

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

Ing. Zdeněk Kratochvíl

AUTOMATICKÉ VYTVÁŘENÍ PRAVIDEL PRO ZNALOSTNÍ
BÁZE METODAMI TZV. SOFT-COMPUTING

AUTOMATIC GENERATION OF RULES FOR KNOWLEDGE
BASES BY METHODS FROM SOFT-COMPUTING AREA

Teze doktorské disertační práce

PhD Thesis

Obor: Kybernetika a informatika

Školitel: Doc. Ing. Pavel Jura, CSc.

Oponenti: Prof. Dr. Ing. Miroslav Pokorný

Doc. Ing. Vladimír Řeřucha, CSc.

Datum obhajoby: 29. června 2001

© Zdeněk Kratochvíl, 2001
ISBN 80-214-1926-1
ISSN 1213-4198

OBSAH

1. ÚVOD	5
2. CÍLE DISERTAČNÍ PRÁCE	6
3. SHRNUÍ DOSAVADNÍCH PŘÍSTUPŮ K ŘEŠENÍ ÚLOH ZALOŽENÝCH NA ANN	7
3.1 SHRNUÍ VLASTNOSTÍ ANN JAKO NÁSTROJE PRO AUTOMATIZOVANÉ ZÍSKÁVÁNÍ ZNALOSTÍ	7
3.2 SHRNUÍ VLASTNOSTÍ ALGORITMŮ PRO EXTRAKCI PRAVIDEL Z ANN	8
3.3 POŽADAVKY NA ALGORITMY PRO EXTRAKCI PRAVIDEL Z ANN	9
4. EXTRAKCE PRAVIDEL Z ANN PRO ÚLOHY SE SYMBOLICKOU VÝSTUPNÍ TŘÍDOU	11
4.1 VÝLEDNÁ STRUKTURA NEURONOVÉ SÍTĚ	11
4.2 SHRNUÍ VLASTNOSTÍ NAVRŽENÉHO ALGORITMU ZÍSKÁVÁNÍ BÁZE PRAVIDEL POMOCÍ ANN	13
4.3 SHRNUÍ EXPERIMENTÁLNÍCH VÝLEDKŮ	14
5. EXTRAKCE PRAVIDEL Z ANN PRO ÚLOHY SE SPOJITOU VÝSTUPNÍ VELIČINOU	16
5.1 VÝLEDNÁ STRUKTURA NEURONOVÉ SÍTĚ	16
5.2 EXPERIMENTÁLNÍ VYHODNOCENÍ	17
5.3 SHRNUÍ	18
6. GENETICKÉ ALGORITMY PRO VYTVÁŘENÍ ZNALOSTNÍCH BÁZÍ FUZZY REGULÁTORŮ	19
6.1 PRINCIP GENETICKÝCH ALGORITMŮ	19
6.2 GA PRO GENEROVÁNÍ ZNALOSTNÍ BÁZE FUZZY REGULÁTORU	20
6.3 EXPERIMENTÁLNÍ VYHODNOCENÍ	21
6.4 SHRNUÍ	23
7. ZÁVĚR	24
8. CONCLUSIONS AND PROSPECTIVES	26
9. LITERATURA	28
10. PUBLIKOVANÉ PRÁCE	31
10.1 PUBLIKOVANÉ PRÁCE VZTAHUJÍCÍ SE K TÉMATU DISERTAČNÍ PRÁCE:	31
10.2 OSTATNÍ PUBLIKOVANÉ PRÁCE:	31
11. ŽIVOTOPIS AUTORA	32

1. ÚVOD

Velkou výhodou uchovávání znalostí ve formě báze pravidel, rozhodovacích stromů a podobně, které se používá v expertních systémech nebo symbolických systémech umělé inteligence je, že takto uložená znalost má poměrně průhlednou strukturu a je velice snadno přenositelná mezi celou řadou systémů. Tato struktura také umožňuje příslušnému systému využívajícímu tuto znalost poskytovat uživateli vysvětlení postupu, kterým jsou vytvářeny jeho výstupy nebo závěry. Při realizaci těchto systémů však často narážíme na problém jak získat dostatečně přesný popis dané problematiky. Například u expertních systémů je nejobtížnější, časově nejnáročnější a také nejdražší fází při jejich vytváření získávání a zejména pak ladění báze pravidel.

Na druhé straně existují metody, které umožňují automatické získávání nebo ladění znalostí. Tato znalost je však často uložena v naprosto nečitelné podobě, která nedovoluje ověřit, zda je znalost úplná - to znamená, zda popisuje celou oblast řešené problematiky. Není-li možné ověřit úplnost získané znalosti, může při činnosti takového systému dojít k situaci, že bude dávat chybné výsledky, jejichž důsledkem mohou být vážné havárie. To vede k malému rozšíření jinak velmi výkonných systémů. Typickým představitelem této kategorie jsou systémy založené na umělých neuronových sítích (ANN - Artificial Neural Network). Neuronové sítě již v řadě aplikací prokázaly, že jsou schopny velice úspěšně řešit náročné problémy, ale jejich širšímu uplatnění brání právě velká neprůhlednost získaných znalostí, které jsou v ní zakódovány v podobě:

- architektury neuronové sítě (počet vrstev v ANN, počet a typ neuronů v jednotlivých vrstvách, způsob propojení neuronů apod.),
- přenosových funkcí neuronů,
- množiny numerických parametrů, tzv. vah.

Podobná situace jako u výše zmiňovaných expertních systémů se objevuje také při používání fuzzy regulátorů. Jejich aplikace zpočátku vycházely zejména z možnosti využití znalosti získané lidskou obsluhou (fuzzy regulátor pracuje na základě báze pravidel naformulované pomocí lingvistických pojmů). I zde se však objevovaly komplikace, protože kvalitní regulace vyžaduje kromě správně sestavené báze pravidel ještě vhodné nastavení rozložení a tvarů fuzzy množin reprezentujících jednotlivé lingvistické hodnoty. Později byla také snaha vytvářet fuzzy regulátory i v případě, kdy nebyla k dispozici prvotní znalost operátora. V tomto okamžiku však vyvstaly problémy s metodologií návrhu těchto regulátorů, a to už u poměrně jednoduchých soustav popsanych přenosovou funkcí nízkých řádů. U systémů vyšších řádů obsahujících výraznější nelinearity je

pak ruční návrh fuzzy regulátorů velice časově náročný a komplikovaný. Přesto však zůstává velikou výhodou potenciál fuzzy regulátorů jako nelineárních řídicích prvků, jejichž činnost je díky znalostní bázi založené na množině pravidel možné poměrně snadno analyzovat a nahlédnout tak do zákonitostí chování systému.

Proto se výzkum zaměřuje na postupy dovolující vytvářet systémy, které by umožnily skloubit výhody obou výše uvedených skupin. Tyto systémy by měly dostatečně průhlednou strukturu používaných znalostí (podobně jako expertní systémy nebo symbolické systémy umělé inteligence) nebo by obsahovaly nástroje umožňující vysvětlit postup vytváření závěrů nebo výsledků a daly tak obsluze možnost zkontrolovat jejich správnost a provést případnou korekci. Na druhé straně by umožňovaly automatické získávání a doladování svých znalostí, čímž by se urychlil, zjednodušil a v neposlední řadě také zlevnil jejich vývoj. To by mohlo vést k vytváření velmi efektivních a výkonných systémů, které by našly uplatnění v nejrůznějších oblastech.

Předkládaná disertační práce je zaměřena na prozkoumání možností, návrh a realizace metod pro automatizované vytváření pravidel pro znalostní báze s využitím nástrojů z oblasti soft-computing a to konkrétně genetických algoritmů a neuronových sítí.

2. CÍLE DISERTAČNÍ PRÁCE

C 1: Základním cílem disertační práce bylo shromáždit a prostudovat dostupnou literaturu z oblasti umělých neuronových sítí a jejich využití pro generování bází pravidel (extrakce pravidel z ANN) a z oblasti genetických algoritmů.

O problematice extrakce pravidel z ANN se podařilo shromáždit několik zajímavých publikací z nichž nejpřínosnější byla „A Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks“ (viz. [2]), kterou publikovali R. Andrews, J. Diedrich a A. B. Tickle. Tato práce obsahuje přehled a porovnání významných prací z dané oblasti.

U genetických algoritmů byla základem převážně literatura obecně popisující tuto metodu resp. její aplikaci na demonstračních úlohách (např. [9]).

Na základě poznatků získaných v této etapě byly stanoveny další cíle:

C 2: Na základě poznatků získaných studiem literatury navrhnout vhodnou strukturu ANN, metodu jejího učení a následné extrakce pravidel tak, aby byly odstraněny základní nedostatky doposud používaných algoritmů (viz. kapitola 3.2. Shrnutí vlastností algoritmů pro extrakci

pravidel z ANN a 3.3. Požadavky na algoritmy pro extrakci pravidel z ANN).

- C 3: Realizovat navrženou neuronovou síť a ověřit její funkčnost na úlohách z oblasti klasifikace vzorů.
- C 4: Prezentovat možné rozšíření metody pro extrakci pravidel z ANN z klasifikačních úloh na úlohy pracující se spojitou výstupní veličinou (jeden z požadavků definovaných v kapitole 3.3. Požadavky na algoritmy pro extrakci pravidel z ANN) včetně realizace a základního ověření funkčnosti.
- C 5: Navrhnout a implementovat metodu pro vytváření a optimalizaci bází fuzzy pravidel založenou na genetických algoritmech.
- C 6: Experimentálně ověřit funkčnost na úlohách vytváření fuzzy regulátorů jak pro řízení lineárních tak i nelineárních soustav.

3. SHRUTÍ DOSAVADNÍCH PŘÍSTUPŮ K ŘEŠENÍ ÚLOH ZALOŽENÝCH NA ANN

3.1 SHRUTÍ VLASTNOSTÍ ANN JAKO NÁSTROJE PRO AUTOMATIZOVANÉ ZÍSKÁVÁNÍ ZNALOSTÍ

Jedním z nástrojů, který umožňuje automatizované získávání znalostí jsou umělé neuronové sítě. Jejich vývoj probíhá již celou řadu let, během nichž vznikla celá řada typů neuronových sítí přizpůsobených nejrůznějším účelům od rozpoznávání vzorů, aproximaci funkcí až po asociativní paměti nebo kompresi dat. S tím také souvisí celá řada úspěšných praktických aplikací v oblastech tak rozmanitých jako je obchod, věda, průmysl a medicína. Toto rozšíření ANN nabízí jasné důkazy jejich schopností v oblasti analýzy dat a získávání znalostí.

Existuje několik charakteristických rysů neuronových sítí, na nichž je založena jejich úspěšnost, a kterými se odlišují od většiny jiných přístupů:

- Prvním z nich je poměrně jednoduchý způsob, kterým ANN získávají informace o dané problematice prostřednictvím trénování. Tento proces je docela odlišný od komplikovanějších procesů znalostního inženýrství používaných u symbolických systémů v umělé inteligenci.
- Druhou charakteristikou je kompaktnost formy, ve které jsou získané informace uloženy v natrénované ANN, a relativní jednoduchost a rychlost, se kterou tyto znalosti mohou být zpřístupněny a použity.
- Třetí charakteristikou je robustnost řešení získaného pomocí ANN za přítomnosti šumu ve vstupních datech.

Kromě těchto charakteristik je nejdůležitější výhodou natrénovaných ANN vysoký stupeň přesnosti výsledků v případech, kdy je řešení ANN použito pro zobecnění na množinu vzorů nevyskytujících se při trénování.

Na druhou stranu však výše uvedené vlastnosti neuronových sítí vedou k jejich určitým nedostatkům. Za jejich nejzávažnější nevýhodu je považována neprůhledná struktura naučené znalosti a z toho plynoucí nemožnost zpětné kontroly výsledků jimi generovaných. Rozšíření možností ANN o tuto zpětnou kontrolu se ukazuje jako velice žádoucí a v některých případech i nezbytně nutné pro další rozvoj aplikací na nich založených. Tento požadavek je obzvláště důležitý v případech, kdy mají být ANN využity v oblastech označovaných za kritické z hlediska bezpečnosti jako je například letectví nebo energetika.

Na základě toho v posledních letech směřuje snaha mnoha vědců na rozšíření algoritmů neuronových sítí o část umožňující vysvětlit ve srozumitelné podobě postup, kterým jsou vytvářena jejich rozhodnutí nebo výstupy. Pro tyto účely jsou vyvíjeny metody pro extrakci pravidel z ANN resp. metody umožňující co nejlépe popsat znalost obsaženou v ANN pomocí báze pravidel. Na tuto znalostní bázi je pak již možné aplikovat propracované algoritmy pro vysvětlování postupu vytváření závěrů, které byly vyvinuté již dříve například pro expertní systémy, jejichž jsou nedílnou součástí.

3.2 SHRNUTÍ VLASTNOSTÍ ALGORITMŮ PRO EXTRAKCI PRAVIDEL Z ANN

Ze studia podkladů vyplynulo, že již byla vytvořena celá řada algoritmů pro extrakci pravidel z ANN, které byly s většími či menšími úspěchy aplikovány na nejrůznější úlohy. Tyto přístupy však zatím většinou vymezují základní metody pro řešení tohoto úkolu. V současné době se úsilí zaměřuje na vývoj systémů, které budou vycházet z poznatků získaných právě na základě těchto základních metod a budou odstraňovat jejich nedostatky, které jsou shrnuty v této kapitole.

Hlavním nedostatkem tzv. dekompozičního přístupu k extrakci pravidel z klasických ANN typu back-propagation, která zabránila širšímu rozvoji těchto metod, je velice obtížná (nebo téměř nemožná) srozumitelná interpretace výstupů neuronů uvnitř sítě. U vícevrstvých sítí je z popisu dané problematiky znám pouze význam vstupních atributů a výstupů ANN, kdežto signály mezi vnitřními vrstvami nejsou dobře interpretovatelné a znepřehledňují výslednou strukturovanou bázi znalostí. Další vlastností většiny algoritmů tohoto typu je, že vyžadují přibližně prahovou přenosovou funkci neuronů, což také výrazně omezuje možnosti jejich použití. Počet extrahovaných pravidel je úměrný celkovému počtu neuronů v síti a počtu vstupních signálů jednotlivých neuronů. Proto při řešení složitějších úloh vyžadujících rozsáhlejší ANN hrozí exponenciální nárůst jejich počtu a tím i doby potřebné pro extrakci. Tato vlastnost musí být při návrhu algoritmu brána v potaz. Výhodou naopak je relativní jednoduchost těchto metod.

Na druhou stranu u tzv. pedagogického přístupu není počet pravidel dán strukturou neuronové sítě, ale velikost výsledné báze závisí na stanovení vhodného kritéria, které hodnotí jak přesně extrahovaná pravidla popisují znalost uloženou v ANN a určuje vhodný okamžik pro ukončení vytváření dalších pravidel. Při chybném stanovení tohoto kritéria tedy může poměrně snadno dojít k neúměrnému nárůstu počtu pravidel, takže se výsledná báze pravidel stane velice neprůhlednou. Ztratíme tak jeden ze základních motivů pro nasazení algoritmů pro extrakci pravidel, a to je zpřístupnit ve srozumitelné podobě znalost naučenou neuronovou sítí. Další nevýhodou tohoto přístupu je relativně vysoká časová náročnost vlastního procesu extrakce pravidel z již natrénované sítě, což je zapříčiněno nutností určitým způsobem nalézt antecedenty pravidel, to znamená nalézt hranice ve vstupním prostoru tak, aby co nejlépe separovaly jednotlivé třídy vzorů. Tato vlastnost se projeví zejména u problémů popsaných vyšším počtem vstupních atributů, kdy je nutné prohledávat mnohorozměrný prostor. Velkou výhodou ovšem je, že většina metod patřících do této kategorie není závislá na vlastní struktuře neuronové sítě (tyto algoritmy jsou univerzálně použitelné pro nejrůznější architektury sítí) a také to, že generovaná pravidla jsou ve tvaru, kdy antecedenty jsou definovány na základě hodnot vstupních atributů a v konsekventu je přímo hodnota výstupního parametru sítě.

Výše zmíněné nedostatky obou přístupů se projevily zejména při použití na klasické neuronové sítě typu back-propagation. Proto se v poslední době stále více prosazují ANN se speciální architekturou, která usnadňuje následnou extrakci pravidel a snaží se také spojit výhody dekompozičního a pedagogického přístupu (například možnost vytváření pravidel na základě interpretace jednotlivých neuronů se současným přímým mapováním vstupního prostoru na výstupní).

Dalším nedostatkem je, že dosud vytvářené algoritmy byly zaměřeny na úlohy se symbolickou výstupní třídou a nedokázaly zpracovávat spojitou výstupní proměnnou. V některých případech byl tento problém řešen prostřednictvím předzpracování výstupních hodnot formou diskretizace s následným použitím metod pracujících se symbolickou výstupní veličinou. Některé metody nebyly dokonce schopny pracovat ani se spojitými vstupními atributy.

3.3 POŽADAVKY NA ALGORITMY PRO EXTRAKCI PRAVIDEL Z ANN

Primární požadavky kladené na generovaná pravidla vyplývají již z jejich dalšího použití.

Z předchozího přehledu problematiky extrakce pravidel vyplývají některé požadavky na generovaná pravidla:

- zamezení redundantnosti pravidel,

- zajištění maximální jednoduchosti výsledných pravidel.

Splněním těchto dvou nejzákladnějších požadavků je dosaženo co největší průhlednosti výsledné báze pravidel, což je nezbytným předpokladem pro její další úspěšné použití. Ušlechtlí se tím např. detekce kritických kombinací vstupních hodnot jež nejsou pokryty žádným z výsledných pravidel nebo při nichž je aktivováno více pravidel s rozpornými závěry. Výhody maximální jednoduchosti znalostní báze se také projeví při použití vysvětlování založeného na pravidlech, které ze špatně organizované znalostní báze, v níž se mohou vyskytovat pravidla se stovkami předpokladů, nemůže být považováno za průhledné.

Dalším základním a samozřejmým požadavkem je, aby výsledná báze pravidel co nejlépe popisovala chování původní neuronové sítě. Tento požadavek se výrazně projevuje zejména při extrakci klasických pravidel. Je to dáno tím, že téměř všechny typy neuronových sítí používají neurony se spojitou přenosovou funkcí, která jim zajišťuje větší variabilitu a lepší schopnost učení. Proto při přechodu od neuronové sítě k pravidlům, která používají dvouhodnotovou logiku, dochází k určité ztrátě informací.

Shrneme-li výše uvedené požadavky (viz. 2. Cíle disertační práce bod C 1), dospějeme k závěru, že výsledkem další práce v oblasti automatizovaného vytváření bází pravidel pomocí jejich extrakce z natrénované ANN by tedy měla být neuronová síť s vhodnou strukturou navrženou tak, aby zohledňovala následný přechod na pravidla. Tím se dosáhne zachování maximálního množství informací zakódovaných v ANN a současně časová náročnost procesu extrakce nebude velká. Algoritmus by měl také zajistit generování co nejkompaktnější báze pravidel.

Dalším závěrem vyplývajícím ze studia publikovaných prací je absence algoritmů pro extrakci klasických pravidel v případě, že se u řešené problematiky nejedná o klasifikaci do diskrétních tříd, ale výstup je tvořen spojitou proměnnou. Bylo by vhodné, aby některý z nově vytvářených algoritmů byl schopen pracovat s těmito spojitými výstupy, ať už formou jejich automatické diskretizace (taková diskretizace by měla být součástí učení neuronové sítě, aby konečné diskrétní hodnoty byly optimalizovány v závislosti na podobě pravidel) a nebo dokonce ve formě, kdy budou konsekventy pravidel tvořeny funkcemi vstupních atributů (opět optimalizovanými v průběhu učení sítě).

Na základě těchto poznatků byly definovány cíle C 2, C 3 a C 4 této disertační práce (viz. kapitola 2. Cíle disertační práce).

4. EXTRAKCE PRAVIDEL Z ANN PRO ÚLOHY SE SYMBOLICKOU VÝSTUPNÍ TŘÍDOU

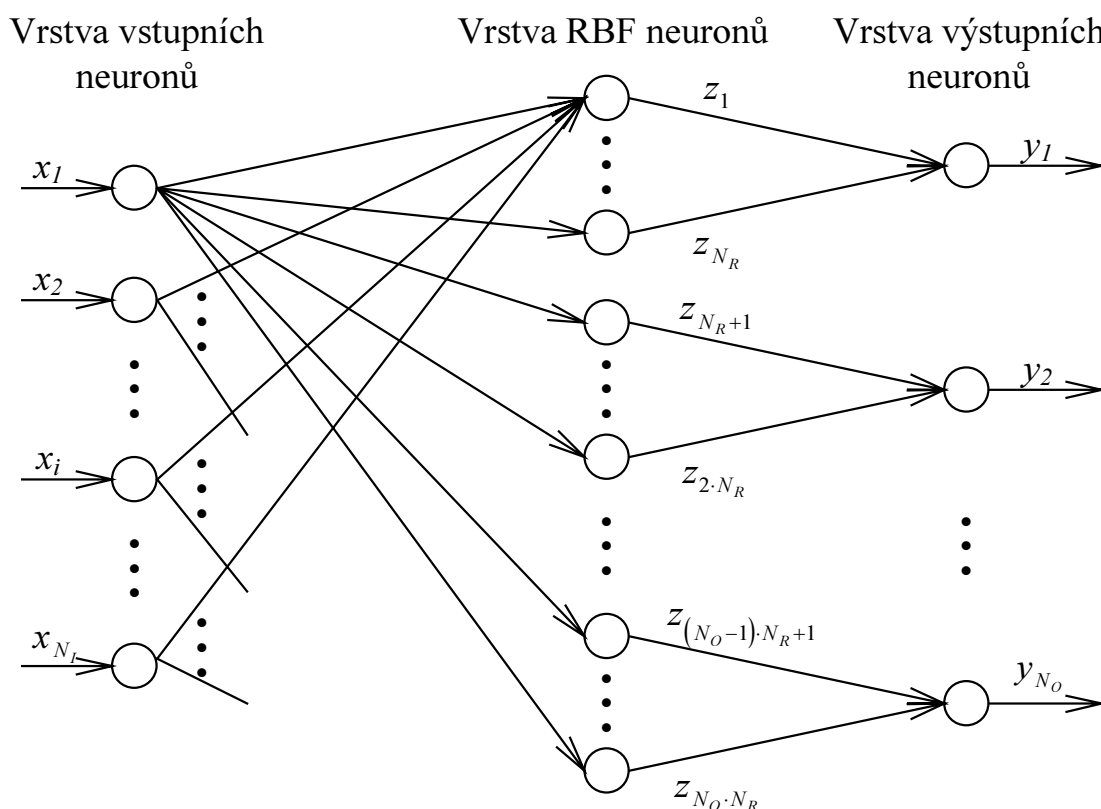
4.1 VÝSLEDNÁ STRUKTURA NEURONOVÉ SÍTĚ

Výsledná struktura ANN vychází ze sítě využívající neuronů s radiálními bázovými přenosovými funkcemi (RBF - Radial Basis Functions) (viz. [10]), které jsou svým tvarem vhodné pro modelování antecedentu pravidla. RBF neuronová síť je třívrstvá ANN, jejíž první vrstvu tvoří neurony šířící hodnoty vstupních atributů do všech neuronů prostřední vrstvy (rozdělovací vrstva). Tato druhá vrstva se skládá z RBF neuronů, které mají selektivní odezvu pro určitý rozsah vstupního signálu. Ve výstupní vrstvě klasické RBF neuronové sítě jsou sumační členy, jejichž vstupy jsou tvořeny výstupy RBF neuronů z_j násobenými vahami w_{kj} .

Tato struktura ANN byla modifikována tak, aby po naučení sítě byla následná extrakce báze pravidel, která předpokládáme ve tvaru

$$\text{if } (T_{L1} < a_1 < T_{H1}) \text{ and } (T_{L2} < a_2 < T_{H2}) \text{ and } \dots \text{ and } (T_{LN_1} < a_{N_1} < T_{HN_1}) \text{ then } y_{Rk}$$

co nejjednodušší.



Obr. 1 Schéma navržené neuronové sítě

Výsledná síť (viz. Obr. 1) je opět třívrstvá. Vstupní vrstva byla ponechána jako rozdělovací a počet neuronů v ní je roven počtu vstupních atributů. Vazby mezi první a druhou vrstvou nemají žádné váhy a propojují všechny neurony vstupní vrstvy se všemi neurony prostřední vrstvy.

V druhé vrstvě jsou RBF neurony s přenosovou funkcí:

$$z'_{ji} = \begin{cases} e^{-\frac{(x_i - t'_{Lji})^2}{\sigma_{ji}^2}} & \text{pro } x_i \leq t'_{Lji} \\ 1 & \text{pro } t'_{Lji} < x_i < t'_{Hji} \\ e^{-\frac{(x_i - t'_{Hji})^2}{\sigma_{ji}^2}} & \text{pro } x_i \geq t'_{Hji} \end{cases} \quad (1)$$

$$z_j = \prod_{i=1}^{N_I} z'_{ji}, \quad (2)$$

kde je x_i hodnota i -tého vstupního atributu,

t'_{Lji} hodnota dolního prahu jádra j -tého neuronu prostřední vrstvy pro i -tý vstupní atribut,

t'_{Hji} hodnota horního prahu jádra j -tého neuronu prostřední vrstvy pro i -tý vstupní atribut,

σ_{ji}^2 koeficient ovlivňující strmost poklesu funkce z'_{ji} jestliže se hodnota i -tého atributu vzdaluje od jádra j -tého neuronu prostřední vrstvy pro i -tý vstupní atribut (strmost přenosové funkce RBF neuronu),

z'_{ji} dílčí reakce j -tého neuronu prostřední vrstvy na hodnotu i -tého vstupního atributu,

z_j výstup j -tého neuronu prostřední vrstvy,

N_I počet vstupních atributů.

RBF neurony jsou rozděleny na N_O skupin (N_O - počet tříd) po N_R prvcích (N_R - maximální počet pravidel pro jednu třídu vzorů) a reprezentují jednotlivá výsledná pravidla. Celkově tak je v této vrstvě $N_R \cdot N_O$ neuronů.

Počet neuronů ve výstupní vrstvě je dán počtem tříd (N_O), do kterých se mají vzory roztrždit. Výstupní neuron má z pohledu výsledných pravidel funkci logického členu OR mezi pravidly vytvářenými pro rozpoznání vzorů jedné třídy. Pro tento účel byly použity klasické nelineární perceptrony se sigmoidou jako aktivační funkcí. Jejich přenosová funkce je:

$$y'_k = \sum_{j=1}^{N_R} z_j \quad (3)$$

$$y_k = \frac{1}{1 + e^{-\lambda(y'_k - \gamma)}}, \quad (4)$$

kde je y'_k výsledek agregační funkce na vstupu k -tého neuronu výstupní vrstvy,

N_R počet neuronů prostřední vrstvy patřících k jednomu výstupnímu neuronu,

z_j výstup j -tého RBF neuronu spojeného s k -tým výstupním neuronem,

y_k výstup k -tého neuronu výstupní vrstvy,

λ strmost sigmoidy,

γ práh sigmoidy.

Protože je potřebné, aby se výstup k -tého neuronu výstupní vrstvy blížil 1 v případě, že se 1 blíží výstup alespoň jednoho RBF neuronu patřícího do k -té třídy a naopak, aby tento výstup konvergoval k 0, konvergují-li k 0 výstupy všech RBF neuronů příslušné třídy, je práh výstupních neuronů nastaven na hodnotu 0.5.

4.2 SHRUTÍ VLASTNOSTÍ NAVRŽENÉHO ALGORITMU ZÍSKÁVÁNÍ BÁZE PRAVIDEL POMOCÍ ANN

Struktura neuronové sítě a postup používaný pro její učení byl zvolen tak, že neuronová síť splňuje požadavky, které vyplynuly z analýzy řešené problematiky (viz. kapitola 2. Cíle disertační práce bod C 2 a kapitola 3.3. Požadavky na algoritmy pro extrakci pravidel z ANN).

Díky použitému typu neuronů ve skryté vrstvě bylo zejména dosaženo minimalizace ztráty informace při přechodu od ANN k pravidlům. Podílil se na tom také postupné zvyšování strmostí přenosových funkcí RBF neuronů a výstupních neuronů, čímž se v průběhu doučování chování neuronové sítě přibližuje k chování pravidel.

Využití RBF neuronů také vedlo k současnému získání pozitivních vlastností jak dekompozičního tak i pedagogického přístupu. Při extrakci znalostní báze jsou vytvářena pravidla vyjadřující přímo vztah mezi vstupními atributy a výstupem, přičemž vlastní proces transformace ANN je velice jednoduchý a také rychlý. Délka celého procesu je tak dána zejména dobou potřebnou pro natrénování ANN. Další výhodou z toho plynoucí je získání stejně jednoduché metody i pro vložení již existující znalostní báze do neuronové sítě.

Díky implementovanému algoritmu zjednodušování pravidel a případně i snižování jejich počtu je zajištěno, že v bázi pravidel nedochází k redundancím (každé pravidlo v sobě obsahuje určitou část znalosti, která není obsažena v žádném jiném pravidle) a je maximálně kompaktní, což zvyšuje kvalitu získané znalosti.

4.3 SHRnutí EXPERIMENTÁLNÍCH VÝSLEDKŮ

S neuronovou sítí byly prováděny experimenty jak na speciálních umělých datech, která byla konstruována tak, aby co nejlépe otestovala základní vlastnosti navržené ANN, tak i na trénovací množině reálných dat. Výsledky získané metodou extrakce pravidel z natrénované ANN byly také porovnávány s výsledky dosaženými pomocí systému C4.5.

Přehled použitých množin dat:

- **Základní úloha (ZU)** – nejjednodušší úloha. Jednotlivé třídy jsou od sebe separovatelné konečným počtem pravidel.
Počet vstupních atributů/výstupních tříd: 2/2
Počet vzorů v trénovací/testovací množině: 1000/5000
Max. počet neuronů na jednu třídu ANN: 5
- **Lineárně separabilní problém (LSP)** – jednotlivé třídy jsou od sebe separovatelné nekonečným počtem pravidel.
Počet vstupních atributů/výstupních tříd: 2/2
Počet vzorů v trénovací/testovací množině: 1000/5000
Max. počet neuronů na jednu třídu ANN: 10
- **Lineárně neseperabilní problém (LNP)** – jednotlivé třídy jsou od sebe separovatelné nekonečným počtem pravidel.
Počet vstupních atributů/výstupních tříd: 2/2
Počet vzorů v trénovací/testovací množině: 1000/5000
Max. počet neuronů na jednu třídu ANN: 105
- **Neseperabilní problém (NP)** – jednotlivé třídy od sebe nejsou separovatelné (vzájemně se překrývají).
Počet vstupních atributů/výstupních tříd: 2/4
Počet vzorů v trénovací/testovací množině: 1000/5000
Max. počet neuronů na jednu třídu ANN: 5
- **Rozpoznávání znaků (RZ)** – identifikace černobílých pravoúhlých obrazců zobrazujících jedno z 26 velkých písmen anglické abecedy. Znaky byly vytvořeny na základě dvaceti různých fontů a každé písmeno v těchto dvaceti fontech bylo náhodně deformováno tak, aby byl vytvořen soubor 20000 unikátních vzorů. Každý z těchto vzorů byl popsán šestnácti jednoduchými numerickými atributy (statistické momenty a počty hran), které potom byly transformovány do množiny celočíselných hodnot od 0 do 15.
Počet vstupních atributů: 16
Počet výstupních tříd: 26 (velká písmena A - Z)
Počet vzorů v trénovací/testovací množině: 10400/9600
Max. počet neuronů na jednu třídu ANN: 25

V následující tabulce je uveden přehled dosažených výsledků navržené ANN a metody C4.5 na výše popsaných množinách dat.

Úloha	Metoda	Množina	Správně klasifikováno	Chybně klasifikováno	Správně i chybně klas.
ZU	ANN	trénovací	999 (99.90%)	0 (0.00%)	1 (0.10%)
		testovací	4987 (99.74%)	6 (0.12%)	7 (0.14%)
	C4.5	trénovací	1000 (100.00%)	0 (0.00%)	-
		testovací	4973 (99.46%)	27 (0.54%)	-
LSP	ANN	trénovací	996 (99.60%)	3 (0.30%)	1 (0.10%)
		testovací	4905 (98.10%)	92 (1.84%)	3 (0.06%)
	C4.5	trénovací	999 (99.90%)	1 (0.10%)	-
		testovací	4886 (97.72%)	114 (2.28%)	-
LNP	ANN	trénovací	980 (98.00%)	12 (1.20%)	8 (0.80%)
		testovací	4732 (94.64%)	228 (4.56%)	40 (0.80%)
	C4.5	trénovací	996 (99.60%)	4 (0.40%)	-
		testovací	4836 (96.72%)	164 (3.28%)	-
NP	ANN	trénovací	680 (68.00%)	303 (30.30%)	17 (1.70%)
		testovací	3299 (65.98%)	1580 (31.60%)	121 (2.42%)
	C4.5	trénovací	741 (74.10%)	256 (25.60%)	-
		testovací	3465 (69.30%)	1535 (30.70%)	-
RZ	ANN	trénovací	9993 (96.09%)	169 (1.62%)	238 (2.29%)
		testovací	7899 (82.28%)	935 (9.74%)	766 (7.98%)
	C4.5	trénovací	9465 (91.01%)	935 (8.99%)	-
		testovací	7841 (81.68%)	1759 (18.32%)	-

Z výsledků vyplývá, že algoritmus extrakce pravidel z ANN pro úlohy se symbolickou výstupní třídou realizovaný v rámci této disertační práce je schopen vytvářet velice kvalitní báze pravidel (viz. kapitola 2. Cíle disertační práce bod C 3).

Při testech se také projeví rozdíly mezi metodou založenou na extrakci pravidel z ANN a metodou založenou na budování rozhodovacích stromů. Rozdíly plynou z principu učení ANN a jedná se zejména o možnost neúplného pokrytí vstupního prostoru pravidly a také možnost vzájemného překrývání pravidel. Tyto vlastnosti mohou například usnadnit identifikaci kombinací vstupních parametrů nedostatečně popsanych ve stávající trénovací množině nebo obecně problematických kombinací hodnot vstupních atributů. U složitých úloh vyznačujících se zejména vysokým počtem výstupních tříd může částečné překrývání některých pravidel vést místo jednoznačně chybné klasifikace k vytvoření závěru, že daný vzor náleží do jedné ze dvou tříd. (Viz. Rozpoznávání znaků kde je z dosažených výsledků patrné, že asi 8 – 10 % testovacích vzorů bylo klasifikováno do dvou tříd z nich jedna byla vždy správná. To je u algoritmu pro rozpoznávání znaků také velice užitečné, protože konečná správná klasifikace může být provedena např. na základě slovníku přípustných kombinací písmen.).

5. EXTRAKCE PRAVIDEL Z ANN PRO ÚLOHY SE SPOJITOU VÝSTUPNÍ VELIČINOU

Neuronová síť umožňující zpracování úloh se spojitou výstupní veličinou vznikla jako zobecnění ANN pro úlohy se symbolickým výstupem. Před realizací tohoto zobecnění bylo vytyčeno několik cílů, jejichž splnění bylo při návrhu sledováno.

Základním cílem bylo zahrnutí diskretizace výstupní třídy do procesu učení ANN (na počátku je stanoven počet diskrétních úrovní, jejichž hodnoty se optimalizují v průběhu učení neuronové sítě).

Druhý, rozšiřující požadavek vyplynul z nevýhodnosti samotné diskretizace. Při použití malého počtu tříd se procesem diskretizace zanáší do systému značná chyba. Na druhou stranu zvyšování počtu tříd s sebou přináší problémy s narůstajícím počtem pravidel, snižujícím se počtem vzorů připadajících na jednu třídu a rostoucí nehomogenitou tříd. Všechny tyto vlivy zvyšují nebezpečí přetrénování ANN. Výhodnější by byla možnost aproximovat popisovanou funkci například pomocí polynomů, čímž by se dosáhlo daleko přesnějšího popisu při zachování nízkého počtu výsledných pravidel. To by mělo za následek další zprůhlednění získané znalosti a lepší orientaci v nelineárních vztazích mezi vstupními atributy a výstupem.

5.1 VÝSLEDNÁ STRUKTURA NEURONOVÉ SÍTĚ

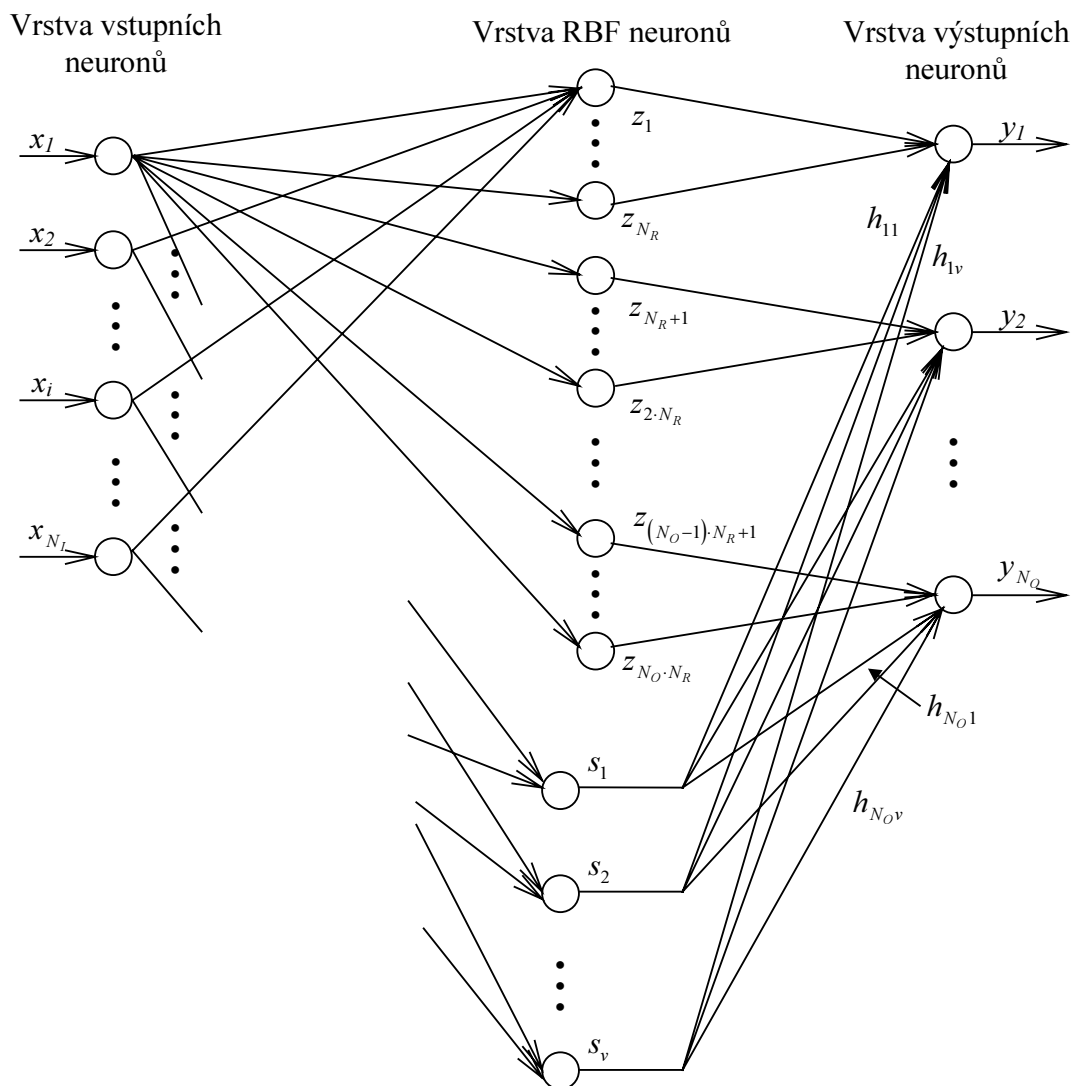
Struktura ANN vychází ze sítě navržené pro úlohy se symbolickou výstupní třídou, jejíž popis je uveden v kapitole 4.1. Výsledná struktura neuronové sítě. Tato architektura byla rozšířena o vrstvu jednotek počítajících hodnoty jednotlivých členů aproximačních polynomů použitých pro modelování antecedentů. Tyto neurony jsou na vstupech propojeny se vstupními neurony ANN vazbami o váze 1 a jejich výstup je propojen se všemi výstupními neurony sítě váhovanými vazbami. Jejich přenosová funkce s_m má tvar

$$s_m = \prod_{i=1}^{N_I} x_i^{l_i},$$

přičemž hodnoty l_i splňují následující podmínky:

$$l_1, l_2, \dots, l_{N_I} \in \langle 0, r \rangle,$$

$$\sum_{i=1}^{N_I} l_i \leq r.$$



Obr. 2 Architektura ANN pro zpracování úloh se spojitou výstupní veličinou

Takto upravená neuronová síť (viz. Obr. 2) umožňuje s příslušně modifikovaným algoritmem učení splnit požadavek na zahrnutí diskretizace do procesu adaptace a její zobecnění tak, aby konsekvent výsledných pravidel nemusel být tvořen pouze konstantou, ale i aproximačním polynomem zvoleného řádu.

5.2 EXPERIMENTÁLNÍ VYHODNOCENÍ

Přehled použitých množin dat:

- **Lineární funkce (LF)** – lineární funkce dvou proměnných

$$c = -2 + 1.5 \cdot a_1 + a_2$$

Počet vstupních atributů: 2

Počet vzorů v trénovací/testovací množině: 5000/1000

Max. počet neuronů na jednu třídu: 5

- **Kvadratická funkce (KF)** – kvadratická funkce dvou proměnných

$$c = 3 - 1.5 \cdot a_1 + 2 \cdot a_1 \cdot a_2 + 0.5 \cdot a_2^2$$

Počet vstupních atributů: 2

Počet vzorů v trénovací/testovací množině: 5000/1000

Max. počet neuronů na jednu třídu: 5

V následující tabulce je uveden přehled dosažených výsledků navržené ANN a metody C4.5 na výše popsanych množinách dat.

Úloha	Max. řád polynomu	Množina	Správně klasifikováno	Chybně klas.	Správně i chybně klas.	Střední kvadratická odchylka
LF	0	trénovací	4496 (89.92%)	333 (6.66%)	171 (3.42%)	0.444
		testovací	896 (89.60%)	70 (7.00%)	34 (3.40%)	0.438
	1	trénovací	5000 (100.00%)	0 (0.00%)	0 (0.00%)	$4.489 \cdot 10^{-8}$
		testovací	1000 (100.00%)	0 (0.00%)	0 (0.00%)	$4.592 \cdot 10^{-8}$
	2	trénovací	5000 (100.00%)	0 (0.00%)	0 (0.00%)	$4.489 \cdot 10^{-8}$
		testovací	1000 (100.00%)	0 (0.00%)	0 (0.00%)	$4.592 \cdot 10^{-8}$
KF	0	trénovací	4575 (91.50%)	266 (5.32%)	159 (3.18%)	10.954
		testovací	911 (91.10%)	61 (6.10%)	28 (2.80%)	11.176
	1	trénovací	4550 (91.00%)	285 (5.70%)	165 (3.30%)	4.457
		testovací	896 (89.60%)	67 (6.70%)	37 (3.70%)	4.763
	2	trénovací	5000 (100.00%)	0 (0.00%)	0 (0.00%)	$1.485 \cdot 10^{-3}$
		testovací	1000 (100.00%)	0 (0.00%)	0 (0.00%)	$1.351 \cdot 10^{-3}$

5.3 SHRNU TÍ

V této kapitole (5. Extrakce pravidel z ANN pro úlohy se spojitou výstupní veličinou) je prezentováno úspěšné rozšíření ANN popsané v kapitole 4. Extrakce pravidel z ANN pro úlohy se symbolickou výstupní třídou na úlohy se spojitou výstupní veličinou (viz. kapitola 2. Cíle disertační práce bod C 4).

Na dvou úlohách uvedených v kapitole 5.2. Experimentální vyhodnocení je ukázána schopnost navržené ANN generovat znalostní bázi s optimalizovaným rozložením jednotlivých pravidel v prostoru vstupních hodnot a také polynomů reprezentujících výstupní třídy. Algoritmus provedl vždy velmi efektivní rozdělení vstupních vzorů do jednotlivých tříd čehož výsledkem je velmi malý počet vzorů klasifikovaných do implicitní třídy (příslušné vzory nejsou pokryty žádným z pravidel – maximálně 0.7% vzorů) a také nízký počet vzorů současně pokrytých pravidly různých tříd (maximálně 3.7% vzorů).

Na výsledcích pro lineární funkci je také vidět, že učení proběhlo správně i v případě povoleného využití polynomů vyššího řádu než jakého dosahuje aproximovaná funkce.

Při implementaci se projeví určité problémy související zejména s vhodným zařazováním vzorů do jednotlivých tříd a z toho se odvíjející adaptace ANN. Proto je možné hlavní směr dalšího vývoje algoritmů pro extrakci pravidel z neuronových sítí vidět zejména v zdokonalování schopnosti zpracovávat úlohy se spojitou výstupní veličinou bez nutnosti předchozí diskretizace.

6. GENETICKÉ ALGORITMY PRO VYTVÁŘENÍ ZNALOSTNÍCH BÁZÍ FUZZY REGULÁTORŮ

Druhým nástrojem, který je náplní této disertační práce, jsou genetické algoritmy (Genetic Algorithms - GA). Jedná se o obecný optimalizační nástroj určený pro hledání řešení daného problému založený na mechanismech přirozeného výběru a přírodní genetiky.

Genetické algoritmy byly v tomto případě použity jako nástroj pro automatizované vytváření znalostní báze fuzzy regulátoru, která zahrnuje:

- rozsahy univerz vstupních a výstupní proměnné,
- definice tvarů fuzzy množin charakterizujících lexikální hodnoty vstupních a výstupní proměnné,
- báze fuzzy pravidel definující vlastní vztah mezi vstupními veličinami a výstupem fuzzy regulátoru.

6.1 PRINCIP GENETICKÝCH ALGORITMŮ

Úkolem GA jako optimalizačního prostředku je prohledávat složitý prostor potenciálních řešení a identifikovat ta nejlepší. Pojem *nejlepší řešení* se rozumí takové řešení, které optimalizuje předem stanovenou numerickou míru pro vyhodnocování konkrétního problému. Pro výpočet míry se používá funkce obvykle nazývaná jako *ohodnocovací funkce*.

GA pracuje v každé iteraci s celou množinou (populací) řešení (tzv. jedinců nebo chromozomů) čímž se zásadně liší od většiny ostatních optimalizačních metod. Jedinec je pak tvořen řetězcem bitů jejichž podmnožiny označované jako geny odpovídají jednotlivým nastavovaným parametrům.

V praxi se různé implementace GA mohou v detailech dosti lišit, avšak společně sdílejí některé typické rysy. Patří k nim především:

- iterativní zlepšování kvality členů množiny potenciálních řešení (populace),
- ohodnocování kvality jednotlivých členů populace v každé iteraci pomocí ohodnocovací funkce,

- generování nové populace pravděpodobnostním výběrem z nejprizpůsobenějších jedinců současné populace,
- použití genetických operátorů (křížení, mutace aj.) na vybrané jedince.

Obecně je možné činnost GA popsat následujícím postupem:

1. Zvol počet jedinců v populaci (velikost populace).
2. Stanov maximální počet zpracovávaných populací.
3. Definuj kritérium ukončení evoluce.
4. Vytvoř počáteční populaci buď pomocí náhodně generovaných hodnot atributů jedinců, nebo s využitím nějaké výchozí znalosti. Každého jedince reprezentuj pomocí řetězce složeného z podřetězců hodnot jednotlivých atributů. Každý jedinec musí být v populaci unikátní.
5. Iterativně prováděj následující kroky s populací, dokud není splněno ukončovací kritérium nebo nebyl-li dosažen maximální počet generací:
 - a) Každého jedince ohodnot' pomocí ohodnocovací funkce.
 - b) Vytvoř novou populaci z jedinců pomocí tří genetických operátorů (reprodukce, křížení a mutace) aplikovaných na každého jedince vybraného z populace na základě jeho kvality:
 - b1) Reprodukuj vybrané jedince z kopírováním do nové populace.
 - b2) Z nové populace vyber páry jedinců, v jejichž řetězcích náhodně zvol pozici pro křížení a proved' operaci křížení.
 - b3) Vzniklé nové řetězce modifikuj mutací na náhodně zvolených pozicích.
6. Identifikuj jedince s nejlepší mírou přizpůsobení jako (aproximační) řešení problému.

6.2 GA PRO GENEROVÁNÍ ZNALOSTNÍ BÁZE FUZZY REGULÁTORU

Konkrétní implementace algoritmu je charakterizována zejména:

- způsobem kódování nastavovaných parametrů do podoby řetězce bitů,
- implementací genetických operátorů – křížení a mutace,
- volbou průběhu testování členů populace (jedinců),
- tvarem ohodnocovací funkce a metodou přepočtu kvality jedinců na pravděpodobnost jejich přežití do další populace,
- a dalšími méně významnými parametry jako např. počet jedinců v populaci, způsob vytvoření počáteční populace apod.

V průběhu návrhu implementace GA pro generování znalostní báze fuzzy regulátoru byla zvláštní pozornost věnována několika jeho částem.

Prvním z nich byl způsob kódování optimalizovaných parametrů. Základním parametrem pro dekódování jednotlivých genů na skutečné hodnoty použité ve fuzzy regulátoru je velikost univerza. Velikost jednotlivých částí fuzzy množin není určena přímo hodnotou příslušného

genů, ale její relativní velikostí vůči hodnotám ostatních genů příslušejícím parametrům fuzzy množin stejné proměnné. Od toho se také odvíjí rozmístění fuzzy množin. Při testech s jinými úlohami se ukázalo, že tato částečná vnitřní provázanost je výhodná a usnadňuje optimalizaci.

Další z důležitých částí je postup pro přepočítání hodnot určujících kvalitu regulátoru reprezentovaného chromozómem na pravděpodobnost přežití tohoto chromozómu z jedné populace do následující. Zde je významným rysem schopnost tzv. škálování (automatická změna měřítka podle rozsahu vstupních hodnot) - při snižujícím se rozsahu hodnot ohodnocovací funkce chromozomů zůstává výsledný rozsah pravděpodobností téměř nezměněn. To umožňuje pokračovat v optimalizaci i v případě, že všichni jedinci v populaci mají takřka identickou kvalitu. Současně s touto vlastností byla také:

- dodržena základní podmínka pravděpodobnosti - součet pravděpodobností přežití všech jedinců v populaci je roven 1;
- zajištěna nenulová pravděpodobnost přežití všech jedinců;
- přibližně zachován relativní rozestup jedinců (jestliže rozdíl kvality jednoho chromozómu od druhého je dvakrát větší než druhého od třetího, pak také rozdíly pravděpodobností přežití budou přibližně dvojnásobné).

Velice vhodnou se ukázala také zvolená implementace mutace s proměnnou pravděpodobností inverze hodnoty některých bitů řetězce reprezentujícího člena populace, která zabraňuje degradaci populace (Při degradaci populace se všichni jedinci v ní shluknou poblíž jednoho bodu stavového prostoru což zamezí další optimalizaci).

Významnou částí je také kritériální funkce pro stanovení kvality jedinců, na jejímž tvaru závisí, jaký bude průběh regulačního děje. Lze pomocí ní ovlivnit, jaké požadavky bude výsledný regulátor splňovat (např. chceme, aby automobil jel po středu vozovky, aby po silnici „nekličkoval“ apod.) a jaký důraz bude při optimalizaci na tyto jednotlivé požadavky kladen (malá odchylka od středu vozovky nevádí, ale automobil by měl jet rovně).

Poslední podstatnou částí je volba podmínek, za kterých je prováděno testování regulátoru v průběhu jeho optimalizace (ty by měly dostatečně pokrývat pozdější provozní podmínky systému). Výsledný řídicí člen je totiž dobře navržen především na podmínky, při kterých pracoval v těchto testech.

6.3 EXPERIMENTÁLNÍ VYHODNOCENÍ

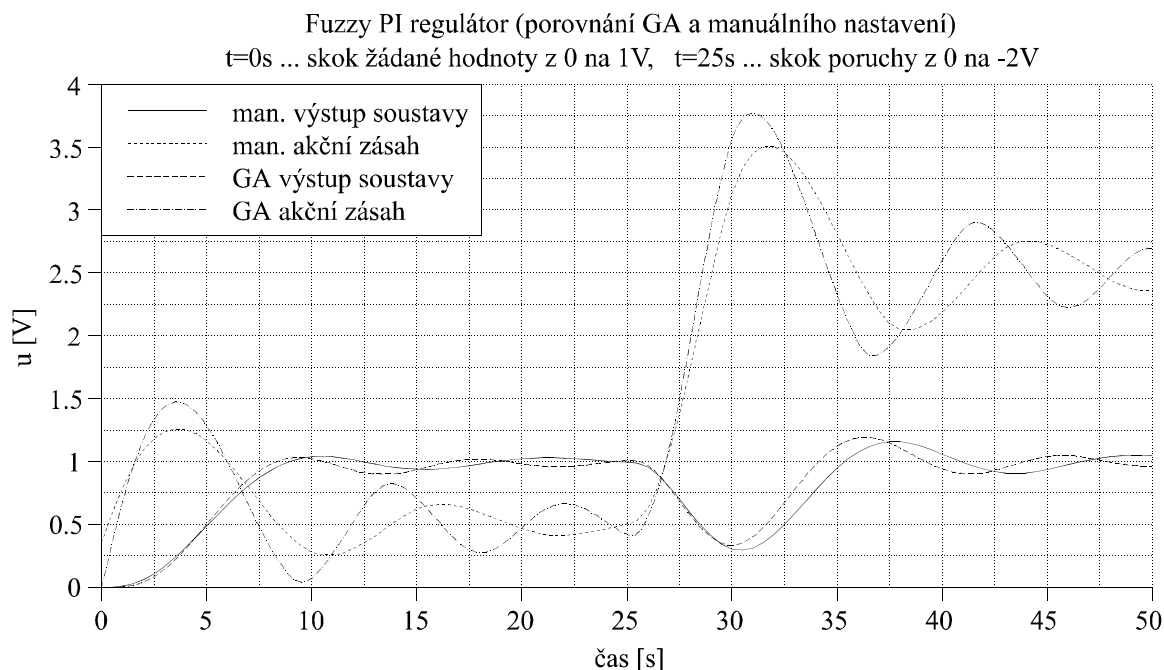
Navržená implementace genetických algoritmů byla testována na úloze regulace soustavy s přenosovou funkcí

$$F(s) = \frac{2}{(10 \cdot s + 1) \cdot (s + 1)^2}$$

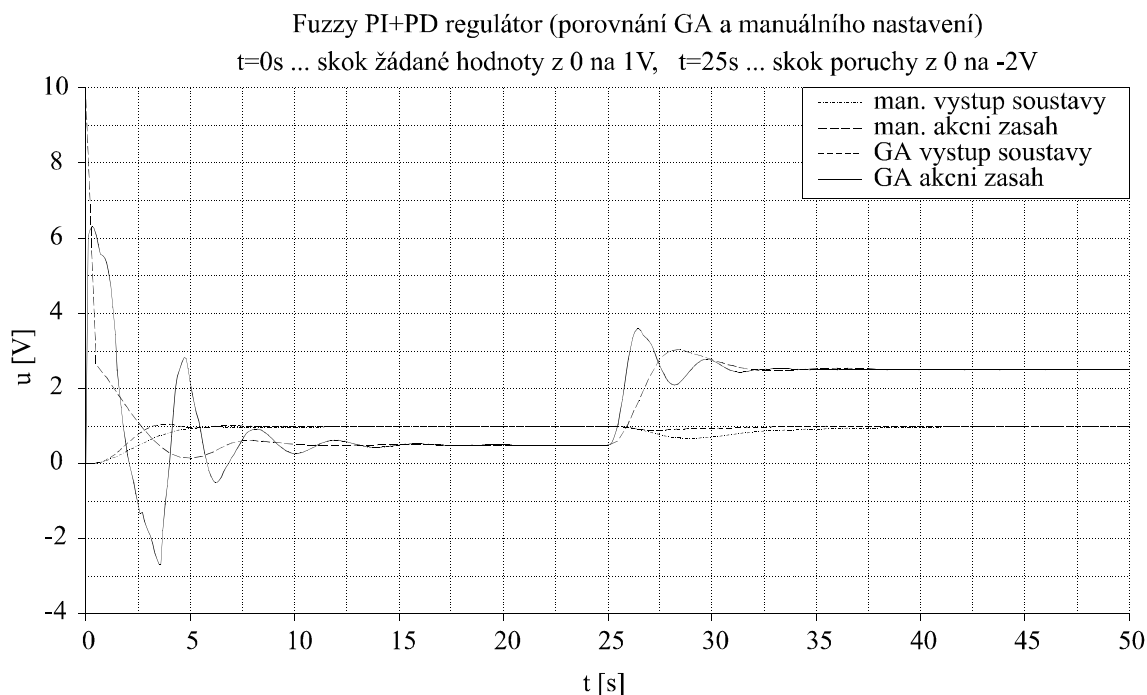
Perioda vzorkování byla stanovena na 0,1 s, požadovaná hodnota byla v rozmezí od -10 V do +10 V, omezení akčního zásahu bylo určeno na 10 V.

Testovací průběh byl definován následujícím způsobem:

- v čase $t = 0$ s skoková změna požadované hodnoty z 0 V na +1 V,
- v čase $t = 25$ s skoková porucha (působící na vstupu soustavy) z hodnoty 0 V na -2 V,
- překmit nesmí přesáhnout 5 % změny požadované hodnoty.



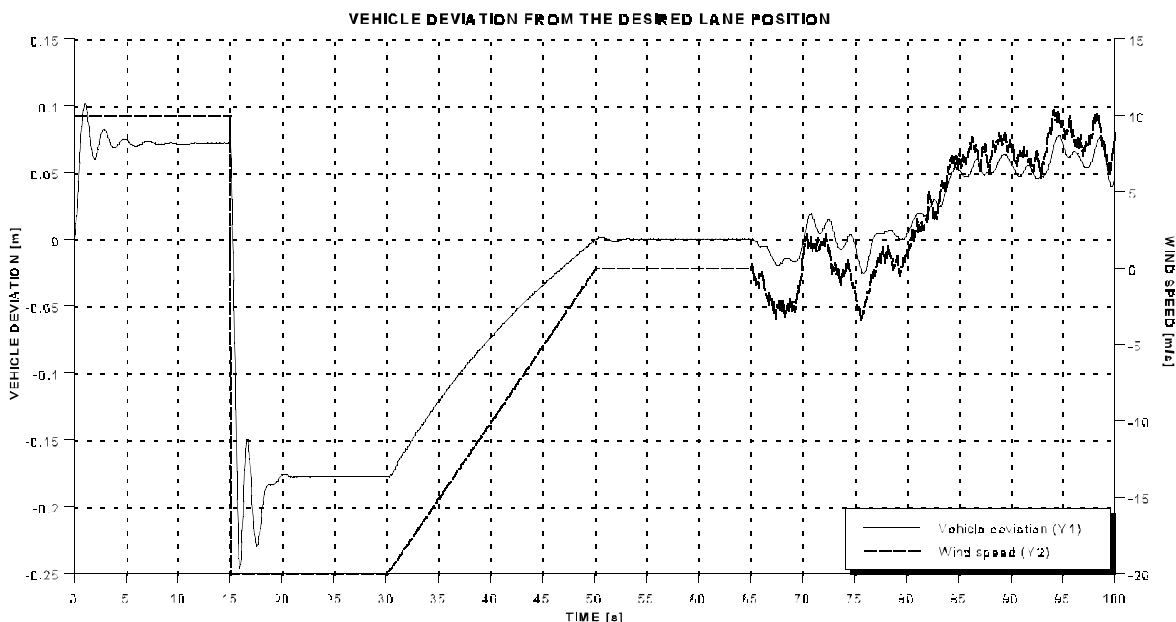
Obr. 3 Porovnání vlastností fuzzy PI regulátoru vyvinutého pomocí GA a fuzzy PI regulátoru nastaveného



Obr. 4 Porovnání vlastností fuzzy PI+PD regulátoru vyvinutého pomocí GA a fuzzy PI+PD regulátoru nastaveného manuálně

Pro tuto úlohu byly pro porovnání k dispozici také výsledky dosažené při ručním nastavení fuzzy regulátoru (viz. Obr. 3 a Obr. 4).

Výhody vytvořeného algoritmu se plně projeví na úloze kompenzace vlivu bočního větru na jedoucí automobil (viz. Obr. 5). Jedná se o složitý nelineární problém (při optimalizaci byl pro testování používán rovinný dynamický model vozidla), ve kterém je kompenzována porucha působící na řízenou soustavu. Na této úloze byla také velice úspěšně testována možnost návrhu regulátoru za přítomnosti šumu ve vstupních datech.



Obr. 5 Kompenzace vlivu bočního větru na jedoucí automobil

6.4 SHRnutí

Podarilo se úspěšně realizovat systém pro automatické vytváření a optimalizaci znalostních bází fuzzy regulátorů, který se ukázal být velice silným nástrojem (viz. kapitola 2. Cíle disertační práce bod C 5 a C 6). Při všech prováděných testech bylo dosahováno vynikajících výsledků. Výrazným přínosem vytvořeného algoritmu oproti metodám ručního nastavování je lepší využití potenciálu fuzzy regulátorů zejména s ohledem na jejich možnou nelinearitu. Výhody se projevují tím výrazněji, čím složitější řídicí člen je vytvářen. Také stále se zvyšující výkon počítačů způsobující urychlení optimalizace má vliv na rostoucí použitelnost této metody.

Vytvořený algoritmus dokázal například během několika hodin (přibližně 8 hodin na počítači osazeném procesorem Pentium 6x86 P150 na 120 MHz) optimalizovat fuzzy regulátor pro kompenzaci vlivu bočního větru na automobil skládající se z paralelní kombinace dvou PI a dvou PD fuzzy regulátorů. Pro zhodnocení složitosti takového regulátoru je nutné si uvědomit, že jeho znalostní báze může při povoleném maximálním počtu

7 fuzzy množin pro každou vstupní i výstupní veličinu obsahovat až 9604 pravidel. Dále vyžaduje optimalizaci velikosti 12 univerz a optimalizaci tvaru a umístění až 140 funkcí příslušnosti fuzzy množin. Kvalitní ruční nastavení takového regulátoru by bylo daleko časově náročnější ne-li zcela nemožné.

7. ZÁVĚR

V předložené disertační práci jsou představeny dva velice silné nástroje z oblasti soft-computing, umělé neuronové sítě a genetické algoritmy, jako prostředky pro automatizované budování znalostních bází. Ani jeden z těchto algoritmů nebyl původně vytvořen pro řešení tohoto úkolu. ANN jsou zejména známy jako mocný nástroj pro oblast klasifikace vzorů a aproximace složitých funkcí, který čerpá znalost z množiny trénovacích vzorů. Chybí zde však možnost zpřístupnění získané znalosti ve srozumitelné podobě. Genetické algoritmy jsou zase velice obecným optimalizačním nástrojem vyžadujícím při použití v konkrétní oblasti úloh vhodnou realizaci celé řady segmentů (genetické operátory, způsob kódování optimalizovaných parametrů), na kterých záleží výsledná úspěšnost. Oba výše zmiňované nástroje prokázaly v implementacích, které byly vytvořeny v rámci této disertační práce, svou schopnost automaticky produkovat kvalitní znalostní báze. Vynikajících výsledků bylo dosaženo zejména při aplikaci genetických algoritmů pro návrh fuzzy regulátorů, kdy se podařilo vhodným způsobem realizovat všechny části GA. Tím byly úspěšně splněny cíle stanovené v kapitole 2. Cíle disertační práce (podrobnější rozbor splnění jednotlivých cílů je možné nalézt v kapitolách 3.3, 4.2, 4.3, 5.3 a 6.4).

System na bázi neuronové sítě s RBF neurony byl naprogramován jako univerzální nástroj pro vytváření klasických pravidel v podobě

$$\text{if}(T_{L1} < a_1 < T_{H1}) \text{and}(T_{L2} < a_2 < T_{H2}) \text{and} \dots \text{and}(T_{LN_i} < a_{N_i} < T_{HN_i}) \text{ then } y_{Rk}$$

pro oblast rozpoznávání vzorů a bylo také prezentováno možné rozšíření této neuronové sítě pro řešení úloh se spojitou výstupní veličinou. Kvalita vytvořené báze pravidel byla porovnávána s výsledky dosaženými pomocí C4.5 (velice úspěšný nástroj z oblasti umělé inteligence pro vytváření rozhodovacích stromů s možností následné transformace na pravidla). Při dosahování shodné klasifikační přesnosti byla pomocí ANN mnohdy produkována kompaktnější báze pravidel. Při testech se také ukázaly rozdíly mezi vlastnostmi znalostní báze vytvořené na metodou využívající rozhodovací stromy a extrakci pravidel z ANN plynoucí z principu učení ANN.

Další možné pokračování vývoje algoritmů pro extrakci pravidel z neuronových sítí by mělo být zaměřeno zejména na zpracování úloh se spojitou výstupní veličinou bez nutnosti předchozí diskretizace. V této oblasti

je vývoj ještě na počátku. I u zde prezentované metody se projeví určité problémy související zejména s vhodným zařazováním vzorů do jednotlivých tříd a z toho se odvíjející adaptace.

Metoda založená na genetických algoritmech byla navrhována pro specifickou oblast automatizovaného vytváření znalostníchází fuzzy regulátorů. Tento systém umožňuje u všech veličin optimalizaci velikostí univerz, rozmístění fuzzy množin a jejich tvaru a případným odstraňováním některých fuzzy množin oproti jejich maximálnímu povolenému počtu také zjednodušovat fuzzy pravidla. Při testování byly vytvářeny fuzzy regulátory pro řízení lineární dynamické soustavy 3. řádu, pro kterou byl k dispozici také ručně optimalizovaný fuzzy regulátor. Z porovnání dosažených výsledků vyplynulo, že kvalitnější řízení je získáno při použití fuzzy regulátoru vytvořeného genetickým algoritmem. Další testy pak byly prováděny na složitější nelineární soustavě, kterou byl rovinný dynamický model automobilu. V tomto případě bylo cílem kompenzovat vliv bočního větru působícího v průběhu jízdy. I zde dosahovaly regulátory optimalizované GA velice dobrých výsledků. Ukázalo se, že je možné touto metodou vytvořit fuzzy regulátory schopné zvládnout náročné provozní podmínky (šum ve vstupních datech, mění se dynamika systému). Předpokladem pro to však je správná volba podmínek, za kterých je prováděno testování v průběhu optimalizace. Výsledný regulátor je dobře navržen především na podmínky, při kterých pracoval v těchto testech.

Velice zajímavým směrem pro další vědeckou práci by mohlo být využití zde prezentovaného algoritmu pro nalezení množiny pravidel pro ruční nastavování fuzzy regulátorů, a to zejména pro systémy obsahující některé základní nelinearity jako jsou např. tření, vůle v převodech apod. Tato studie by však vyžadovala dlouhodobou a systematickou práci a její výsledky by mohly být náplní samostatné diplomové nebo disertační práce.

8. CONCLUSIONS AND PROSPECTIVES

This work presents two powerful tools for automatic generation of knowledge bases. Both of these new systems originated from the soft-computing area, namely from the area of Artificial Neural Networks (ANNs) and of Genetic Algorithms (GAs).

The common purpose of ANNs is pattern classification and approximation of complex dependencies using training sets; in many tasks, a perceptible disadvantage of ANNs is the incomprehensible form of extracted knowledge. GAs are a very flexible general optimisation tool; their basic weakness is that their application to a concrete task requires lots of ad-hoc specifications from the user (e.g. genetic operators, coding of optimised parameters) that impact the achievement substantially. The systems developed within this work exceed these basic shortcomings of ANNs and GAs. During the tests, both of the presented systems proved the ability to produce automatically knowledge bases of high quality - specifically for pattern recognition and for fuzzy-controller construction.

The first system developed within this work is an extension of ANNs with RBF neurons (neurons with Radial Basis transfer Functions). The system produces knowledge bases consisting of common rules in the form

$$\textit{if } (T_{L1} < a_1 < T_{H1}) \textit{and } (T_{L2} < a_2 < T_{H2}) \textit{and } \dots \textit{and } (T_{LN_i} < a_{N_i} < T_{HN_i}) \textit{then } y_{Rk}$$

for pattern recognition area. Additionally, a modification of this system for tasks with continuous output quantity was suggested and implemented. The quality of the generated rule base was compared against the benchmark rule base obtained by C4.5. (C4.5 is one of the best known artificial-intelligence based algorithms for knowledge bases extraction from training sets.) The result of the tests was that the classification accuracy was the same; moreover, the extended ANN system often produced a more compact rule base. The extended ANN system may show also other differences: Several contradictory rules overlap, or - on the contrary - a part of the input space is not covered at all. This particularity may sometimes appear as a disadvantage, but in concrete cases may be very useful: Missing rules may detect insufficiently covered regions of the input space in the training set. In complex tasks with numerous output classes, the contradictory classification of a pattern may evoke use of additional context information (e.g. use of a dictionary of allowable letter combinations) to particularise the classification.

Further research of ANNs application to rules extraction should focus on tasks with continuous output quantity, as the recent state of this field is in an

initial phase only. Also the method developed within the presented work is just a first step.

The other system created within this work is a specific GA that was evolved for the generation of knowledge bases for fuzzy controllers. This system provides automatically the necessary optimisations, particularly the optimisation of the scope of universe, the location of fuzzy sets, and the shape of fuzzy sets. Even more, the system is able to reduce the number of generated fuzzy rules and to simplify individual fuzzy rules by removing some components. To verify the achievement of this system, two kinds of tests were performed: The GA system generated fuzzy controllers for linear dynamic system of 3rd order, for which a manually optimised fuzzy controller was already available. The tests demonstrated that the control performance by GA-optimised fuzzy controller exceeded the manually optimised fuzzy controller. Further tests were performed on a more complex non-linear system – on a plane dynamic model of the car. In this case, the task was to compensate the side wind influences during car ride. The GA-optimised fuzzy controllers proved very excellent results. These controllers managed also difficult operational conditions (noisy input data, changing dynamic of the controlled system); however, the requirement was an appropriate choice of conditions during optimisation – the controllers worked best under analogous conditions.

An interesting topic for further research in the area of GA-based optimisation of fuzzy controllers is the application of the presented method to construction of sets of rules for manual setting of fuzzy controllers, particularly for dynamic systems with basic types of non-linearity.

9. LITERATURA

- [1] Andrews, R. - Diedrich, J. - Giles, L.: Rule Extraction from Neural Networks, workshop on Rule Extraction from Trained Neural Networks, Brighton, England, 1996.
- [2] Andrews, R. - Diedrich, J. - Tickle, A. B.: A Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks, Queensland University of Technology, Australia, 1995.
- [3] Berenji, H. R.: A Reinforcement Learning-Based Architecture for Fuzzy Logic Control, in Fuzzy Sets for Intelligent Systems (D. Dubois, H. Prade and R. R. Yager, Eds.), Morgan Kaufman Pub., San Mateo, California, pp.368 - 380.
- [4] Berenji, H. R.: Refinement of Approximate Reasoning-based Controllers by Reinforcement Learning, Proc. of the Eighth International Machine Learning Workshop, 1991, pp. 475 - 479.
- [5] Blasig, R.: GDS: Gradient Descent Generation of Symbolic Classification Rules, pp. 1093 - 1100.
- [6] Craven, M. W. - Shavlik, J. W.: Using Sampling and Queries to Extract Rules from Trained Neural Networks, Proc. of the Eleventh International Conference on Machine Learning, San Francisco, CA, 1994, pp. 37 - 45.
- [7] Duch, W. - Adamczak, R. - Grabczewski, K.: Extraction of Logical Rules from Training Data Using Backpropagation networks.
- [8] Gallant, S. I.: Neural Network Learning and Expert Systems, MIT Press, Cambridge, Massachusetts.
- [9] Goldberg, D. E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Company, 1989.
- [10] Hassoun, M. H.: Fundamentals of Artificial Neural Networks, A Bradford Book, The MIT Press, 1995.
- [11] Ivanova, I - Kubat, M.: Initialization of Neural Networks by Means of Decision Trees, Knowledge Based Systems 8, 1995, pp. 333 - 344.
- [12] Jura P.: Fuzzy control of the extraction of organic binders from ceramic materials. KoREMA'96, September 18-20, 1996, Vol.2, pp.41-44, Opatia, Croatia.
- [13] Jura P. - Šolc F.: Struktura a dynamika fuzzy regulátoru. Automatizace 39, ISSN 0005-125X, Praha, Česká republika, č. 9/96, s. 406-410.
- [14] Jura P.: Základy fuzzy logiky pro řízení a modelování", skriptum FEI VUT Brno, 1998, 92 stran, ISBN 80-214-1203-8, vydal PC-DIR Real, s.r.o., Brno.

- [15] Kim, H. M. - Mendel, J. M.: Fuzzy Basis Functions: Comparisons with Other Basis Functions, IEEE Transactions on Fuzzy Systems, May 1995, Vol. 3, No. 2, pp. 158 - 168.
- [16] Kohn, M.: Udržování přímého směru jízdy vozidla systémem řízeným fuzzy pravidly. Diplomová práce, ÚAMT FEI VUT Brno, květen 1994.
- [17] Kruse, R. – Gebherdt, J. – Klawonn, F.: Foundation on Fuzzy Systems. J. Wiley and Sons, 1994.
- [18] Leitch, D. – Probert, P.: Genetic Algorithms for the Development of Fuzzy Controllers for Autonomous Guided Vehicles. Proceedings of the Second European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany, September 20 – 30, 1994, Vol. 1, pp. 464 – 469.
- [19] Nauck, D. - Kruse, R.: NEFCLASS - A Neuro-Fuzzy Approach for the Classification of Data, presented at the Symposium on Applied Computing, Nashville, 1995.
- [20] Nauck, D.: Building Neural Fuzzy Controllers with NEFCON-I. In: Kruse, R. - Gebhard, J. - Palm, R.: Fuzzy Systems in Computer Science. Vieweg Verlag, 1994.
- [21] Novák, M.: Neuronové sítě a neuropočítače. Výběr, SENZO, 1992.
- [22] Novák, M. - Faber, J. - Kufudaki, O.: Neuronové sítě a informační systémy živých organismů. Praha, Grada, 1995.
- [23] Omlin, Ch. W. - Giles, C. L.: Constructing Deterministic Finite-State Automata in Recurrent Neural Networks, published in Journal of the ACM, 1996, Vol. 43, No. 6, pp. 937 - 972.
- [24] Pivoňka, P.: Fuzzy PI/PD/PID regulátory. Automatizace, ISSN 0005-125X, Praha, Česká republika, č. 6/98-8/98, s. P1-P11.
- [25] Pivoňka, P. - Žižka, J.: The Use of Fuzzy PI/PD/PID Controllers in Fuzzy Control. Proc. of the Third European Congress EUFIT'95, Vol. 2, Aachen, Germany, pp. 1041 – 1045.
- [26] Řeřucha, V.: Intelligent Control - Modern part of Cybernetics. In: Artificial Intelligence in Control and Measurement, International Summer School, CEPUS SK-46, Brno, 2000, pp 114-119.
- [27] Řeřucha, V.: Inteligentní řízení krácejícího robota. Habilitační práce. Brno, 1997, 240 str.
- [28] Setiono, R. - Liu, H.: NeuroLinear: A System for Extracting Oblique Decision Rules from Neural Networks, Proc. of the ECML, Praha, 1997, pp. 221 - 233.
- [29] Towell, G. G. - Shavlik, J. W.: Extracting Refined Rules for Knowledge-Based Neural Networks, Mashine Learning, Vol. 13, No. 1, October 1993, pp. 71 - 101.
- [30] Towell, G. G. - Shavlik, J. W.: Knowledge-Based Artificial Neural Networks, Artificial Intelligence, 70:1-2, pp. 119-165.

- [31] Widmer, G. – Kubát, M.: Learning in the Presence of Concept Drift and Hidden Context. *Machine Learning*, 1996, 23(1), pp. 69 – 101.
- [32] Winston, P. H.: *Artificial Intelligence*. 3rd Edition. Addison-Wesley Publishing Company, 1992.
- [33] Žižka, J. – Pivoňka, P.: The Influence of Selected Inference Operators on the Design of Fuzzy Controllers. *Proc. of the 2nd International Workshop on Artificial Intelligence Techniques AIT'95*, September 18-20, 1995, Brno, Czech Republic, pp. 127-139.
- [34] Žižka, J.: Fuzzy množiny v řízení a regulaci. *Automatizace*, ISSN 0005-125X, Praha, Česká republika, č. 11/97-12/97, s. P1-P11.

10. PUBLIKOVANÉ PRÁCE

10.1 PUBLIKOVANÉ PRÁCE VZTAHUJÍCÍ SE K TÉMATU DISERTAČNÍ PRÁCE:

- [1] Zizka, J. - Kratochvil, Z.: An Application of Genetic Algorithms to the Automatic Generation of the Fuzzy-Logic Based Car-Controllers, Proc. of 7th International DAAAM Symposium, ISBN 3-901509-02-x, Vienna, Austria, 1996, pp. 497-498.
- [2] Zizka, J. - Kratochvil, Z.: Genetic Algorithms-Based Creation of Fuzzy-Rule Bases for the Car Control, Proc. of 23rd International Seminar on Current Trends in Theory and Practice of Informatics, ISBN 0302-9747, Milovy, Czech Republic, November 1996, pp. 475-482.
- [3] Zizka, J. - Kratochvil, Z.: Genetic Algorithms-Based Evolution of Multiple Parallel PI+PD Fuzzy-Controller Systems with Noisy Input Data, Proc. of the 7th World Congress of International Fuzzy Systems Association, ISBN 80-200-0633-8, Praha, Czech Republic, June 1997, Vol. 2, pp. 469-474.
- [4] Zizka, J. - Kratochvil, Z.: An Application of RBF – Based ANN to Extraction of Rules, Proc. of the NC'98 International ICSC/IFAC Symposium on Neural Computation, ISBN 3-906454-14-2, Viena, Austria, September 23-25, 1998, pp. 558-564.
- [5] Zizka, J. – Kratochvil, Z. – Popelinsky, L.: Simple Speech Classification Using Automatically Generated Rules, Proc. of the First Workshop on Text, Speech, Dialogue TDS'98, ISBN 80-210-1899-2, Brno, Czech Republic, September 23-26, 1998, pp. 339-344.
- [6] Kratochvil, Z. - Žižka, J.: Genetické algoritmy pro nastavování fuzzy regulátorů, Automatizace, ISSN 0005-125X, Praha, Česká republika, č. 4/99-6/99, s. P39-P47.

10.2 OSTATNÍ PUBLIKOVANÉ PRÁCE:

- [7] Pavlickova, J. - Kratochvil, Z.: Integration of Artificial-Intelligence Techniques in Controller Design, Proc. of the 8th International DAAAM Symposium Intelligent Manufacturing Automation, ISBN 3-901509-04-6, Dubrovnik, Croatia, October 1997, pp. 261 - 262.
- [8] Pavlickova, J. – Kratochvil, Z.: Regression Tree Models for Load Forecasting: Initial Case Study, Proc. of the 9th International DAAAM Symposium "Intelligent Manufacturing, Automation & Networking", ISBN 3-901509-08-9, Cluj-Napoca, Romania, October 22-24, 1998, pp. 375-376.

11. ŽIVOTOPIS AUTORA

Ing. Zdeněk Kratochvíl
narozen 30. 5. 1973 v Brně

Vzdělání:

- říjen 1996 Interní postgraduální doktorandské studium v odboru
- září 1999 Kybernetika a informatika – Vysoké učení technické v Brně,
Fakulta elektrotechniky a informatiky.
- říjen 1991 Inženýrské studium v odboru Automatizace a měřicí
- červen 1996 technika – Vysoké učení technické v Brně, Fakulta
elektrotechniky a informatiky.
Státní závěrečná zkouška – diplom inženýra elektrotechniky
s vyznamenáním; ocenění děkana za diplomovou práci.
- květen 1991 Maturitní zkouška s vyznamenáním na SPŠ elektrotechnické
v Brně.

Odborná praxe:

- od října 1999 Zaměstnán jako samostatný programátor a analytik
(zaměřený na programování v Oracle) v softwarové firmě
Berit, a. s. zabývající se databázovými aplikacemi (GIS).
V současné době v následujících pozicích:
- analytik a konzultant pro jednoho z významných zákazníků
firmy,
- produktový specialista a konzultant pro Oracle,
- správce některých instancí firemního Oracle serveru.
- září 1997 Práce na zakázku pro současného zaměstnavatele
- říjen 1999 (databázové aplikace, programování SQL a PL/SQL).
- od července Soukromé podnikání v oblasti poskytování software.
1996
- říjen 1996 V rámci postgraduálního doktorandského studia vědecká
- září 1999 činnost v oblasti umělé inteligence (zejména metody soft
computing), vedení projektů a výuka v oboru strojového
učení a umělé inteligence.
- od června 1996 Spolupráce s firmou Baspelin, s. r. o. v oblasti počítačového
- ledna 2000 řízení procesů, shromažďování a analýza snímaných dat.
Vývoj softwarového systému – zejména programování
v Delphi.