho efektu. Niekedy môže dokonca dôjsť až k vyhynutiu populácie, potom hovoríme o tzv. silnom Alleeho efektu. (Courchamp, a kol. 2008)

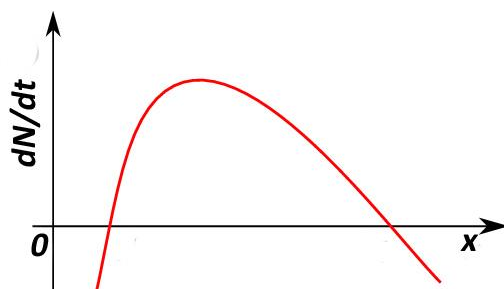
10.1 SILNÝ A SLABÝ ALLEE EFEKT

Silný Allee efekt (obrázok 31) je demografický Allee efekt s kritickou veľkosťou alebo hustotou populácie. Slabý Allee efekt (obrázok 32) je demografický Allee efekt bez kritickej veľkosti alebo hustoty populácie. Rozdiel medzi týmito dvoma pojmami je založený na tom, či daná populácia vykazuje kritickú veľkosť alebo hustotu obyvateľstva. Populácia, ktorá vykazuje slabý účinok Alleeho efektu bude mať nižšiu mieru rastu na obyvateľa (priamo súvisiacu s individuálnou zdatnosťou obyvateľstva) pri nižšej hustote alebo veľkosti obyvateľstva. Avšak aj pri tejto malej veľkosti alebo hustote obyvateľstva

bude populácia vždy vykazovať pozitívnu mieru rastu na obyvateľa. Populácia vykazujúca silný účinok Alleeho efektu bude mať kritickú veľkosť alebo hustotu obyvateľstva, pri ktorej sa miera rastu populácie stane negatívnou. Preto keď hustota alebo veľkosť populácie zasiahne číslo pod túto hranicu, populácia bude určená na vyhynutie bez akejkoľvek ďalšej pomoci. Silný Allee efekt sa často dá ľahšie demonštrovať empiricky pomocou údajov z časových radov, keďže je možné určiť veľkosť populácie alebo hustotu, pri ktorej sa miera rastu na obyvateľa stáva negatívnou. (Courchamp, a kol. 2008)



Obrázok 31 Silný Allee efekt⁷



Obrázok 32 Slabý Allee efekt⁸

⁷ Obrázok 30: <http://portal.matematickabiologie.cz>

⁸ Obrázok 31: <http://portal.matematickabiologie.cz>

ZÁVER

Cieľom diplomovej práce bolo nasimulovať rôzne druhy matematických modelov typu dravec-korist'.

V teoretickej časti práce sme si predstavili základné informácie o modelovaní a simulácii. Opisali sme jednotlivé fázy modelovania a tiež všetko, čo je s modelovaním späté. Predstavili sme si aj modelovanie pomocou rovníc. Nemohli sme zabudnúť ani na program, vďaka ktorému sme naprogramovali a simulovali rasty populácií na základe matematických modelov. Matlab využíva milióny inžinierov a vedcov po celom svete a patrí k najlepším programom pre technické výpočty. Teoretickú časť sme zakončili funkčnými odpoveďami, ktoré zaviedol Holling.

Praktickú časť sme začali s modelom rastu populácie. Predstavili sme jednoduchú diferenciálnu rovnicu, ktorú sme aj graficky znázornili. Ďalej sme predstavili exponenciálny a logistický rast až po model dravec-korist'. Pri zmene parametru sme v modeli upravili nosnosť aj veľkosti počiatočných populácií, čo malo za následok cyklické správanie.

Modelom Lotka-Voltera dravec-korist' sme sa zaoberali v šiestej kapitole našej diplomovej práce. Po predstavení zakladateľov modelu sme prešli na matematický model. V tomto modeli sme vytvorili populačný graf a populačnú krivku modelu Lotka-Voltera dravec-korist'. Na vytvorenom obrázku populačného grafu sme pozorovali veľkosti populácií dravcov a koristi v čase t . Videli sme, že správanie oboch populácií sa správalo cyklicky. V podkapitole Zmena parametrov sme pristúpili k zmene jednotlivých parametrov A , B , C , D . Zmenou parametra rastu koristi (A) nám rástla populácia dravcov a aj populácia koristi. Keď sme sa pozreli na periódy, tak sme videli, že perióda medzi jednotlivými cyklami sa výrazne skracovala. Parametre B a D nám opisovali vzájomne pôsobenie týchto dvoch druhov. Pri zmene B dravci takmer vyhynuli zo znižujúcou sa populáciou koristi, naopak pri zmene parametra D vyhynula takmer populácia koristi. V oboch prípadoch sa nám perióda predlžovala. Zvýšenie miery úmrtnosti dravcov (parameter C) znamenalo podobné riešenie ako pri zvýšení rastu koristi, ale vymenili sa nám veľkosti oboch populácií.

V siedmej kapitole sme pracovali s modelom z jedným dravcom a dvoma korisťami. V tomto modeli sme pracovali s tromi populáciami. Po nastavení počiatočných parametrov sme si všimli, ako populácia dravca prudko rástla, naopak populácie koristi klesali, až takmer zanikli. Po čase dravec už nemal k dispozícii toľko

koristi a jeho populácia klesla až pod úroveň koristi, ktorá sa zvyšovala vďaka menšiemu počtu dravcov. Neskôr sa opäť populácia dravca dostala do popredia.

Rosenzweig-MacArthur model dravec-korist' s funkčnou odpoveďou Hollingovho II typu sme si predstavili v ôsmej kapitole. V tomto modeli sme sa stretli s paradoxom obohatenia, javom, ktorý má pomôcť prežiť koristiam zvýšením nosnej kapacity prostredia.

S modelom, v ktorom sme sa stretli s hostiteľmi a parazitmi sme rozoberali v deviatej kapitole. Na začiatok modelu Nicholson-Bailey sme si predstavili jeho zakladateľom. Podľa rovníc sme v Matlabe pripravili model. V zmene parametrov sme sa zamerali na zmenu dvoch parametrov. Prvý parameter, ktorý sme upravili bol r , ktoré nám predstavovalo rozsah plodnosti. Z výsledkov sme usúdili, že pri zmene parametru r sa počet hostiteľov zvyšoval. S týmto zvyšovaním súviselo aj zvyšovanie populácie parazita, ktorý v nich prežíva. Druhý upravený parameter bol a (účinnosť parazita). Ten sme najprv znížili pod pôvodnú hodnotu z modelu, nestabilná populácia mala za následok zánik parazitov. Pri zvýšenej účinnosti parazitov bola populácia tiež nestabilná.

V desiatej kapitole sme si predstavili Alleho efekt. Jav, ktorý bol pomenovaný po Warderovi Clyde Alleem. Populácie môžu vykazovať slabý účinok a silný účinok Allee, rozdiel medzi týmito dvoma účinkami sme si zdôvodnili v podkapitole.

ZOZNAM BIBLIOGRAFICKÝCH ODKAZOV

- BAIGENT, S. 2018. Predator-Prey systems, University College London. Dostupné na internete:
<<http://www.ucl.ac.uk/~ucess29/page2/MATH3506Chapters//MATH3506Chapter4.pdf>>.
- COURCHAMP, F. – BEREĆ, J. – GASCIOGNE, J. 2008. Allee effects in ecology and conservation. Oxford, New York, USA: Oxford University Press.
- DAWES, J. H. P. – SOUZA, M. O. 2013. A derivation of Holling's type I, II and III functional responses in predator-prey systems. Journal of Theoretical Biology. 2013. Vol.327. ISSN 0022-5193. Dostupné na internete:
<<https://www.sciencedirect.com/science/article/pii/S002251931300091X>>.
- HAEFNER, J. W. 2001. The Holling Disc Equation. Indiana University. Spring, 2001. Dostupné na internete: <<http://www.indiana.edu/~oso/lessons/BioMath/hd-root.pdf>>.
- HUMUSOFT Dostupné na internete: <<http://www.humusoft.cz/matlab/details/>>.
- KARBAN, P. 2006. Výpočty a simulace v programech Matlab a Simulink. 1.vyd.. Brno: Computer Press, 2006. 220 s. ISBN 80-251-1301-9.
- KREJČÍ, A. – REITINGER, J. – TIHELKA, D. – VANĚK, J. 2014. Úvod do programového prostředí Matlab. 2014. 35 s. Dostupné na internete:
<<http://www.kky.zcu.cz/uploads/courses/matl/matlab2014.pdf>>.
- LOGAN, J. D. – WOLESENSKY, W.R. 2009. Mathematical Methods in Biology. Pure and Applied Mathematics; a Wiley-interscience Series of Texts, Monographs, and Tracts. John Wiley & Sons. 214 s. ISBN 978-0-470-52587-6.
- MATHWORKS Dostupné na internete: <<https://www.mathworks.com/>>.

- MOHAN, K. S. – VENKATESHA, P. 2017. Mathematical Modelling Of Predator-Prey Equations to Find the Population Growth & Decay. International Journal of Emerging Research in Management & Technology ISSN 2278-9359. Dostupné na internete:
<https://www.ermt.net/docs/papers/Special_Issue/2017/ICETSE/110.pdf>.
- MOLER, C. 2011. Experiments with MATLAB. Electronic edition published by MathWorks. Dostupné na internete: < <https://uk.mathworks.com/moler.html>>.
- MOLER, C. 2004. The Origins of MATLAB. Dostupné na internete:
<<https://uk.mathworks.com/company/newsletters/articles/the-origins-of-matlab.html>>.
- PELÁNEK, R. 2011. Modelování a simulace komplexních systémů. Nakladatelství Masarykovy univerzity, 2011. 236 s. ISBN: 978-80-210-5318-2. Dostupné na internete: < <http://www.radekpelanek.cz/dokumenty/ms-web.pdf>>.
- POWELL, J. 2009. Spatio-Temporal Models in Ecology; an Introduction to Integro-Difference Equations. Department of Mathematics and Statistics. Utah State University Dostupné na internete:
<<http://www.math.usu.edu/~powell/wauclass/labs.pdf>>.
- ROSENZWEIG, M. 1971. Paradox of Enrichment; Destabilization of Exploitation Ecosystems in Ecological Time. Science, 1971. vol. 171, no. 3969, s. 385-387. DOI:10.1007/978-3-662-03418-7-5.
- SAMPLES, L. 2016. Differential Equations. New York: White Word Publications. 2016. ISBN 978-1-283-50266-5. Dostupné na internete:
<<http://ebooks.wtbooks.com/static/wtbooks/ebooks/9781283502665/9781283502665.pdf>>.

WHITE, R.E. 2008. Predator-Two Prey. North Carolina State University. Dostupné na internete: <<http://www4.ncsu.edu/eos/users/w/white/www/white/bob.html>>.

ZAPLATÍLEK, K. – DOŇAR, B. 2004. MATLAB: tvorba uživatelských aplikací. 1.vyd.. Praha: BEN, 2004. 215 s. ISBN 80-730-0133-0.

ZOZNAM PRÍLOH

Príloha A - CD nosič

Príloha A – CD nosič

Priložené CD obsahuje:

A1 Diplomovú prácu vo formáte pdf

A2 Vytvorené programy v Matlabe