

Modeliranje dinamičnih sistemov z avtomatskim odkrivanjem zakonitosti

Ljupčo Todorovski, Sašo Džeroski

Laboratorij za umetno inteligenco, Institut Jožef Stefan, Jamova 39, 61111 Ljubljana, Slovenija
Elektronska pošta: Ljupco.Todorovski@ijs.si

Povzetek. Programi za avtomatsko odkrivanje zakonitosti poskušajo odkriti naravne zakonitosti v množici eksperimentalnih podatkov. Večina zakonitosti, ki so jih taki programi odkrili, opisuje statične situacije, kjer je opazovani sistem dosegel ravnotežje. V tem prispevku obravnavamo problem odkrivanja zakonitosti, ki opisujejo obnašanje dinamičnih sistemov. Predstavljamo program LAGRANGE, ki odkriva množico diferencialnih in/ali algebraičnih enačb na podlagi numeričnih podatkov o obnašanju dinamičnega sistema. Program smo preskusili na problemih generiranja različnih modelov dinamičnih sistemov iz podanih primerov obnašanja.

Ključne besede: umetna inteligenca, avtomatsko učenje, avtomatsko odkrivanje zakonitosti, identifikacija dinamičnih sistemov.

Modeling Dynamic Systems with Machine Discovery

Abstract. Machine discovery systems help humans to find natural laws from collections of experimentally collected data. Most of the laws found by existing machine discovery systems describe static situations, where a physical system has reached equilibrium. In this paper, we consider the problem of discovering laws that govern the behavior of dynamic systems. We present LAGRANGE, a system that constructs a set of differential and/or algebraic equations from numerical descriptions of dynamic system behavior. We illustrate its use by generating a variety of dynamic system models from example behaviors.

Key words: artificial intelligence, machine learning, machine discovery, dynamical system identification.

1 Uvod

Ker je odkrivanje zakonitosti obnašanja oziroma modeliranja dinamičnih sistemov zelo pogost problem, bi uspešna metoda za avtomatsko modeliranje le-teh imela daljnosežne posledice. Večina obstoječih metod za identifikacijo dinamičnih sistemov predvideva znano strukturo modela oziroma obliko diferencialnih enačb, ki opisujejo dinamiko sistema. Na osnovi izmerjenega obnašanja modeliranega sistema poskušajo omenjene metode identificirati parametre oziroma določiti njihove optimalne vrednosti v enačbah dinamike.

Čeprav obstaja več različnih programov za avtomatsko odkrivanje zakonitosti (BACON [15,16], FAHRENHEIT [21,16], ABACUS [10]), se noben ni uveljavil kot orodje za modeliranje realnih sistemov. Vzrok za to je tudi nezmožnost teh programov za odkrivanje zakonitosti v dinamičnih domenah [2]. Večina zakonitosti, ki so jih taki sistemi odkrili, opisuje statične situacije, kjer je opazovani sistem dosegel ravnotežje.

Program LAGRANGE, ki uporablja tehnike avtomatskega odkrivanja zakonitosti, ne zahteva vnaprej podane oblike enačb, temveč določa hkrati obliko in

parametre enačb, ki opisujejo obnašanje dinamičnega sistema. Iz podanega obnašanja, to je zaporedja meritev stanja dinamičnega sistema, generira LAGRANGE množico algebraičnih in/ali diferencialnih enačb, ki jim obnašanje zadošča. Na ta način LAGRANGE identificira tako strukturo kot parametre dinamičnega sistema.

LAGRANGE je sposoben odkriti enačbe dinamike za preproste dinamične sisteme iz sintetično generiranih podatkov. Tako je LAGRANGE uspešno rekonstruiral več modelov, ki opisujejo kemične in biološke procese, med katerimi so tudi nelinearni modeli oscilatornih procesov. Poleg tega je bil LAGRANGE preskušen na problemu iz dinamike tekočin ter na sistemu invertiranega nihala, ki je standardni problem za testiranje različnih metod vođenja dinamičnih sistemov.

V drugem poglavju podajamo kratek pregled področja avtomatskega odkrivanja zakonitosti in postavljamo metode, uporabljene v obstoječih programih, v povezavo z dinamičnimi sistemi. Tretje poglavje zajema podrobni opis algoritma, ki ga uporablja LAGRANGE. Opis in rezultate poskusov rekonstrukcije različnih modelov dinamičnih sistemov podajamo v četrtem poglavju. V petem opišemo razširitev programa LAGRANGE. Šesto poglavje povzema prispevke pro-

grama LAGRANGE in rezultate poskusov ter nakazuje smernice za nadaljnje delo.

2 Avtomatsko odkrivanje zakonitosti

Avtomatsko učenje (*machine learning*) je podpodročje umetne inteligence, ki proučuje metode za pridobivanje novega in organiziranje obstoječega znanja [5]. Različne oblike učenja se vrstijo od učenja po nasvetu (*learning by being told*) do učenja z odkrivanjem (*learning by discovery*) [4]. Značilnost učenja z odkrivanjem je v tem, da učni program nima na voljo nobenega učitelja, ki bi usmerjal učni proces. Učenje (odkrivanje) poteka samo na podlagi rezultatov poskusov in opazovanja.

Raziskovalci na področju avtomatskega učenja z odkrivanjem kot neusmerjenega učenja se lotevajo odkrivanja s treh različnih vidikov: pojmovno razvrščanje v skupine (*conceptual clustering*), oblikovanje pojasnjevalnih teorij (*formulation of explanation theories*) in odkrivanje opisnih zakonitosti (*induction of descriptive laws* ali *empirical discovery*) [16]. Nalogo avtomatskega odkrivanja zakonitosti lahko neformalno definiramo z naslednjim prepisom:

- **podano** imamo množico opazovanj (podatki)
- **poiskati** moramo eno ali več zakonitosti, ki veljajo za te podatke

Večina zakonitosti, ki so jih odkrili sistemi za avtomatsko odkrivanje, opisuje statične situacije, kjer je opazovani sistem dosegel ravnotežje. Te zakonitosti imajo obliko algebraičnih enačb, ki povezujejo opazovane spremenljivke. Metoda, ki bi jo uporabljal program za avtomatsko odkrivanje zakonitosti v dinamični domeni, mora izpolnjevati dva pogoja. Prvi je ta, da zna odkrivati zakonitosti z večji spremenljivkami brez dodatnih poskusov, drugi pa je, da mora odkrivati množico več zakonitosti, ki veljajo v celotni domeni.

Noben izmed obstoječih programov za avtomatsko odkrivanje zakonitosti, kot so BACON [15,16], FAHRENHEIT [21,16] in ABACUS [10], ne izpolnjuje obeh pogojev hkrati. Za odkrivanje zakonov z večimi spremenljivkami, programi BACON in FAHRENHEIT zahtevajo dodatne poskuse, pri čemer morajo biti vrednosti določenih spremenljivk modeliranega sistema konstantne, kar je v kontekstu dinamičnih sistemov neustrezno. Programi FAHRENHEIT in ABACUS odkrivajo množico več zakonitosti, vendar le te veljajo v disjunktnih področjih celotne domene.

3 LAGRANGE

Nalogo avtomatskega odkrivanja zakonitosti v dinamični domeni neformalno definiramo tako, da na podlagi izmerjenega obnašanja dinamičnega sistema v nekem končnem časovnem intervalu iščemo matematični model, ki opisuje dinamični sistem. Matematični model, ki ga odkrije LAGRANGE, je množica diferencialnih in/ali algebraičnih enačb.

LAGRANGE na vhodu zahteva meritve spremenljivk dinamičnega sistema v ekvidistančnih časovnih točkah

Tabela 1. Ekvidistančne meritve spremenljivk dinamičnega sistema

čas	spremenljivke dinamičnega sistema			
	X_1	X_2	...	X_n
t_0	x_{11}	x_{21}	...	x_{n1}
$t_0 + h$	x_{12}	x_{22}	...	x_{n2}
⋮	⋮	⋮	⋮	⋮
$t_0 + (N - 1)h$	x_{1N}	x_{2N}	...	x_{nN}

(tabela 1). Lahko podamo tudi merske enote posameznih spremenljivk. Dodatno je treba opredeliti vrednosti petih parametrov: red o dinamičnega sistema je stopnja najvišjega časovnega odvoda, ki nastopa v enačbah dinamike, največja globina d novih spremenljivk, ki jih uvajamo z množenjem, največje število spremenljivk r , ki lahko nastopajo v enačbah, ter pragovi t_R in t_S za ocenjevanje signifikantnosti enačb.

Izhod programa LAGRANGE je množica enačb oblike:

$$F_i(X_1, \dots, X_n, \dot{X}_1, \dots, \dot{X}_n, \dots, X_1^{(o)}, \dots, X_n^{(o)}) = 0, \quad (1)$$

pri čemer so X_1, \dots, X_n spremenljivke dinamičnega sistema, o red dinamičnega sistema, ter F_i polinomske funkcije stopnje d ali manj.

Algoritem LAGRANGE, katerega formalni opis je podan v tabeli 2, v grobem razdelimo na tri korake: računanje časovnih odvodov, uvajanje novih spremenljivk in iskanje enačb z linearno regresijo. Podroben opis algoritma LAGRANGE je podan v [18,8,9].

3.1 Računanje časovnih odvodov

V prvem koraku LAGRANGE izračuna časovne odvode spremenljivk dinamičnega sistema do uporabniško določene stopnje o . Privzeta vrednost za o je 1. Časovne odvode računamo z numeričnim odvajanjem po naslednji formuli ([3], str. 73)

$$\dot{x}_i = \frac{1}{12h}(x_{i-2} - 8x_{i-1} + 8x_{i+1} - x_{i+2}),$$

pri čemer so x_i in \dot{x}_i ustrezne vrednosti spremenljivke X in njenega odvoda \dot{X} v točkah $t_0 + (i - 1)h$. V primerih, ko je mogoče natančnejše merjenje odvodov spremenljivk dinamičnega sistema kot njihovo računanje po zgornji formuli, podamo meritve odvodov na vhodu programa LAGRANGE in jih ne računamo numerično.

3.2 Uvajanje novih spremenljivk

LAGRANGE uvaja nove spremenljivke na dva načina.

Pri prvem uporablja informacijo o merskih enotah spremenljivk dinamičnega sistema. Če je merska

Tabela 2. Formalni opis algoritma LAGRANGE

```

Lagrange( $V_S, o, d, r, t_R, t_S$ )
1.  $V := V_S$ 
    $D_0 := V_S$ 
   for  $i := 1$  to  $o$  do
      $D_i := \emptyset$ 
     forall  $x \in D_{i-1}$  do
        $D_i := D_i \cup \left\{ \frac{d}{dt} x \right\}$ 
      $V := V \cup D_i$ 
2. forall  $x \in V_S \wedge \dim(x) = [\text{rad}]$  do
    $V := V \cup \{\sin x, \cos x\}$ 
    $V_1 := V$ 
   for  $l := 2$  to  $d$  do
      $V_l := \emptyset$ 
     forall  $(x, y) \in V_1 \times V_{l-1}$  do
        $V_l := V_l \cup \{x \cdot y\}$ 
       (* brez duplikatov spremenljivk *)
      $V := V \cup V_l$ 
3.  $M := \emptyset$ 
   for  $i := 1$  to  $r$  do
     forall  $V_R \in \mathcal{P}(V) \wedge |V_R| = i$  do
       izberi odvisno spremenljivko  $y \in R$ 
        $V_R := V_R \setminus \{y\}$ 
       if  $LR(y, V_R)$  je signifikantna then
          $M := M \cup \left\{ y = c_0 + \sum_{x \in V_R} c_x x \right\}$ 
         (* dodaj enačbo v model *)
return( $M$ )
    
```

enota spremenljivke X radian, uvedemo dve novi spremenljivki $\sin X$ in $\cos X$.

Pri drugem načinu uvajamo nove spremenljivke z množenjem starih. Pri tem globina spremenljivk postopoma narašča. Največjo globino novih spremenljivk d določi uporabnik, privzeta vrednost je 2. Globina spremenljivke $Y = \prod_{i=1}^n X_i^{\alpha_i}$ ($\alpha_i \geq 0$) je enaka $\sum_{i=1}^n \alpha_i$. Tako ima na primer stara spremenljivka X_1 globino 1 ter novi spremenljivki $X_1 X_2$ in $X_1^2 X_2^3$ globini 2 in 5.

3.3 Iskanje enačb z linearno regresijo

LAGRANGE v tretjem koraku generira vse mogoče linearne enačbe, ki vsebujejo največ r spremenljivk in ocenjuje njihovo signifikantnost glede na podane meritve. Če je signifikantnost nad pragom, ki ga določi uporabnik, dodamo enačbo v model. Vrednost r določi uporabnik, privzeta vrednost je 3.

Konstantne člene enačb računamo z linearno regresijo [19]. Linearna regresija na podlagi meritev m neodvisnih spremenljivk X_1, X_2, \dots, X_m ter odvisne Y

določi koeficiente c_0, c_1, \dots, c_m tako, da je kvadratna napaka

$$E = \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad \hat{y}_i = c_0 + \sum_{j=1}^m c_j x_{ji}$$

čim manjša. V zgornji formuli so $y_i, x_{1i}, \dots, x_{mi}$ ustrezne meritve odvisne spremenljivke Y ter neodvisnih spremenljivk X_1, X_2, \dots, X_m v točkah $t_0 + (i-1)h$.

LAGRANGE uporablja dve merili za ocenjevanje signifikantnosti enačb, dobljenih z linearno regresijo:

1. Korelacijski koeficient R ([19], str. 274):

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2}, \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$$

R zavzame vrednosti iz intervala $[0, 1]$. Večja vrednost R pomeni večjo signifikantnost enačbe in nasprotno. Enačbo dodamo v model, če je $R > 1 - t_R$. Prag t_R določi uporabnik, privzeta vrednost je 0.01.

2. Standardizirano odstopanje S :

$$S^2 = \frac{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}{\bar{y}^2 + e^{-\bar{y}^2}}$$

Standardizirano odstopanje je definirano kot kvadratni koren razmerja med povprečno kvadratno napako in kvadratom povprečja \bar{y}^2 odvisne spremenljivke. Člen $e^{-\bar{y}^2}$ bistveno spremeni vrednost S le pri $\bar{y} \approx 0$. Takrat je namreč S približno enak ne-standardiziranemu odstopanju ($S^2 = \frac{E}{N}$). Za druge vrednosti \bar{y} je vpliv tega člena zanemarljiv. S zavzame vrednosti iz intervala $[0, \infty)$. Čim manjši je S , tem večja je signifikantnost enačbe in nasprotno. Enačbo dodamo v model, če je $S < t_S$. Prag t_S določi uporabnik, privzeta vrednost je 0.01.

Če je linearna enačba oblike $y = c_0$ (ni neodvisnih spremenljivk, $m = 0$) je $R = 0$. Signifikantnost takih enačb ocenjujemo samo z drugim merilom. Za ocenjevanje signifikantnosti drugih enačb LAGRANGE lahko uporablja enega zgornjih meril ali pa njihovo kombinacijo.

Omenimo še, da so vrednosti koeficientov, ustrezne napake in signifikantnost enačbe odvisne od izbire odvisne spremenljivke. Za zdaj LAGRANGE vzame za odvisno spremenljivko največje globine. Med spremenljivkami iste globine izbere prvo po leksikografskem vrstnem redu.

4 Poskusi s programom LAGRANGE

Program LAGRANGE smo preskusili na problemih odkrivanja zakonitosti v domenah iz dinamike tekočin (sistem povezanih posod Utube), kemičnih reakcij (Chem ter oscilatorna kemična reakcija Brus), populacijske rasti

Tabela 3. Pregled poskusov s programom LAGRANGE

Domena	n	N	o	d	r	t_R	t_S	št. enačb	čas[s]
Utube	2	1000	1	1	3	0.01	0.01	6	0.66
Chem	2	1000	1	1	3	0.01	0.01	14	1.03
Brus	2	1000	1	3	3	0.01	0.01	4	16.62
Pop	2	1000	1	2	3	0.01	0.01	2	1.85
Monod	2	1000	1	2	3	0.01	0.01	4	1.83
Spring	1	1000	2	2	3	0.01	0.01	5	0.73
Cart	2	125	2	3	3	0.01	0.0001	4	221.39

(Pop in Monod) ter mehanske dinamike (vzmetno nihalo Spring in invertirano nihalo Cart).

Za vsako posamezno domeno je poskus potekal takole: najprej smo izbrali množico diferencialnih enačb, ki opisuje dinamični sistem, nabor začetnih vrednosti spremenljivk ter vrednosti parametrov, ki nastopajo v diferencialnih enačbah. Tako izbrane diferencialne enačbe smo z metodo Runge-Kutta četrte stopnje ([17], str. 550–554) integrirali v $N = 1000$ časovnih točkah s korakom $h = 0.01$. Dobljeni časovni potek vrednosti spremenljivk dinamičnega sistema smo podali na vhod programa LAGRANGE.

LAGRANGE je implementiran v programskem jeziku C, poskusi so izvedeni na delovni postaji Sun SPARC-Station10. V tabeli 3 podajamo pregled poskusov za posamezne domene.

Vrednosti parametrov o in d smo izbirali v odvisnosti od kompleksnosti diferencialnih enačb originalnega modela. Parameter r je imel pri vseh poskusih privzeto vrednost 3. Vrednosti pragov t_R in t_S so bile enake privzetim (0.01), razen za domeno invertiranega nihala. V tem primeru smo za ocenjevanje signifikantnosti enačb uporabili merilo S s pragom $t_S = 10^{-4}$. S privzetima vrednostima pragov t_R in t_S je LAGRANGE odkril preveč enačb. Vrednost praga t_S smo zmanjševali, dokler nismo dobili obvladljivega števila enačb. Pri zmanjševanju vrednosti praga ni potrebno ponovno poganjanje programa LAGRANGE. Zadostuje le izločanje enačb, ki ne ustrezajo zmanjšani vrednosti praga.

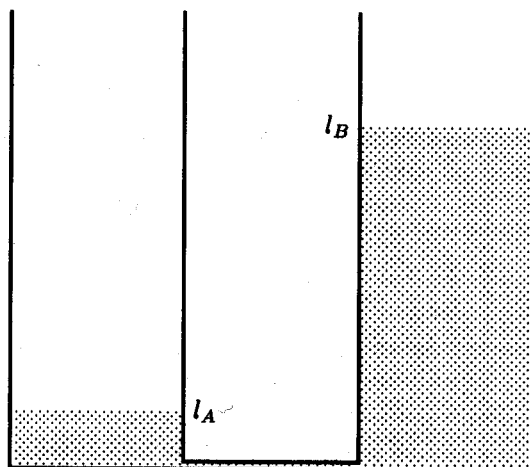
4.1 Sistem povezanih posod

Sistem povezanih posod [14] je sestavljen iz dveh posod A in B , ki sta povezani z ozko cevjo in napolnjeni s tekočino do ravni l_A oz. l_B (Slika 1). Razlika v ravneh povzroči pretakanje tekočine skozi cev iz posode B v posodo A . Spremembo ravni tekočin, ki nastane zaradi pretakanja, opišemo z naslednjim sistemom linearnih diferencialnih enačb:

$$\dot{l}_A = c(l_B - l_A) \quad (2)$$

$$\dot{l}_B = -l_A. \quad (3)$$

Iz podatkov, ki smo jih dobili z integriranjem enačb za $c = 2$ z začetnimi vrednostmi $l_A(0) = 10$ in $l_B(0) =$



Slika 1. Sistem povezanih posod

210, je LAGRANGE odkril naslednje enačbe:

$$l_B = 210 - l_A$$

$$\dot{l}_A = 420 - 4l_A$$

$$\dot{l}_A = -420 + 4l_B$$

$$\dot{l}_B = -420 + 4l_A$$

$$\dot{l}_B = 420 - 4l_B$$

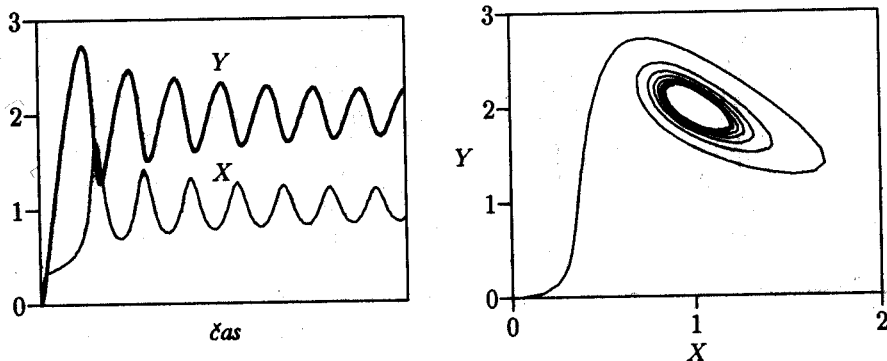
$$\dot{l}_B = -l_A.$$

Prva enačba izraža zakon o ohranitvi volumna tekočine, zadnja pa je ekvivalentna enačbi (3). Drugo in tretjo enačbo dobimo tako, da upoštevamo zakon o ohranitvi volumna ter ustrezno vrednost parametra c v enačbi (2). Preostali dve enačbi, ki ju je LAGRANGE odkril, lahko dobimo če upoštevamo zadnjo enačbo v drugi in tretji.

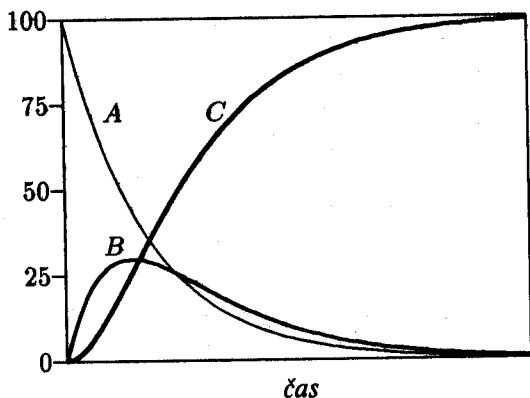
4.2 Preprosta kemična reakcija

Na začetku preproste kemične reakcije imamo neko koncentracijo A_0 nestabilne snovi A . Snov A s hitrostjo k_1 prehaja v snov B , ta pa s hitrostjo k_2 v snov C , ki je stabilna in ne razpada naprej.

Model preproste kemične reakcije opisuje časovni potek koncentracij snovi A , B in C in se da zapisati



Slika 3. Nihanje koncentracij snovi X in Y



Slika 2. Časovni potek koncentracij snovi A, B in C

kot sistem treh linearnih diferencialnih enačb ([1], str. 39):

$$\dot{A} = -k_1 A \quad (4)$$

$$\dot{B} = k_1 A - k_2 B \quad (5)$$

$$\dot{C} = k_2 B. \quad (6)$$

Začetni pogoj je $A(0) = A_0$, $B(0) = C(0) = 0$. Pri integriranju diferencialnih enačb smo za začetno koncentracijo vzeli $A_0 = 100$ ter hitrosti reakcij $k_1 = 2$ in $k_2 = 3$. Na podlagi dobljenega časovnega poteka spremenljivk A, B in C (slika 2) je LAGRANGE odkril štirinajst enačb:

$$\dot{A} = -2A$$

$$\dot{B} = 2A - 3B$$

$$\dot{C} = 3B$$

$$C = 100 - A - B$$

$$\dot{C} = -\dot{A} - \dot{B}$$

$$\dot{A} = -200 + 2B + 2C$$

$$\dot{B} = -300 + 5A + 3C$$

$$\dot{B} = 200 - 5B - 2C$$

$$\dot{B} = -3B - \dot{A}$$

$$\dot{B} = -300 + 3C - 2.5\dot{A}$$

$$\dot{C} = 300 - 3A - 3C$$

$$\dot{C} = 2A - \dot{B}$$

$$\dot{C} = 300 - 3C + 1.5\dot{A}$$

$$\dot{C} = 120 - 1.2C - 0.6\dot{B}.$$

Prve tri enačbe so ekvivalentne enačbam modela (4)–(6) za ustrezne vrednosti parametrov. Četrta izraža zakon o ohranitvi mase ($A + B + C = \text{const}$), peto pa dobimo z odvajanjem obeh strani četrte enačbe. Preostale enačbe, ki jih je odkril LAGRANGE, so redundantne, ker jih lahko dobimo z linearnimi kombinacijami prvih petih enačb.

4.3 Oscilatorna kemična reakcija

Bruselator ([1], str. 175–177) je kemični reaktor, v katerem koncentraciji dveh snovi X in Y nihata okrog nekega ravnovesnega stanja, kot je razvidno iz slike 3. Spreminjanje koncentracij snovi lahko opišemo z naslednjima nelinearnima enačbama:

$$\dot{X} = A - (B + 1)X + X^2 Y \quad (7)$$

$$\dot{Y} = BX - X^2 Y. \quad (8)$$

Pri integriranju diferencialnih enačb smo vzeli začetni koncentraciji $X(0) = Y(0) = 0$ ter parametra $A = 1$ in $B = 2$. Iz dobljenega obnašanja je LAGRANGE odkril naslednje enačbe:

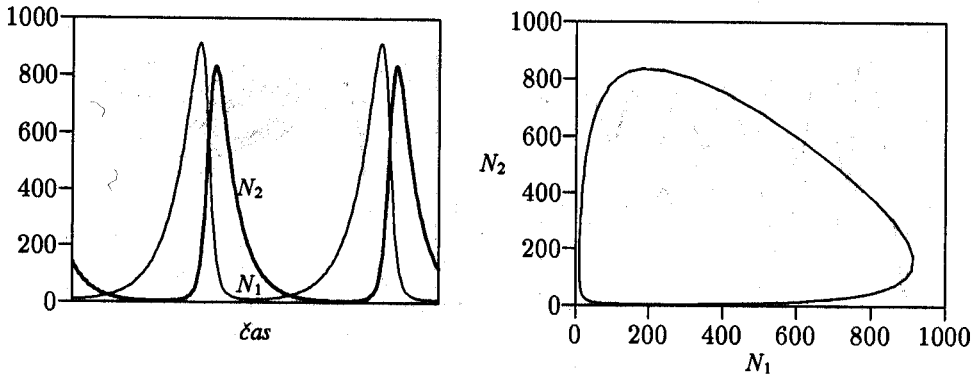
$$\dot{Y} = 1 - X - \dot{X}$$

$$X^2 Y = -1 + 3X + \dot{X}$$

$$X^2 Y = 2X - \dot{Y}$$

$$X^2 Y = 2 - 2\dot{X} - 3\dot{Y}.$$

Druga in tretja enačba sta ekvivalentni enačbam modela (7) in (8), preostali dve dobimo z linearnimi kombinacijami druge in tretje enačbe.



Slika 4. Nihanje velikosti populacij volkov in zajcev

4.4 Populacijska dinamika

Predstavljajmo si sožitje populacij volkov in zajcev v istem okolju. Predpostavimo še da so zajci edina hrana volkovom, zajci pa imajo neomejene zaloge korenja. Če je populacija zajcev velika, populacija volkov narašča, ker imajo veliko hrane. Naraščanje traja toliko časa, dokler ne začne zmanjkovati hrane, torej zajcev. Potem se število volkov začne zmanjševati. To pa takoj izkoristijo zajci, katerih število začne naraščati (slika 4).

Nihanje velikosti populacij volkov in zajcev lahko opišemo z modelom Volterra-Lotka ([1], str. 145):

$$\begin{aligned}\dot{N}_1 &= k_1 N_1 - s N_2 N_1 \\ \dot{N}_2 &= s N_1 N_2 - k_2 N_2,\end{aligned}$$

pri čemer sta N_2 in N_1 ustrezni velikosti populacij volkov in zajcev. Pri integriranju enačb smo vzeli začetne velikosti $N_1(0) = 10$ in $N_2(0) = 140$ ter parametre $k_1 = 1.6$, $s = 0.01$ in $k_2 = 2$. LAGRANGE je na podlagi časovnega poteka spremenljivk N_1 in N_2 (slika 4) odkril enačbi, ki sta ekvivalentni enačbam modela:

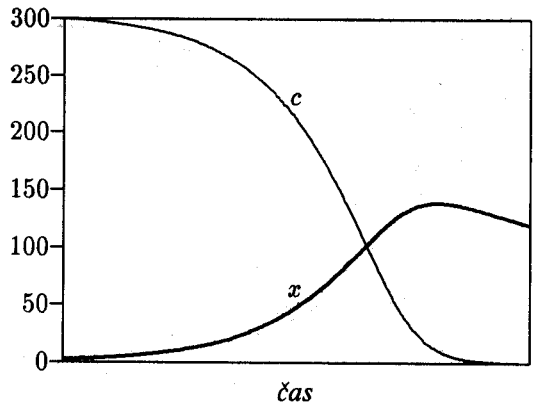
$$\begin{aligned}N_1 N_2 &= 160 N_1 - 100 \dot{N}_1 \\ N_1 N_2 &= 200 N_2 - 100 \dot{N}_2.\end{aligned}$$

4.5 Monod

V posodi imamo neko koncentracijo hranilne raztopine c in koncentracijo bio mase x . Posoda je izolirana od okolja, tako da ni dotoka hrane in bio mase od zunaj, ter ima konstantno temperaturo. Velika začetna koncentracija hranilne raztopine povzroči razvoj bio mase. Koncentracija bio mase narašča, dokler ne zmanjka hrane, potem pa začne padati (slika 5). Spreminjanje koncentracij bio mase in hranilne raztopine opišemo z naslednjima diferencialnima enačbama ([11], str. 301):

$$\dot{c} = -\frac{\mu_{max}}{y} \frac{c}{c+k_s} x \quad (9)$$

$$\dot{x} = \left(\mu_{max} \frac{c}{c+k_s} - k_d\right) x. \quad (10)$$



Slika 5. Časovni potek koncentracij bio mase in hranilne raztopine

Zgornje enačbe smo integrirali pri vrednostih parametrov $\mu_{max} = 0.1$, $k_d = 0.01$, $k_s = 100$ in $y = 0.6$ z začetnima koncentracijama $c(0) = 300$ in $x(0) = 2$ ter časovnim korakom $h = 0.1$. Časovni korak je večji kot v drugih poskusih zaradi počasne dinamike spreminjanja koncentracij bio mase in hranilne raztopine. LAGRANGE je odkril naslednje štiri enačbe:

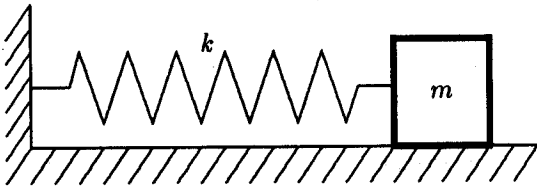
$$\begin{aligned}\dot{x} &= -0.01x - 0.6\dot{c} \\ c\dot{c} &= -100\dot{c} - 0.167cx \\ c\dot{x} &= 60\dot{c} + 0.09cx \\ c\dot{x} &= 6\dot{c} - 0.54c\dot{c}.\end{aligned}$$

Druga enačba je ekvivalentna enačbi (9) za ustrezne vrednosti parametrov. Upošteva prve dve enačbi dobimo enačbo, ki je ekvivalentna enačbi (10). Torej sta prvi dve enačbi, ki ju je LAGRANGE odkril, ekvivalentni modelu, preostali dve pa sta redundantni, ker se ju da dobiti s kombiniranjem prvih dveh enačb.

4.6 Vzmetno nihalo

Telo z maso m je pritrjeno na prožno vzmet s prožnostno konstanto k (slika 6). V ravnovesni legi vzmet ni ne raztegnjena ne skrčena. Če telo izmaknemo iz ravnovesne lege za x_0 in nato spustimo (ob tem zanemarimo trenje med telesom in podlago), začne le to nihati z amplitudo x_0 . Spreminjanje odmika telesa od ravnovesne lege x lahko opišemo z naslednjo linearno diferencialno enačbo drugega reda ([12], str. 44):

$$\ddot{x} = -\frac{k}{m}x \quad (11)$$



Slika 6. Vzmetno nihalo

Začetna pogoja diferencialne enačbe sta $x(0) = x_0$ in $\dot{x}(0) = 0$. LAGRANGE je iz podatkov, dobljenih z integriranjem diferencialne enačbe (11) pri $x(0) = 10$, $k = 2$ in $m = 1$, odkril naslednje diferencialne enačbe:

$$\begin{aligned} \ddot{x} &= -2x \\ \dot{x}^2 &= 4x^2 \\ \ddot{x}^2 &= 400 - 2\dot{x}^2 \\ \dot{x}^2 &= 200 - 2x^2 \\ \dot{x}^2 &= 200 + x\ddot{x} \end{aligned}$$

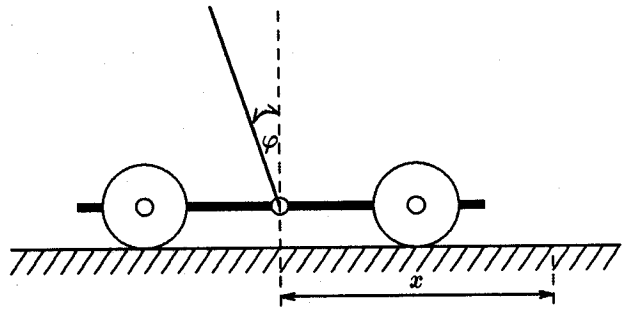
Prva enačba je enaka enačbi (11) za ustrezno vrednost parametra k . Drugo enačbo dobimo s kvadriranjem obeh strani prve. Preostale tri enačbe lahko izpeljemo, če upoštevamo analitično rešitev enačbe (11): $x = x_0 \cos \sqrt{\frac{k}{m}}t$, ki te enačbe pripelje do oblike $\sin^2 \sqrt{\frac{k}{m}}t + \cos^2 \sqrt{\frac{k}{m}}t = 1$. Zadnjo enačbo lahko interpretiramo kot zakon o ohranitvi energije, pri čemer pomeni člen $x\ddot{x}$ potencialno, člen \dot{x}^2 pa kinetično energijo telesa.

4.7 Invertirano nihalo

Vodenje invertiranega nihala je standardni problem za preskus različnih metod vodenja dinamičnih sistemov. Invertirano nihalo je sestavljeno iz vozička, na katerem je vrtljivo vpeta palica, ki je gibljiva v vertikalni ravnini, ki gre skozi tirnice (slika 7). Če sta M in m masi vozička in palice, l dolžina palice, F sila s katero delujemo na voziček, x položaj vozička glede na izhodišče in φ kot nagiba palice, potem dinamiko sistema palice na vozičku opišemo z naslednjima diferencialnima enačbama [6]:

$$(M + m)\ddot{x} + \frac{1}{2}ml(\ddot{\varphi} \cos \varphi - \dot{\varphi}^2 \sin \varphi) = F \quad (12)$$

$$\ddot{x} \cos \varphi + \frac{2}{3}l\ddot{\varphi} = g \sin \varphi \quad (13)$$



Slika 7. Invertirano nihalo

Pri integriranju enačb smo vzeli za $M = 1[kg]$, $m = 0.1[kg]$, $l = 1[m]$ ter $F = 7.5[N]$. Integracijski korak je bil $h = 0.01$. Iz začetnega stanja $x(0) = \dot{x}(0) = \varphi(0) = 0$ ter $\dot{\varphi}(0) = 3\pi/16$ je po $N = 125$ časovnih korakih palica padla ($\varphi = \pi/2$). LAGRANGE je iz dobljenega časovnega poteka spremenljivk x , φ , \dot{x} in $\dot{\varphi}$ odkril naslednje štiri enačbe:

$$\begin{aligned} \cos^2 \varphi &= 1 - \sin^2 \varphi \\ \ddot{x} \cos \varphi &= -0.67\ddot{\varphi} + 9.81 \sin \varphi \\ \dot{\varphi}^2 \sin \varphi &= -150 + 22\dot{x} + \dot{\varphi} \cos \varphi \\ \ddot{x} \cos \varphi \sin \varphi &= 9.81 - 0.67\ddot{\varphi} \sin \varphi - 9.81 \cos^2 \varphi \end{aligned}$$

Prva enačba izraža znano zvezo med sinusom in kosinusom kota ($\sin^2 \varphi + \cos^2 \varphi = 1$). Druga in tretja sta ekvivalentni enačbama (12) in (13). Zadnjo enačbo dobimo, če zmnožimo drugo s $\sin \varphi$ in upoštevamo zvezo $\sin^2 \varphi + \cos^2 \varphi = 1$.

5 Razširitve programa LAGRANGE

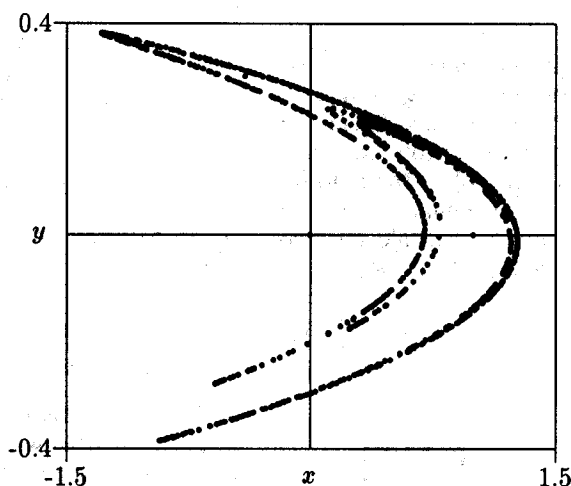
5.1 Modeli dinamičnih sistemov v diskretnem času

Matematični modeli dinamičnih sistemov v diskretnem času slonijo na diferencialnih enačbah. Zelo preprosta razširitev osnovne verzije programa LAGRANGE omogoča odkrivanje diferencialnih enačb. V prvem koraku algoritma namesto z odvajanjem uvajamo nove spremenljivke s časovnim zamikom osnovnih spremenljivk po predpisu: $x'_i = x_{i+1}$, pri čemer sta x_{i+1} ter x'_i ustrezni vrednosti spremenljivke $X(t)$ in njenega časovnega zamika $X'(t) = X(t+h)$ v točki $t = t_0 + (i-1)h$. Drugi in tretji korak algoritma sta nespremenjena. Izhod je množica enačb oblike:

$$\begin{aligned} F_i (X_1(t), \dots, X_n(t), \\ X_1(t+h), \dots, X_n(t+h), \dots, \\ X_1(t+oh), \dots, X_n(t+oh)) = 0, \quad (14) \end{aligned}$$

pri čemer so X_1, \dots, X_n spremenljivke dinamičnega sistema, o red dinamičnega sistema, ter F_i polinomske funkcije.

Tako prilagojeno verzijo programa LAGRANGE smo preskusili na problemu rekonstrukcije Henonovih



Slika 8. Henonov atraktor

diferenčnih enačb:

$$x' = 1 - ax^2 + y \quad (15)$$

$$y' = bx \quad (16)$$

Simulirali smo zgornje enačbe za vrednosti parametrov $a = 1.4$ in $b = 0.3$ ter začetno stanje $x_0 = y_0 = 0$ v $N = 1000$ časovnih točkah. Na podlagi tako dobljenih podatkov (slika 8) je LAGRANGE odkril naslednje enačbe:

$$y' = 0.3 \cdot x$$

$$y'^2 = 0.09 \cdot x^2$$

$$x^2 = 0.71429 + 0.71429 \cdot y - 0.71429 \cdot x'$$

$$xy' = 0.21429 + 0.21429 \cdot y - 0.21429 \cdot x'$$

$$y'^2 = 0.06429 + 0.06429 \cdot y - 0.06429 \cdot x'$$

Prva in tretja enačba, ki ju je odkril LAGRANGE, sta ekvivalentni enačbam (15) in (16) za ustrezne vrednosti parametrov. Drugo enačbo dobimo s kvadriranjem prve, zadnji dve pa dobimo če upoštevamo prvo enačbo v tretji enkrat oziroma dvakrat, ustrezno.

5.2 Eksplicitno izražanje odvodov najvišje stopnje

V tej verziji programa LAGRANGE smo se omejili na iskanje enačb, ki so linearne glede na časovne odvode najvišje stopnje o . Sistem enačb te oblike omogoča eksplicitno izražanje odvodov najvišje stopnje, tako da modele, ki jih odkrije ta verzija programa LAGRANGE, lahko zapišemo v obliki:

$$X_i^{(o)} = F_i(X_1, \dots, X_n, \dot{X}_1, \dots, \dot{X}_n, \dots, X_1^{(o-1)}, \dots, X_n^{(o-1)}), \quad (17)$$

pri čemer so X_1, \dots, X_n spremenljivke dinamičnega sistema, o red dinamičnega sistema (stopnja najvišjega

časovnega odvoda, ki se pojavlja v enačbah dinamike) ter F_1, \dots, F_n racionalne funkcije. Motivacija za razvoj te različice programa LAGRANGE je možnost preprostejše simulacije dobljenih enačb. Dodatna pridobitev je zmanjševanje preiskovalnega prostora.

Osnovni algoritem smo spremenili tako kot je opisano v nadaljevanju. Pri uvajanju novih spremenljivk z množenjem, v drugem koraku algoritma LAGRANGE, uvajamo samo spremenljivke, ki vsebujejo največ en časovni odvod najvišje stopnje. Če je $o = 1$ in $d = 2$, uvedemo spremenljivke $X_1^2, X_1X_2, X_1\dot{X}_1, X_1\dot{X}_2, X_2^2, X_2\dot{X}_1$ ter $X_2\dot{X}_2$. Spremenljivk $X_1^2, X_1\dot{X}_2$ in \dot{X}_2^2 , ki jih uvede osnovna verzija programa LAGRANGE, v tej verziji ne uvajamo, ker enačba oblike (17) ne more vsebovati nobene izmed teh treh spremenljivk.

V tretjem koraku algoritma LAGRANGE obravnavamo samo enačbe, v katerih vsaj ena spremenljivka vsebuje časovni odvod najvišje stopnje. Če se vrnemo k primeru iz prejšnjega odstavka, bi bil primer enačbe, ki je ne obravnavamo $X_2^2 = c_1X_1 + c_2X_2$, ker ne vsebuje nobenega časovnega odvoda najvišje stopnje.

S tako prilagojeno verzijo program LAGRANGE smo ponovili vse poskuse iz četrtega poglavja. Kot je razvidno iz tabele 4, so časi izvajanja krajši, kar je posledica zmanjševanja števila novih spremenljivk, uvedenih z množenjem, ter števila obravnavanih enačb. V vseh primerih so bili odkriti modeli povsem ekvivalentni originalnim.

5.3 GoldHorn

Program GoldHorn [13] je razširitev programa LAGRANGE v smeri obravnave šumnih podatkov iz realnih meritev. Odpravlja dve pglavitni slabosti programa LAGRANGE: numerično računanje časovnih odvodov najvišje stopnje in problem izbire odvisne spremenljivke pri linearni regresiji [13]. Dodatno je v programu GoldHorn uvedena možnost predprocesiranja vhodnih podatkov z različnimi digitalnimi filtri, da bi se zmanjšale relativne

Tabela 4. Pregled poskusov z verzijo programa LAGRANGE, ki eksplicitno izraža odvode najvišje stopnje

Domena	n	N	o	d	r	t_R	t_S	št. enačb	čas[s]
Utube	2	1000	1	1	3	0.01	0.01	5	0.68
Chem	2	1000	1	1	3	0.01	0.01	13	0.99
Brus	2	1000	1	3	3	0.01	0.01	4	4.23
Pop	2	1000	1	2	3	0.01	0.01	2	1.14
Monod	2	1000	1	2	3	0.01	0.0001	4	1.18
Spring	1	1000	2	2	3	0.01	0.01	2	0.57
Cart	2	125	2	3	3	0.01	0.0001	3	106.36

napake meritev.

Pri preiskovanju prostora enačb GoldHorn eksplicitno izraža časovne odvode najvišje stopnje kot (racionalne) funkcije spremenljivk dinamičnega sistema in njihovih nižjih odvodov. Pri tem se izogne numeričnemu računanju časovnih odvodov najvišje stopnje tako, da ocenjuje signifikantnost in koeficiente enačbe oblike (17) z integriranjem. Ker pa v tem primeru obravnavane enačbe niso več linearne, uporablja pri tem postopek nelinearne numerične optimizacije [17] v prostoru parametrov.

Program GoldHorn je bil preskušen na domenah iz četrtega poglavja, razen na domenah populacijske dinamike in invertiranega nihala, kot tudi na dveh realnih domenah [13]. Pri poskusih na sintetičnih domenah je bil simuliranemu obnašanju kasneje dodan normalno porazdeljen šum. GoldHorn je odkril ustrezne modele dinamičnih sistemov, tudi takrat, ko je bil v vhodnih podatkih razmeroma visok odstotek šuma.

6 Zaključki in nadaljnje delo

Bistveni prispevek programa LAGRANGE je razširitev področja uporabe programov za avtomatsko odkrivanje zakonitosti v smeri modeliranja dinamičnih sistemov. Metode, uporabljene v drugih programih, smo prilagodili posebnostim, ki se kažejo pri odkrivanju zakonitosti v dinamični domeni. Tako je LAGRANGE zmožen odkrivanja množice zakonitosti nad več spremenljivkami brez izvajanja dodatnih poskusov, medtem ko drugi programi za avtomatsko odkrivanje zakonitosti običajno zahtevajo eksperimentalno kontrolo nad posameznimi spremenljivkami modeliranega sistema.

LAGRANGE smo preskusili na problemih rekonstrukcije modelov dinamičnih sistemov iz dinamike tekočin, kemičnih reakcij, populacijske ter mehanske dinamike. Med njimi so tudi nelinearni modeli oscilatornih procesov ter dinamični model inverznega nihala, ki je standardni problem za testiranje različnih metod vodenja dinamičnih sistemov. V vseh domenah je LAGRANGE uspešno rekonstruiral enačbe modela, poleg njih pa še enačbe, ki jih lahko razvrstimo v štiri skupine:

- zakon o ohranjanju volumna oziroma mase (Utube, Chem) ter energije (Spring)
- trigonometrijska identiteta $\sin^2 \varphi + \cos^2 \varphi = 1$ (Cart)

- enačbe, ki izvirajo iz analitične rešitve (Spring)
- redundantne enačbe, ki jih lahko izpeljemo iz enačb modela (vse domene razen Pop)

Opisali smo tudi razširitev osnovne verzije programa LAGRANGE, ki omogoča obravnavo dinamičnih sistemov v diskretnem času. Posebno pomembna razširitev je program GoldHorn [13], ki omogoča obravnavo šumnih podatkov iz realnih meritev. GoldHorn odpira možnosti za uporabo metod avtomatskega odkrivanja zakonitosti za modeliranje realnih dinamičnih sistemov.

Program LAGRANGE bi lahko preprosto razširili z možnostjo hkratnega obravnavanja več različnih obnašanj dinamičnega sistema. Zanimive bi bile tudi razširitve v smeri predlaganja novih poskusov (pridobivanja meritev o novih obnašanjih dinamičnega sistema), po analogiji z obstoječimi pristopi k avtomatskemu odkrivanju zakonitosti.

Programu LAGRANGE bi lahko dodali tudi različne dodatne transformacije za uvajanje novih spremenljivk. V programu GoldHorn dodajamo lahko nove spremenljivke z operatorji $|x|$, $\ln x$, $\operatorname{sgn} x$ ter $\exp x$. Uvajanje novih transformacij bi povzročilo povečanje števila novih spremenljivk ter s tem tudi kompleksnosti prostora mogočih enačb. Povečanje bi lahko omilili z uporabo heuristik za uvajanje novih spremenljivk ter za preiskovanje prostora mogočih enačb.

Alternativna metoda za avtomatsko odkrivanje zakonitosti v dinamični domeni bi bila uporaba genetskega programiranja za preiskovanje prostora enačb. Džeroski in Petrovski [7] sta že naredila poskuse v tej smeri. Metoda omogoča tudi uporabo znanja s konkretnega področja uporabe. Znanje ima obliko podizrazov, ki nastopajo v enačbah dinamike (podizraz $c/(c+k_s)$ v enačbah iz domene Monod). Uvajanje predznanja v proces preiskovanja prostora mogočih enačb je kompromis med metodami identifikacije dinamičnih sistemov, ki predvidevajo vnaprej znano obliko enačb, in pristopom popolnoma neznane strukture, ki ga uporablja program LAGRANGE. Seveda je uporaba predznanja lahko tudi ena od smernic za nadaljnji razvoj programa LAGRANGE.

7 Literatura

- [1] A. Babloyantz. *Molecules, Dynamics, and Life*. John Wiley & Sons, New York, 1986
- [2] M. W. Bielecki. Machine discovery approach to dynamic systems in the real laboratory. In J. M. Żytkow, editor, *Proc. of the ML92 Workshop on Machine Discovery*, pages 58–62, 1992
- [3] Z. Bohte. *Numerične metode*. Društvo matematikov, fizikov in astronomov Slovenije, Ljubljana, 1991
- [4] I. Bratko. *Prolog Programming for Artificial Intelligence*. Addison-Wesley, Wokingham, 2nd edition, 1990
- [5] J. Carbonell and P. Langley. Learning, machine. In S. C. Shapiro and D. Eckroth, editors, *Encyclopedia of Artificial Intelligence*, volume 1, pages 464–488, New York, 1987. John Wiley & Sons
- [6] S. Džeroski. Upravljanje inverznega nihala. Diplomaska naloga, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, 1989
- [7] S. Džeroski and I. Petrovski. Discovering dynamics with genetic programming. In *Proc. Seventh European Conference on Machine Learning*, pages 347–350, Berlin, 1994. Springer
- [8] S. Džeroski and Lj. Todorovski. Discovering dynamics. In *Proc. Tenth International Conference on Machine Learning*, pages 97–103, San Mateo, CA, 1993. Morgan Kaufmann
- [9] S. Džeroski and Lj. Todorovski. Discovering dynamics: from inductive logic programming to machine discovery. *Journal of Intelligent Information Systems*, Special issue on Knowledge Discovery in Databases, 1994. In press.
- [10] B. Falkenhainer and R. Michalski. Integrating quantitative and qualitative discovery in the ABACUS system. In Y. Kodratoff and R. Michalski, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 153–190, San Mateo, CA, 1990. Morgan Kaufmann
- [11] S.E. Jorgensen and E. Johnsen. *Principles of Environmental Science and Technology*. Elsevier, Amsterdam, 1989
- [12] R. Kladnik. *Visokošolska fizika*. Državna založba Slovenije, 1989. Del 1: Mehanski in toplotni pojavi
- [13] V. Križman. Obravnavanje šumnih podatkov pri avtomatskem modeliranju dinamičnih sistemov. Magistrsko delo, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, 1994
- [14] B. Kuipers. Qualitative simulation. *Artificial Intelligence*, 29(3):289–338, 1986
- [15] P. Langley, H.A. Simon, and G.L. Bradshaw. Heuristics for empirical discovery. In L. Bolc, editor, *Computational Models of Learning*, Berlin, 1987. Springer
- [16] P. Langley and J. Żytkow. Data-driven approaches to empirical discovery. *Artificial Intelligence*, 40:283–312, 1989
- [17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes*. Cambridge University Press, Cambridge, MA, 1986
- [18] Lj. Todorovski. Modeliranje dinamičnih sistemov z avtomatskim odkrivanjem zakonitosti. Diplomaska naloga, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, 1993
- [19] W. Volk. *Applied Statistics for Engineers*. McGraw-Hill, New York, 1958
- [20] R. Zembowitz and J. Żytkow. Discovery of equations: experimental evaluation of convergence. In *Proc. Tenth National Conference on Artificial Intelligence*, San Mateo, CA, 1992. Morgan Kaufmann
- [21] J. Żytkow and J. Zhu. Application of empirical discovery in knowledge acquisition. In *Proc. Fifth European Working Session on Learning*, pages 101–117, Berlin, 1991. Springer

Ljupčo Todorovski je rojen 29. julija 1969, v Bitoli, Makedonija. Leta 1993 je diplomiral na smeri računalništvo in informatika na Fakulteti za elektrotehniko in računalništvo v Ljubljani. Študent je magistrskega študija računalništva in informatike na isti fakulteti. Njegove raziskave potekajo na področju avtomatskega učenja ter uporabe avtomatskega odkrivanja zakonitosti za modeliranje dinamičnih sistemov.

Sašo Džeroski je rojen 31. maja 1968, v Ohridu, Makedonija. Leta 1989 oziroma 1991 je diplomiral oziroma magistriral na smeri računalništvo in informatika na Fakulteti za elektrotehniko in računalništvo v Ljubljani. Od leta 1989 je mladi raziskovalec v Laboratoriju za umetno inteligenco na Inštitutu Jožef Stefan v Ljubljani. Njegove raziskave potekajo na področju avtomatskega učenja, vključno z teoretičnimi in praktičnimi vidiki induktivnega logičnega programiranja ter uporabo avtomatskega učenja za modeliranje in kontrolo dinamičnih sistemov.