

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta elektrotechniky a komunikačních technologií
Ústav telekomunikací

Ing. Karol Molnár

**APLIKACE UMĚLÝCH NEURONOVÝCH SÍTÍ
VE VYSOKORYCHLOSTNÍCH
AKTIVNÍCH SÍŤOVÝCH PRVCÍCH**

**APPLICATION OF ARTIFICIAL NEURAL NETWORKS
IN HIGH-SPEED ACTIVE NETWORK ELEMENTS**

ZKRÁCENÁ VERZE PHD THESIS

Obor: Teleinformatika

Školitel: Doc. Ing. Vladimír Kapoun, CSc.

Oponenti: Doc. Ing. Jana Tučková, CSc.

Prof. Ing. Vladimír Mikula, CSc.

Doc. Ing. František Židek, CSc.

Datum obhajoby: 6. 12. 2002

KLÍČOVÁ SLOVA

neuronová síť, Hopfieldova neuronová síť, optimalizace, spojovací pole, prioritní přepínání

KEYWORDS

neural network, Hopfield neural network, optimization, switch fabric, priority switching

Disertační práce je uložena na oddělení vědy a výzkumu, Fakulta elektrotechniky a komunikačních technologií, Vysoké učení technické v Brně, Údolní 53, Brno

© Karol Molnár, 2003

ISBN 80-214-2324-2

ISSN 1213-4198

OBSAH

1	ÚVOD	5
2	CÍLE PRÁCE	6
3	SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY	7
3.1	Hopfieldova neuronová síť	7
3.2	Základní matematický popis Hopfieldovy sítě	8
3.2.1	Energetická funkce klasické Hopfieldovy neuronové sítě	9
3.3	Optimalizační problémy	9
3.4	Matematické vyjádření kombinatorních optimalizačních úloh	10
4	ZVOLENÉ METODY ZPRACOVÁNÍ	12
4.1	Optimalizace přepínání	12
4.1.1	Model přepínače využívající umělou neuronovou síť	12
4.2	Podprostor platných řešení	14
4.2.1	Definice vztahů pro pomocné vektory a matice	14
4.2.2	Algebraická reprezentace vektoru výstupních veličin	14
4.2.3	Podprostor nulového součtu	15
4.2.4	Obecný popis podprostoru platných řešení	16
4.2.5	Transformace do podprostoru platných řešení	17
5	HLAVNÍ VÝSLEDKY PRÁCE	18
5.1	Praktická implementace Hopfieldovy sítě pro řešení problému prioritního přepínání	18
5.2	Simulační program pro umělé neuronové sítě	19
5.3	Výsledky testů	19
5.4	Možnosti implementace výsledků práce v aktivních prvcích	22
5.5	Zefektivnění architektury přepínače	23
6	ZÁVĚR	25
	POUŽITÁ LITERATURA	26

1 ÚVOD

Konvergence klasických telekomunikačních sítí a datových sítí je úvodním krokem k vytvoření univerzální širokopásmové integrované sítě zajišťující provozování různých typů uživatelských služeb, včetně náročných videokonferenčních aplikací či multimediálních služeb a jednotné správy sítě. Integrovaná síť musí být schopna zaručit odlišné parametry přenosu, požadované odlišnými službami. Problém je v síťových prvcích, které požadované parametry spojení musí garantovat a navíc poskytnout dostatečně velkou šířku pásma a to vše za přijatelnou cenu. Proto se v oblasti vývoje vysokorychlostních síťových prvků hledají hlavně možnosti, jak zvýšit propustnost a efektivitu aktivních prvků.

Základem všech komunikačních sítí jsou aktivní prvky, které provádějí zpracování přenášených datových jednotek a na základě výsledků zpracování předávají datové jednotky směrem od odesílatele k příjemci. V současnosti nejnáročnější úlohou aktivních prvků je stanovení, ve kterém časovém okamžiku a kterou datovou jednotku má systém zpracovat, aby zpracování odpovídalo prioritě přidělené jednotlivým datovým jednotkám.

Klasické sekvenční zpracování dat je omezeno rychlostí centrální procesorové jednotky. Zvyšující se nároky na rychlost procesorů kladou čím dál tím větší kvalitativní požadavky na technologii výroby. Jiné řešení nabízí paralelní zpracování dat. Při paralelním řazení více komplexních funkčních prvků se značně zvýší nároky na řízení jejich spolupráce, a proto nárůst efektivitu celkového systému není tak výrazný. Často výhodnější možností je paralelní propojení jednodušších funkčních bloků s rozprostřenou pamětí. Koordinace jejich vzájemné spolupráce je jednodušší, a tak je celková efektivita větší než v případě složitějších prvků.

Za příklad druhého způsobu, tj. paralelního propojení velkého počtu poměrně jednoduchých funkčních bloků, lze považovat i umělé neuronové sítě. Umělá neuronová síť je pole vzájemně propojených elementárních funkčních bloků se speciální architekturou a v mnoha případech nabízí velmi perspektivní řešení zpracování dat. Na základě neuronových buněk živých organizmů byla vytvořena řada různých typů neuronových sítí. Disertační práce je zaměřena na jeden speciální typ umělých neuronových sítí, na tzv. Hopfieldovy sítě, které je možné využít na řešení optimalizačních úloh.

Na základě architektury aktivních prvků a na základě úkolů, které tyto prvky realizují lze konstatovat, že pro jejich řízení by se daly velmi efektivně využít umělé neuronové sítě. Náplní disertační práce je proto hledat alternativní cestu pro zvýšení výkonnosti systémů, jejichž činnost je možné realizovat s využitím paralelní architektury. Navíc musí daný systém během své činnosti nalézt řešení optimalizačního problému. Tento záměr poměrně dobře charakterizuje požadavky na moderní aktivní síťové prvky.

2 CÍLE PRÁCE

Cílem disertační práce bylo navrhnout novou architekturu uzlového aktivního síťového prvku, která je dostatečně výkonná a podporuje prioritní zpracování různých typů spojení. Právě chybějící podpora prioritního zpracování je největším nedostatkem většiny současných aktivních síťových prvků. Navržená nová architektura je založena na technologii umělých neuronových sítí. Neuronová síť je využívána na optimalizaci přepojování vstupních požadavků na výstupy.

Nejdůležitějším záměrem práce je ukázat, že možnosti využití umělých neuronových sítí v komunikačních systémech, které lze realizovat pomocí paralelní architektury, jsou velmi rozsáhlé. Systém využívající umělé neuronové sítě by mohl být velmi efektivní a často by nabízel výhodnější řešení než klasické procesorově orientované systémy. Ze základní struktury neuronových sítí vyplývá, že se jedná o funkční jednotky vykonávající jednoduché operace, a proto pracovní kmitočet prvků realizující tyto funkční jednotky může být mnohem vyšší než např. pracovní kmitočet složitějšího procesoru. Dále se dá velmi efektivně využít paralelní zpracování informací, což přinese další zvýšení efektivity systému.

Konkrétním cílem disertační práce je provést podrobnou matematickou analýzu rekurentní Hopfieldovy neuronové sítě a odvodit podmínky, při kterých tato síť vykonává svoji funkci s největší účinností. Dále je potřebné zjistit, jakým způsobem lze transformovat matematický popis optimalizační úlohy na matematický popis neuronové sítě tak, aby vztahy určující parametry neuronů nebyly porušeny.

Po odvození matematických vztahů popisujících umělou neuronovou síť bude žádoucí platnost těchto vztahů ověřit i prakticky. Pro ověření činnosti navržených systémů bude využita softwarová simulace. Během tohoto testování bude cílem ověřit jednak funkčnost matematického modelu a dále bude nutné stanovit optimální hodnoty globálních parametrů neuronové sítě, jako např. maximální počet iteračních kroků atd.

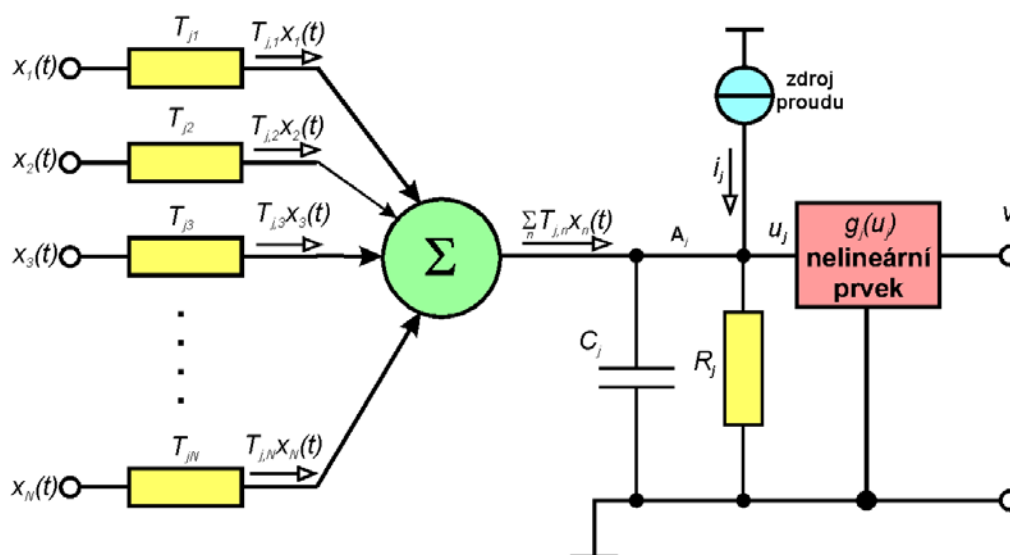
Po vytvoření potřebného teoretického zázemí bude zřejmě možné vytýčit další cíl a to definovat architekturu spojovacího zařízení, která bude schopna využít neuronovou síť pro řízení přepojování. Bude nutné stanovit základní funkční bloky systému a způsob jejich spolupráce.

3 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

3.1 HOPFIELDOVA NEURONOVÁ SÍŤ

J. J. Hopfieldem byla navržena umělá neuronová síť, která se vyznačuje speciální zpětnovazební strukturou. Tzv. Hopfieldova neuronová síť, prezentována v [1] jako spojitý systém a v [2] jako diskrétní systém, je jednovrstvová a výstupy jednotlivých neuronů jsou vedeny zpět na vstup ostatních neuronů. Důsledkem zpětnovazební struktury získáme novou vlastnost neuronové sítě, časovou závislost. Taková struktura umožňuje implementovat algoritmy, které jsou založeny na iteračních procesech. Nejčastějším využitím Hopfieldovy sítě je řešení optimalizačního problému pomocí iteračního procesu. Během řešení iteračního procesu se provádí minimalizace energetické funkce sítě. Proto řešení optimalizační úlohy lze realizovat vhodnou transformací optimalizační úlohy na energetickou funkci neuronové sítě.

Aditivní model neuronové sítě znázorněný na obr. 3.1 [20] vychází z fyzické realizace umělého neuronu pomocí diskrétních elektronických součástek. V tomto případě u j -tého neuronu jsou váhové koeficienty realizovány pomocí rezistorů s vodivostí (T_{jn}) a příslušná vstupní ($x_n(t)$), výstupní (v_j) a stavová veličina (u_j) jsou reprezentovány napětovou úrovní. Nelineární aktivační funkce je nejčastěji realizována pomocí operačního zesilovače, který udržuje velikost vstupního proudu prakticky na nulové hodnotě.



Obr. 3.1 Aditivní model j -tého neuronu zahrnující dynamické vlastnosti

V případě umělé neuronové sítě sestavené z umělých neuronů popsanych aditivním modelem dostaneme systém nelineárních rovnic prvního řádu, který odpovídá stavovému popisu systému

$$C_j \frac{du_j(t)}{dt} = -\frac{u_j(t)}{R_j} + \sum_{n=1}^N T_{j,n} x_n(t) + i_j, \quad j = 1, 2, \dots, N. \quad (3.1)$$

Po uvážení, že se jedná o rekurentní strukturu, tj. $x_n(t) = v_n(t)$ a po zjednodušení

$$C_1 = C_2 = \dots = C_N = C = 1, R_1 = R_2 = \dots = R_N = R = 1$$

a

$$g_j(u_n(t)) = g(u_n(t)) \text{ pro } j, n = 1, 2, \dots, N \quad (3.2)$$

získáme vztah popisující dynamické chování neuronové sítě

$$\frac{du_j(t)}{dt} = -u_j(t) + \sum_{n=1}^N T_{j,n} g(u_n(t)) + i_j, \quad j = 1, 2, \dots, N. \quad (3.3)$$

Pro výstupní veličiny bude platit vztah

$$v_j(t) = g(u_j(t)), \quad (3.4)$$

kde $g(u_j)$ je spojitá nelineární aktivační funkce neuronové sítě.

3.2 ZÁKLADNÍ MATEMATICKÝ POPIS HOPFIELDOVY SÍTĚ

Hopfieldova síť se nejčastěji popisuje pomocí stavových veličin a výstupních hodnot. V případě iteračního procesu výstupní data z kroku k budou využívána jako vstupní data v kroku $k+1$. U dvourozměrné neuronové sítě stavová veličina označená jako $U_{i,j}$ značí stav neuronu v řádku i a ve sloupci j při maticovém vyjádření. Celá matice stavových veličin se značí jako matice \mathbf{U} . Výstupní hodnota neuronu v i -tém řádku a v j -tém sloupci se značí $V_{i,j}$ a celá matice výstupních hodnot se značí jako matice \mathbf{V} . Vzájemný vztah mezi výstupní hodnotou a příslušnou stavovou veličinou je zpravidla dán neklesající, monotónní aktivační funkcí $g_{i,j}$. Obecně platí, že každý neuron může mít odlišnou aktivační funkci, prakticky se však používá stejná aktivační funkce pro každý neuron

$$V_{i,j} = g(U_{i,j}). \quad (3.5)$$

Při dvourozměrném uspořádání neuronů jsou váhové koeficienty neuronové sítě značeny proměnnou $T_{x,y,i,j}$ a vyjadřují váhu výstupu neuronu z i -tého řádku a j -tého sloupce na vstup neuronu v x -tém řádku a y -tém sloupci. Pro označení matice váhových koeficientů se bude používat matice \mathbf{T} . Prahová hodnota neuronu v i -tém řádku a j -tém sloupci bude značena $I_{i,j}$ a matice těchto prahových hodnot bude označena \mathbf{I} . V případě jednorozměrného uspořádání neuronů jsou proměnné $U_{i,j}$, $V_{i,j}$, $T_{x,y,i,j}$ nahrazeny proměnnými u_i , v_i , $T_{i,j}$, a maticové vyjádření \mathbf{U} , \mathbf{V} , \mathbf{I} je nahrazeno vektory \mathbf{u} , \mathbf{v} , \mathbf{i} .

3.2.1 Energetická funkce klasické Hopfieldovy neuronové sítě

Popis klasické Hopfieldovy neuronové sítě, která je vhodná pro řešení optimalizačních problémů byl prezentován J.J. Hopfieldem a D. W. Tankem v článku [3]. Během činnosti Hopfieldovy sítě prochází stavové veličiny sítě stavovým prostorem po určité trajektorii. Přitom dochází ke snižování vnitřní energie systému, což popisuje energetická funkce sítě.

Energetická funkce analogové Hopfieldovy sítě je vyjádřena vztahem

$$E = -\frac{1}{2} \sum_i \sum_j T_{i,j} v_i v_j - \sum_i i_i v_i + \frac{1}{\tau} \sum_i \int_0^{v_i} g_i^{-1}(\alpha) d\alpha, \quad (3.6)$$

Proměnná τ je časovou konstantou a její hodnota nemá vliv na ustálený stav neuronové sítě. Dokonce naprostá většina aplikací předpokládá, že $\frac{1}{\tau} = 0$, což má za následek, že třetí člen ve vztahu (3.6) se vůbec neuvažuje. Tak získáme vztah vyjadřující obecnou energetickou funkci ve tvaru

$$E = -\frac{1}{2} \sum_i \sum_j T_{i,j} v_i v_j - \sum_i i_i v_i. \quad (3.7)$$

3.3 OPTIMALIZAČNÍ PROBLÉMY

Značná skupina optimalizačních úloh patří do skupiny tzv. kombinatorních optimalizačních úloh. Během řešení kombinatorního optimalizačního problému se hledá nejlepší řešení z několika možných. Omezující kritéria specifikují základní omezení problému a pouze vymezují množinu platných řešení. Po určení ohodnocení jednotlivých platných řešení, stanovené účelovou funkcí, budou některá řešení výhodnější a jiná méně výhodná. Cílem optimalizace je najít nejlepší, optimální řešení, kdy účelová funkce dosáhne svého extrému.

Existuje speciální skupina optimalizačních problémů, u kterých neexistuje žádný známý algoritmus, který by umožnil algoritmizovat hledání výsledku. Jediným možným způsobem nalezení optimálního řešení je vyzkoušet všechny platné kombinace, pro každou kombinaci vypočítat účelovou funkci a vybrat kombinaci, při které účelová funkce dosáhne svůj extrém. Největší nevýhodou těchto optimalizačních problémů je, že počet platných řešení roste exponenciálně s nárůstem velikosti problému. Z toho vyplývá, že i čas potřebný k nalezení optimálního řešení narůstá exponenciálně.

3.4 MATEMATICKÉ VYJÁDŘENÍ KOMBINATORNÍCH OPTIMALIZAČNÍCH ÚLOH

Při řešení optimalizačního problému hledáme N -rozměrný vektor

$$\mathbf{v} = [v_1, v_2, \dots, v_N]^T \quad (3.8)$$
$$v_n \geq 0 \text{ pro všechna } n \in \langle 1; N \rangle$$

splňující M lineárních omezení

$$\begin{aligned} A_{1,1}v_1 + A_{1,2}v_2 + \dots + A_{1,N}v_N &= b_1, \\ A_{2,1}v_1 + A_{2,2}v_2 + \dots + A_{2,N}v_N &= b_2, \\ &\dots \\ A_{M,1}v_1 + A_{M,2}v_2 + \dots + A_{M,N}v_N &= b_M, \end{aligned} \quad (3.9)$$

kde $m = 1, 2, \dots, M$, a optimalizující účelovou funkci $f(\mathbf{v})$. Systém lineárních rovnic (3.9) vyjadřujících lineární omezení lze přepsat do vektorové formy podle

$$\mathbf{A} \cdot \mathbf{v} = \mathbf{b}. \quad (3.10)$$

Vztahy (3.9) či (3.10) vyjadřují pouze omezující kritéria optimalizačního problému. Účelovou funkci $f(\mathbf{v})$ pro velkou skupinu optimalizačních úloh lze vyjádřit vztahem

$$f(\mathbf{v}) = \mathbf{c}^T \mathbf{v} + \mathbf{v}^T \mathbf{Q} \mathbf{v}. \quad (3.11)$$

kde vektor \mathbf{c} vyjadřuje váhu přiřazenou příslušnému prvku vektoru \mathbf{v} v případě, že daný prvek je zvolen, tj. má hodnotu 1. Člen $\mathbf{c}^T \mathbf{v}$ tak vyjadřuje vážený součet příspěvků od všech takto vybraných prvků. V tomto případě se jedná o lineární část účelové funkce. Matice \mathbf{Q} , podobně jako vektor \mathbf{c} , obsahuje váhové koeficienty. V daném případě se však jedná o váhové koeficienty, které jsou přiřazeny stavu, kdy jsou zvoleny patřičné dvojice prvků. Proto člen $\mathbf{v}^T \mathbf{Q} \mathbf{v}$ ve vztahu (3.11) udává vážený součet příspěvků od všech dvojic prvků. Jedná se o kvadratickou část účelové funkce. Optimálním řešením bude, když součet obou členů účelové funkce $f(\mathbf{v})$ podle (3.11) bude minimální či maximální v závislosti na optimalizační úloze.

Společným rysem umělé neuronové sítě a optimalizačních problémů je analogie mezi energetickou funkcí a účelovou funkcí. Energetická funkce neuronové sítě může být dokonce ekvivalentní s účelovou funkcí optimalizačního problému, který má být řešen a omezující kritéria optimalizačního problému lze považovat za prohibiční členy energetické funkce. Podle [5] lze energetickou funkci $E(\mathbf{v})$ rozepsat na dvě dílčí komponenty následujícím způsobem

$$E(\mathbf{v}) = f(\mathbf{v}) + \frac{\gamma}{2} E_{\text{cn}}(\mathbf{v}). \quad (3.12)$$

kde člen $f(\mathbf{v})$ odpovídá účelové funkci podle (3.11). Omezující člen E_{cn} je zodpovědný za dodržení omezujících kritérií. V případě správného výsledku je člen E_{cn} nulový a člen $f(\mathbf{v})$ udává minimum účelové funkce. Parametr γ řídí vzájemný poměr těchto dvou členů. Protože primárním požadavkem je, aby byla splněna omezující kritéria a teprve dalším požadavkem je, aby získané řešení bylo optimální, je γ zpravidla větší než 2. Výhodou takto vyjádřené energetické funkce je, že je závislá jen na jediném parametru γ a volbou dostatečně velké hodnoty γ lze zajistit, aby řešení bylo platné.

4 ZVOLENÉ METODY ZPRACOVÁNÍ

Nejvýznamnějším zdokonalením klasické Hopfieldovy sítě bylo zkvalitnění transformace optimalizační úlohy na energetickou funkci. U klasické Hopfieldovy sítě energetická funkce zajistila jak splnění optimalizačních kritérií, tak i minimalizaci účelové funkce. Takto vznikla složitá energetická funkce, jejíž řešení často uvízlo v lokálním minimu. Vylepšená tvorba energetické funkce spočívá v tom, že je stanoven podprostor, který obsahuje všechna platná řešení splňující omezující kritéria optimalizačního problému. Vektor stavových veličin se v každém iteračním kroku promítne do tohoto podprostoru a dále se pracuje pouze s tímto průmětem. Tak je zajištěno, že optimalizační kritéria budou stále splněna. Proto bude stačit, když energetická funkce bude zvolena pouze na základě účelové funkce optimalizačního problému. Takto zjednodušená energetická funkce zajišťuje mnohem lepší konvergenci k optimálnímu řešení.

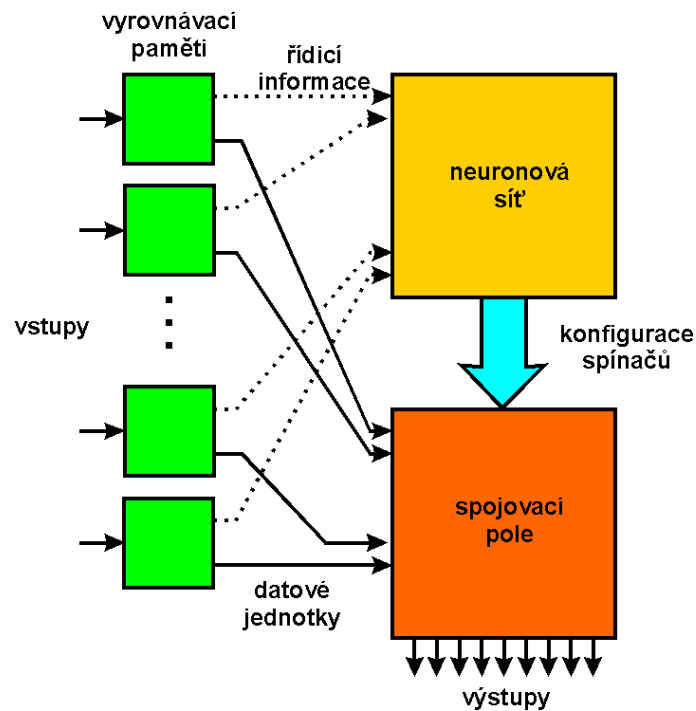
4.1 OPTIMALIZACE PŘEPÍNÁNÍ

Funkce prepínače spočívá v tom, že data přicházející na vstupy musí být převedena na cílový výstup a to co nejrychleji. Rychlost může být omezena blokováním, což je způsobeno situací, kdy datové toky z více vstupů směřují na jeden výstup. Moderní multimediální aplikace často potřebují mechanismy pro řízení spravedlivého či přednostního přidělování komunikačního kanálu. Společnou vlastností těchto řešení je, že se předpokládá identifikace jednotlivých či skupin datových toků a přiřazení priority, na základě které se provádí proces prepínání.

4.1.1 Model prepínače využívající umělou neuronovou síť

Základní model uzlového aktivního síťového prvku se skládá z několika dílčích bloků. Blokové schéma aktivního prvku je znázorněno na obr. 4.1. Přijaté datové jednotky jsou nejdříve uloženy do vstupní vyrovnávací paměti. Protože datové jednotky obsahují cílovou adresu, je možné určit, přes který výstupní port opustí spojovací systém. Kromě informace o cílovém portu se zjišťují také informace o prioritě datového toku, ke kterému datová jednotka patří.

Uvedené informace jsou shromážděny ze všech datových jednotek uchovaných ve vstupních vyrovnávacích pamětech a slouží jako vstupní údaje pro neuronovou síť. Lze sestavit N -rozměrný vektor, tvořený z priorit jednotlivých požadavků. První člen vektoru obsahuje prioritu rámce směřujícího na první výstup, druhý člen vektoru obsahuje prioritu rámce směřujícího na druhý výstup atd.



Obr. 4.1 Blokové schéma spojovacího pole využívající neuronovou síť

Z vektorů priorit získaných od jednotlivých vstupních portů lze vytvořit $N \times N$ – rozměrnou matici, která bude obsahovat vstupní podmínky, které je třeba optimalizovat. Jako odpověď na tyto vstupní údaje neuronová síť vygeneruje určitou konfiguraci spínačů, na základě které budou nastaveny spínače ve spojovacím poli.

Vygenerovaná konfigurace bude ve tvaru $N \times N$ – rozměrné matice, tzv. konfigurační matice. Požadavkem na vygenerovaný výsledek je, aby byly propojeny vždy jeden vstup a jeden výstup. Tato podmínka se promítne do konfigurační matice vygenerované neuronovou sítí tak, že v každém řádku a v každém sloupci bude právě jeden aktivní výstup, tj. hodnota 1 a ostatní budou neaktivní, tj. budou mít hodnotu 0. Po nastavení spínačů může dojít k vlastnímu přenosu datových jednotek ze vstupních vyrovnávacích pamětí na výstupy systému.

Nastavení spínačů má být optimální z hlediska matice priorit. Vybere se konfigurace tak, aby v každém sloupci a v každém řádku byl označen právě jeden prvek a součet takto vybraných priorit byl minimální za předpokladu, že menší hodnota odpovídá vyšší prioritě.

Účelovou funkci optimalizační úlohy prioritního přepínání lze analyticky vyjádřit redukováným vztahem (3.11), kde bude chybět člen s maticí Q . V případě minimalizace součtu vybraných priorit nás zajímají jen vybrané neurony a nikoliv vzájemný vliv dvou vybraných priorit. Proto účelovou funkci lze psát ve tvaru

$$f(\mathbf{v}) = \mathbf{c}^T \mathbf{v} . \quad (4.1)$$

4.2 PODPROSTOR PLATNÝCH ŘEŠENÍ

4.2.1 Definice vztahů pro pomocné vektory a matice

Pro potřeby následující analýzy bude definován pomocný vektor ${}^N\mathbf{o}$ a pomocná matice ${}^N\mathbf{O}$ a ${}^N\mathbf{R}$. Vektor ${}^N\mathbf{o}$ má rozměr N a každý jeho prvek má hodnotu jedna.

$${}^N o_n = 1 \text{ pro } n \in \{1, \dots, N\}. \quad (4.2)$$

Obdobně matice ${}^N\mathbf{O}$ má rozměr $N \times N$ a každý její prvek má hodnotu 1

$${}^N O_{i,j} = 1 \text{ pro } i, j \in \{1, \dots, N\}. \quad (4.3)$$

Prvky matice ${}^N\mathbf{R}$ jsou definovány vztahem

$${}^N R_{i,j} = \delta_{i,j} - \frac{1}{N}, \quad (4.4)$$

kde δ_{ij} je definována vztahem

$$\delta_{ij} = \begin{cases} 1 & \text{pro } i = j, \\ 0 & \text{pro } i \neq j. \end{cases} \quad (4.5)$$

Charakteristickou vlastností matice ${}^N\mathbf{R}$ je, že když se matice ${}^N\mathbf{R}$ použije jako transformační matice, pak z původního vektoru odstraní složku ve směru ${}^N\mathbf{o}$ a přitom složky kolmé na směr ${}^N\mathbf{o}$ neovlivní. Další zajímavou vlastností matice ${}^N\mathbf{R}$ je, že v případě transformace $\mathbf{y} = {}^N\mathbf{R} \cdot \mathbf{x}$ pro libovolný vektor \mathbf{x} bude platit, že $\sum_{n=1}^N y_n = 0$, tj. součet všech prvků vektoru \mathbf{y} bude nulový

$$\sum_{n=1}^N y_n = \sum_{n=1}^N [{}^N\mathbf{R} \cdot \mathbf{x}]_n = 0. \quad (4.6)$$

Důkaz platnosti uvedených závislostí lze najít v publikaci [13].

4.2.2 Algebraická reprezentace vektoru výstupních veličin

Na řešení optimalizačního problému prioritního přepínání lze pohlížet také jako na stanovení trajektorie ve vícerozměrném poli od výchozího stavu k ustálenému stavu, kde ustálený stav splňuje určitá omezení. Zkusme si představit, že v N^2 – rozměrném prostoru je umístěna do počátku souřadnicového systému hyperkrychle s jednotkovou délkou hran. Souřadnice každého rohu této hyperkrychle se dají vyjádřit pomocí jedniček a nul.

Za těchto předpokladů si výsledek optimalizační úlohy, tj. vektor \mathbf{v} vyjádřený jako sled jedniček a nul, můžeme představit jako jeden z rohů hyperkrychle. Potom procesu konvergence neuronové sítě odpovídá trajektorie uvnitř v N^2 - rozměrného prostoru, která končí v některém z rohů, jehož souřadnice (vektor \mathbf{v}) splňuje podmínku (3.9).

Možným řešením omezující části energetické funkce E_{cn} může být množina rohů, jejichž souřadnice přeepsané do maticového tvaru pomocí operace inverzní k operaci *vec* (kapitola 4.2.3) splňují omezující kritéria. Podle [13] lze dokázat, že takto vybrané rohy budou ležet v jednom podprostoru výchozího N^2 - rozměrného prostoru. Tento podprostor bude nazván podprostorem platných řešení. Můžeme dále odvodit, že existuje zobrazení, které libovolný vektor \mathbf{v} v prostoru promítne do tohoto podprostoru.

4.2.3 Podprostor nulového součtu

Z vektoru výstupních veličin \mathbf{v} lze vytvořit matici \mathbf{V} mezi nimiž platí vztah

$$\mathbf{v} = \text{vec}(\mathbf{V}) \quad (4.7)$$

kde maticová operace *vec* je definována podle vztahu

$$\mathbf{f} = \text{vec}(\mathbf{F}) = [\mathbf{F}_{11}, \mathbf{F}_{21}, \dots, \mathbf{F}_{M1}, \mathbf{F}_{12}, \mathbf{F}_{22}, \dots, \mathbf{F}_{M2}, \dots, \mathbf{F}_{1N}, \mathbf{F}_{2N}, \dots, \mathbf{F}_{MN}]^T \quad (4.8)$$

přičemž matice \mathbf{F} má rozměry $M \times N$. V ustáleném stavu je matice \mathbf{V} permutační maticí a tak součet prvků ve všech řádcích a sloupcích matice bude dávat hodnotu 1. Vektor \mathbf{v} proto bude vhodné rozdělit do dvou komponent. První komponenta bude zajišťovat, aby součty prvků ve všech řádcích a ve všech sloupcích měly hodnotu 1. Druhá komponenta bude zajišťovat omezující kritéria a dále součet prvků ve všech řádcích a sloupcích bude mít hodnotu 0. Znamená to, že druhá komponenta bude ležet v tzv. podprostoru nulového součtu, který bude značen indexem z_s (zerosum). Podprostor nulového součtu je tvořen všemi vektory \mathbf{v} , získanými využitím vztahu (4.7) z matice \mathbf{V} , která splňuje podmínku, že součet prvků v jednotlivých řádcích a v jednotlivých sloupcích bude nulový. Nechť je vektor \mathbf{v}_{z_s} definován jako průmět vektoru \mathbf{v} do podprostoru nulového součtu realizovaný transformační maticí \mathbf{T}_{z_s} podle

$$\mathbf{v}_{z_s} = \mathbf{T}_{z_s} \cdot \mathbf{v}, \quad (4.9)$$

kde \mathbf{T}_{z_s} je definován závislostí

$$\mathbf{T}_{z_s} = \left({}^N \mathbf{R} \otimes {}^N \mathbf{R} \right). \quad (4.10)$$

Operace \otimes značí tenzorové násobení a je popsána např. v lit. [17]. V případě, že vektor \mathbf{v}_{z_s} převedeme na matici \mathbf{V}^{z_s} pomocí vztahu

$$\mathbf{v}_{zs} = \text{vec}(\mathbf{V}_{zs}^T), \quad (4.11)$$

lze dokázat, že součty prvků matice \mathbf{V}_{zs} v jednotlivých sloupcích a řádcích budou nulové.

4.2.4 Obecný popis podprostoru platných řešení

Předpokládejme rovnici podle vztahu

$$\mathbf{V} = {}^N\mathbf{R} \mathbf{V} {}^N\mathbf{R} + {}^N\mathbf{o} \mathbf{a}^T = \mathbf{V}_{zs} + {}^N\mathbf{o} \mathbf{a}^T. \quad (4.12)$$

V případě, že matice \mathbf{V} splňuje vztah (4.12), pak součty členů v jednotlivých řádcích a sloupcích matice \mathbf{V} budou dány jedinečně členem ${}^N\mathbf{o} \mathbf{a}^T$, protože člen ${}^N\mathbf{R} \mathbf{V} {}^N\mathbf{R}$ bude mít nulové součty prvků v řádcích i ve sloupcích. Vektor \mathbf{a} je definován závislostí

$$\mathbf{a} = \frac{1}{N} {}^N\mathbf{o}. \quad (4.13)$$

Maticový popis (4.12) lze převést na vektorový popis následujícím způsobem

$$\mathbf{V}^T = {}^N\mathbf{R} \mathbf{V}^T {}^N\mathbf{R} + \mathbf{a} \mathbf{o}^T \quad (4.14)$$

a následně

$$\text{vec}(\mathbf{V}^T) = ({}^N\mathbf{R} \otimes {}^N\mathbf{R}) \text{vec}(\mathbf{V}^T) + ({}^N\mathbf{o} \otimes \mathbf{a}) \text{vec}(\mathbf{1}). \quad (4.15)$$

Vztah (4.45) lze přepsat na obecný vztah podprostoru platných řešení

$$\mathbf{v} = \mathbf{T}_{zs} \mathbf{v} + \mathbf{s}, \quad (4.16)$$

kde matice \mathbf{T}_{zs} je definována podle (4.10) a vektor \mathbf{s} je definován vztahem

$$\mathbf{s} = \frac{1}{N} ({}^N\mathbf{o} \otimes {}^N\mathbf{o}). \quad (4.17)$$

Základní vlastností podprostoru platných řešení je, že součet prvků v jednotlivých sloupcích a řádcích je závislý jen na členu \mathbf{s} . Proto lze vhodnou volbou vektoru \mathbf{s} zajistit, aby do podprostoru správného řešení byly zahrnuty jen ty rohy hyperkrychle, které splňují podmínky optimalizačního problému.

Výsledná rovnice podprostoru platných řešení problému prioritního přepínání bude dána vztahem

$$\mathbf{v} = ({}^N\mathbf{R} \otimes {}^N\mathbf{R}) \mathbf{v} + \frac{1}{N} ({}^N\mathbf{o} \otimes {}^N\mathbf{o}). \quad (4.18)$$

Další charakteristickou vlastností vztahu (4.18) je, že jeho dvě komponenty jsou vzájemně kolmé. Proto změna jedné komponenty nemá vliv na druhou komponentu. Dalším důležitým výsledkem je, že pro libovolný $N.N$ rozměrný vektor \mathbf{v} můžeme definovat jeho průmět do podprostoru platných řešení \mathbf{v}_{cn} podle vztahu

$$\mathbf{v}_{\text{cn}} = \left({}^N \mathbf{R} \otimes {}^N \mathbf{R} \right) \mathbf{v} + \frac{1}{N} \left({}^N \mathbf{o} \otimes {}^N \mathbf{o} \right). \quad (4.19)$$

Cílem iteračního procesu potom bude, aby zajistil konvergenci \mathbf{v}_{cn} do vhodného rohu hyperkrychle představující optimální řešení a ležící v podprostoru platných řešení.

4.2.5 Transformace do podprostoru platných řešení

Každé platné řešení optimalizačního problému leží v podprostoru platných řešení. Proto z vektoru \mathbf{v} pro nás bude důležitá komponenta v tomto podprostoru. Znamená to, že potřebujeme vyjádřit průmět vektoru \mathbf{v} do tohoto podprostoru. Tento průmět získáme lineární transformací

$$\mathbf{v} \leftarrow \mathbf{T}_{\text{zs}} \cdot \mathbf{v} + \mathbf{s} \quad (4.20)$$

Podle literatury [18] jsou matice \mathbf{T}_{zs} a vektor \mathbf{s} definovány vztahy

$$\mathbf{T}_{\text{zs}} = \mathbf{I} - \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{A}, \quad (4.21)$$

$$\mathbf{s} = \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{b}. \quad (4.22)$$

Matice \mathbf{A} a vektor \mathbf{b} již byly definovány v kapitole 3.4. Matice \mathbf{I} je jednotkovou maticí, tj. všechny členy kromě diagonálních jsou nulové a diagonální členy jsou rovny 1. Po dosazení vztahů (4.21) a (4.22) do vztahu (4.1) lze odvodit výraz

$$E(\mathbf{v}) = -\frac{1}{2} \mathbf{v}^T [\gamma (\mathbf{T}_{\text{zs}} - \mathbf{I})] \mathbf{v} - \mathbf{v}^T (\gamma \mathbf{s} - \mathbf{c}) + \frac{\gamma}{2} \mathbf{s}^T \mathbf{s}. \quad (4.23)$$

kde vektor \mathbf{c} byl uveden ve vztahu (3.11). Po porovnání vztahu (4.23) se vztahem (4.16) vyjadřující obecný tvar energetické funkce, lze odvodit váhové koeficienty \mathbf{T} a prahovou hodnotu \mathbf{i} pro náš případ následovně:

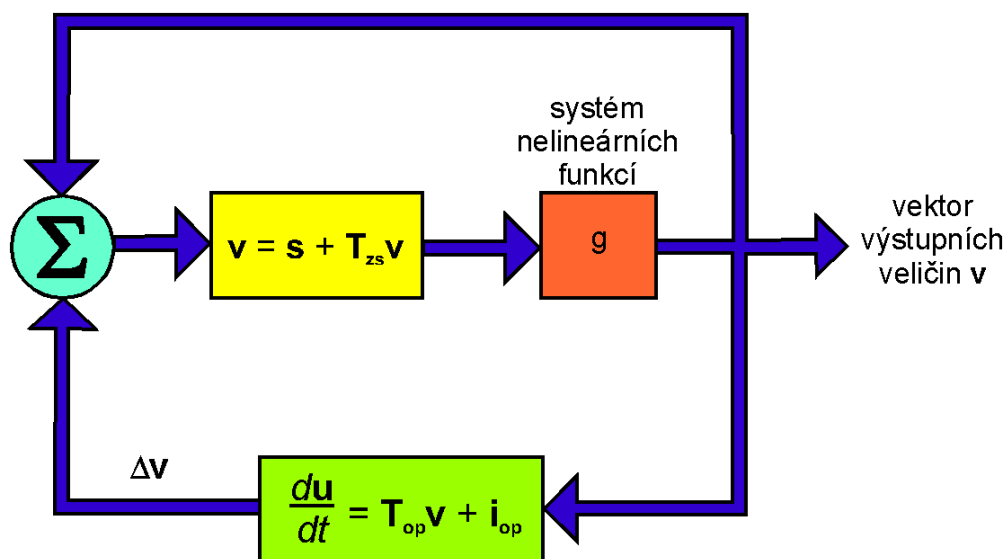
$$\mathbf{T} = \gamma (\mathbf{T}_{\text{zs}} - \mathbf{I}), \quad (4.24)$$

$$\mathbf{i} = \gamma \mathbf{s} - \mathbf{c}. \quad (4.25)$$

5 HLAVNÍ VÝSLEDKY PRÁCE

5.1 PRAKTICKÁ IMPLEMENTACE HOPFIELDOVY SÍTĚ PRO ŘEŠENÍ PROBLÉMU PRIORITYNÍHO PŘEPÍNÁNÍ

Pro skutečné ověření funkčnosti navrženého matematického modelu byl vytvořen simulační model. Jak je vidět na obr. 5.1, iterační proces lze rozdělit do dvou smyček. Horní smyčka zajišťuje, aby vektor \mathbf{v} zůstal v podprostoru platných řešení a spodní smyčka zajišťuje řešení optimalizačního problému. Po každém kroku provedeném pomocí spodní smyčky je nutné provést několik průchodů horní smyčkou. To je potřebné proto, že výsledkem operace spodní smyčky se vektor \mathbf{v} dostane s největší pravděpodobností mimo podprostor platných řešení. První operace z vektoru \mathbf{v} zanechá jen průmět do podprostoru platných řešení. Tato operace ale nezaručuje, že průmět bude uvnitř hyperkrychle s jednotkovou délkou hran. Tuto podmínku zajistí druhá operace horní smyčky pomocí nelineární aktivační funkce. Výsledkem této operace se však vektor \mathbf{v} zase může dostat mimo podprostor platných řešení.



Obr. 5.1 Grafické znázornění iteračního procesu Hopfieldovy sítě s vylepšenou energetickou funkcí

Proto je nutné znovu provést průmět do tohoto podprostoru a zajistit, aby výsledek zůstal uprostřed hyperkrychle atd. Teoreticky se jedná o nekonečný proces, ale prakticky stačí, když bude splněna podmínka

$$\left| \sum_{n=1}^{N^2} v_n - N \right| < 0.1N. \quad (5.1)$$

5.2 SIMULAČNÍ PROGRAM PRO UMĚLÉ NEURONOVÉ SÍTĚ

Pro ověření výsledků teoretické práce bylo nezbytné použít vhodný simulační nástroj. Z důvodu nedostupnosti vhodného komerčního či volně šiřitelného simulačního nástroje jsem byl nucen vyvinout vlastní simulační prostředí.

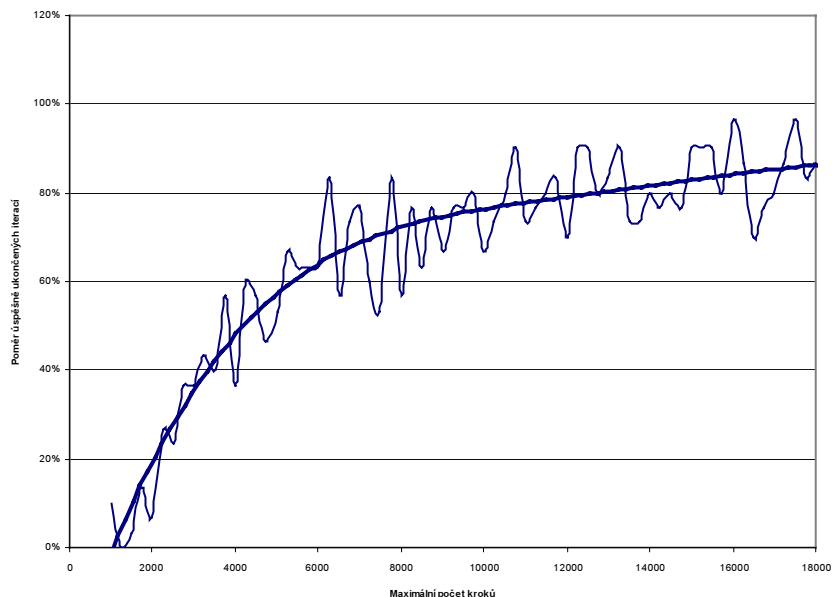
Výsledkem této činnosti je program, který je schopen provádět simulaci různých typů Hopfieldovy neuronové sítě. Umožňuje jednoduše zadávat vstupní podmínky a zobrazit nejen výsledky optimalizačního procesu, ale i průběh celého optimalizačního procesu. Vstupní data pro neuronovou síť lze automaticky vygenerovat nebo ručně nastavit. V případě potřeby umožňuje simulační prostředí i krokování iteračního procesu. Mimo to obsahuje také několik informačních oken pro zobrazení různých parametrů testované sítě, a to jak v číselné tak i v grafické podobě. Dále umožňuje ověřit správnost výsledku generovaného neuronovou sítí.

5.3 VÝSLEDKY TESTŮ

Následující testy byly provedeny s parametrem $N=10$.

Vliv maximálního počtu iteračních kroků

Tento test byl zaměřen na zjištění závislosti úspěšnosti sítě na maximálním počtu iteračních kroků (*MaxStepNumber*).

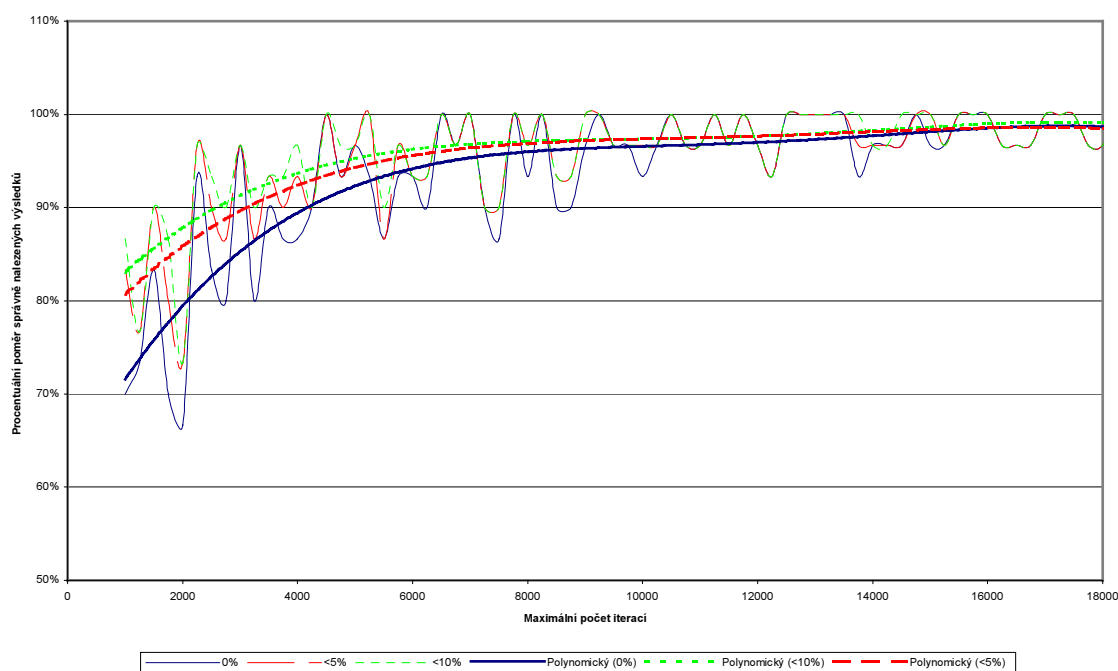


Obr. 5.2 Závislost úspěšně ukončených iterací na maximálním počtu iteračních kroků

Parametr *MaxStepNumber* byl testován v rozsahu 1 až 18000 po krocích 250. Pro každou hodnotu *MaxStepNumber* bylo provedeno 30 měření. Průběh na obr. 5.2

znázorňuje závislost úspěšně ukončených iterací na maximálním počtu iteračních kroků. Iterace byla považována za úspěšně ukončenou, když omezující kritéria byla splněna před dosažením maximálního počtu iteračních kroků. Závislost vyznačená tenčí čarou je vytvořena na základě skutečně získaných hodnot a závislost vyznačená tlustou čarou je polynomiální aproximace vytvořená ze získaných hodnot. Z grafu je vidět, že s nárůstem hodnoty *MaxStepNumber* narůstá i počet úspěšně ukončených iterací, ale tento nárůst se značně zmírní po dosažení *MaxStepNumber* > 6000. Znamená to, že zvyšování hodnoty *MaxStepNumber* po dosažení určité mezní hodnoty již není tak účinné.

Obr. 5.2 se týká ukončení iteračního procesu, ale nepodává žádnou informaci o kvalitě řešení. Pod pojmem kvalita řešení se rozumí shoda výsledku generovaného neuronovou sítí a skutečného optimálního řešení. Během testování byla proto hodnocena procentuální odchylka od skutečné optimální hodnoty v závislosti na hodnotě *MaxStepNumber*. Na následujícím obrázku 5.3 jsou vyznačeny tři závislosti, první vyjadřuje procentuální počet měření, kdy byla dosažena úplná shoda mezi generovaným a skutečně optimálním řešením, druhá vyjadřuje procentuální počet, kdy relativní odchylka byla menší než 5% a třetí vyjadřuje případ, kdy relativní odchylka byla menší než 10%. Ke každé závislosti experimentálně získaných hodnot je uvedena i příslušná polynomiální aproximace.

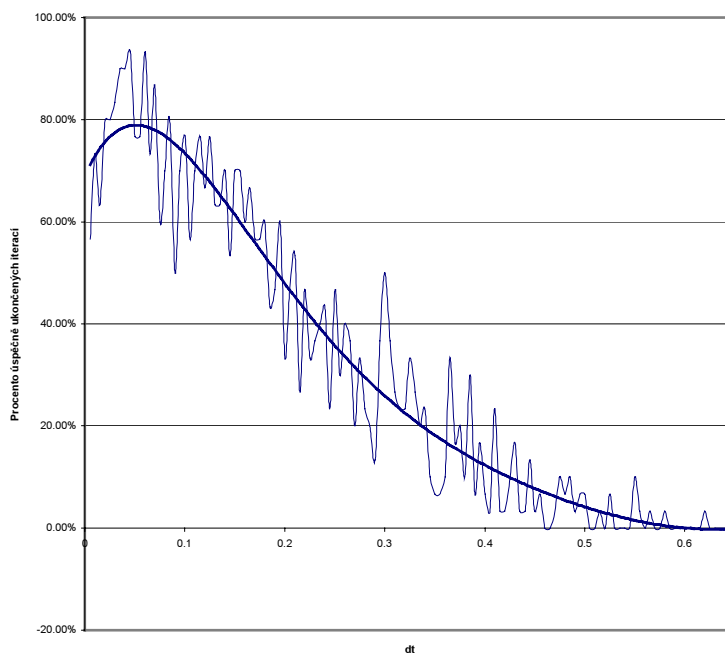


Obr. 5.3 Závislost kvality řešení na maximálním počtu iteračních kroků

Jak je vidět z porovnání průběhů na obr. 5.3, nárůstem hodnoty *MaxStepNumber* se zlepšuje i kvalita řešení, ale toto zlepšení se značně zmírní po dosažení určité mezní hodnoty *MaxStepNumber*. Tato mezní hodnota je kolem 5000.

Vliv hodnoty časového přírůstku dt

Hopfieldova síť při hledání optimálního výsledku řeší systém diferenciálních rovnic. Rovnice jsou řešeny Eulerovou metodou s konečnou hodnotou časového přírůstku dt . Protože se jedná o parametr, který se využívá během celého iteračního procesu, je žádoucí prozkoumat i jeho vliv na trvání iteračního procesu a na kvalitu získaných výsledků.



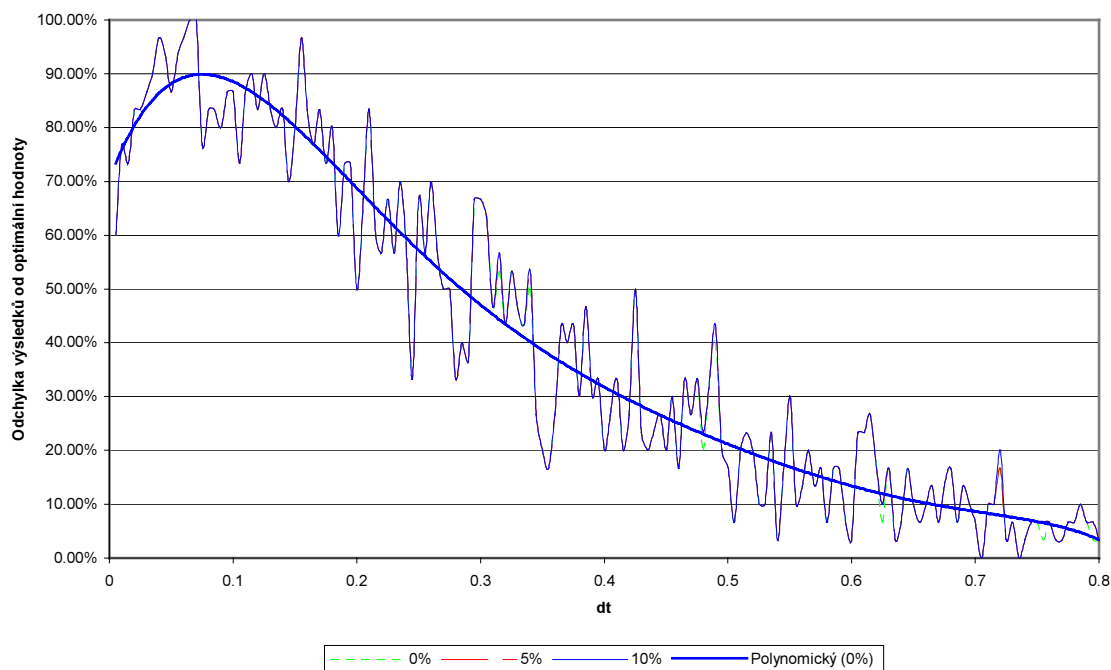
Obr. 5.4 Závislost kvality řešení na parametru $delta t$

Parametr dt byl testován v rozsahu 0,0001 až 0,8 po krocích 0,005. Pro každou hodnotu dt bylo provedeno 30 měření. Výsledky testování jsou uvedeny na obr. 5.4 a obr. 5.5. Graf na obr. 5.4 znázorňuje vliv parametru dt na procentuální počet úspěšně dokončených iterací. Testování bylo provedeno až do hodnoty $dt = 0,8$, ale pro větší názornost byla do grafu vynesena jen část průběhu. Nezobrazená část byla prakticky nulová.

Jak je vidět ze závislosti na obr. 5.4, existují určité hranice, mezi kterými je vhodné zvolit hodnotu parametru dt . Podle výsledků experimentů jsou tyto hranice v intervalu (0,01; 0,04). Příliš malá hodnota parametru dt zbytečně zvyšuje počet iteračních kroků. Na druhé straně vlivem příliš velké hodnoty parametru dt se neuronová síť stane nestabilní a přestane konvergovat k platnému řešení.

Graf na obr. 5.4 sice vyjadřuje závislost mezi relativním počtem úspěšně ukončených iterací a parametrem dt , ale neobsahuje informace o kvalitě generovaných řešení. Výsledky analýzy kvality řešení jsou vidět na obr. 5.5, kde jsou zobrazeny tři průběhy. První průběh vyjadřující procento výsledků, kdy byla dosažena úplná shoda mezi generovaným a skutečně optimálním řešením, druhý

vyjadřuje procento případů, kdy relativní odchylka byla menší než 5% a třetí vyjadřuje procento případů, kdy relativní odchylka byla menší než 10%. Všechny tři případy mají prakticky stejný průběh, a proto se na obrázku vzájemně překrývají. Lze tak říct, že parametr dt prakticky nemá vliv na kvalitu úspěšně ukončených iterací, ale výrazně může ovlivnit průběh iteračního procesu.



Obr. 5.5 Závislost kvality řešení na parametru dt

Ze závislosti kvality řešení na velikosti parametru dt vyplývá, že existuje optimální hodnota $dt = 0,08$, při které síť dosahuje největší procentuální počet správně nalezených výsledků. Při této hodnotě se poměr správně nalezených výsledků pohybuje kolem 90%. Protože průběhy závislosti procentuálního počtu úspěšně ukončených iterací a průběhy kvality úspěšně ukončených iterací dosahují svého maxima při odlišné velikosti parametru dt , konečná volba parametru dt bude závislá na tom, která ze dvou vlastností bude pro konkrétní optimalizační problém důležitější.

5.4 MOŽNOSTI IMPLEMENTACE VÝSLEDKŮ PRÁCE V AKTIVNÍCH PRVCÍCH

Protože aktivní prvky linkové a síťové vrstvy provádí analýzu adres, jsou tato zařízení schopna identifikovat jednotlivá spojení. Proto je možné u těchto zařízení zavést rozdělení spojení podle přiřazených priorit. Linková a fyzická vrstva se liší jen v tom, na které úrovni, tj. jak jemně, jsou schopny rozlišit jednotlivá spojení.

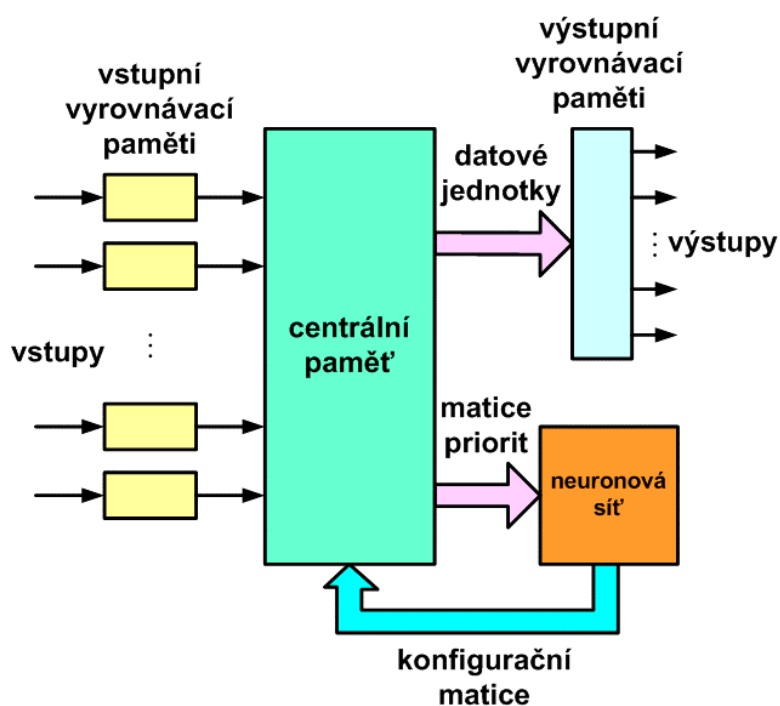
Prvky ATM sítě již obsahují dělení spojení do různých tříd, ale u aktivních prvků datových sítí, hlavně v poslední době, je tato vlastnost velmi postrádána. Proto

výsledky disertační práce jsou využitelné právě u těchto zařízení. U zařízení ATM sítě se tak jedná o alternativní způsob řešení přepínání, na rozdíl od způsobů využívaných v současnosti. U ostatních aktivních prvků se pak jedná o návrh, jak implementovat zpracování spojení s odlišnými prioritami. Je důležité zdůraznit, že nestačí jen doplnit aktivní prvky o podporu, ale také komunikační protokoly musí podporovat tuto novou vlastnost. Největší problémy jsou s implementací v sítích Ethernet, kde ani samotné hlavičky Ethernet rámců neumožňují zavedení priorit, protože neobsahují vhodnou položku pro tyto údaje. Možným řešením by bylo využití technologie MPLS (MultiProtocol Label Switching), která přidává návěští k rámcům vstupujícím do sítě a přepínání se dále provádí na základě tohoto návěští. MPLS je novou technologií, která se v současnosti začíná prosazovat.

V případě využití priorit při směrování je situace jednodušší, protože např. hlavička IP datagramu obsahuje položku pro prioritu. Při vývoji nové verze protokolu IP, verze 6, byla podpora prioritního systému značně propracována a doplněna. Protože podpora priorit na síťové vrstvě je v pokročilejším stavu než podpora na vrstvě linkové, nabízí se možnost využití výsledků práce i na této úrovni. Na druhé straně je navržený algoritmus vhodnější pro hardwarovou než pro softwarovou implementaci.

5.5 ZEFEKTIVNĚNÍ ARCHITEKTURY PŘEPÍNAČE

Architektura přepínače na obr. 4.1 přehledně znázorňovala komponenty systému a vzájemné návaznosti mezi těmito komponentami. V některých případech však může být tato architektura velmi neefektivní. Základní vlastností uvedeného spojovacího pole je, že umožňuje propojit současně jeden vstup s jedním výstupem a zpravidla přenést jednu datovou jednotku. V případě, že vyrovnávací paměť vybraného portu je prázdná, uskuteční se také propojení s vybraným výstupem, ale nedojde k přenosu datové jednotky. Tato architektura nebude efektivně pracovat např. v případě, kdy vstupní vyrovnávací paměť i -tého portu obsahuje dva nebo více rámců a vyrovnávací paměť j -tého portu, kde $i \neq j$, neobsahuje žádný rámeček. Z vyrovnávací paměti i -tého portu se přenesou dva rámečky a j -tý vstup bude také propojen s vybraným výstupem, ale nepřenesou se žádné datové jednotky. Přesunem některé datové jednotky z vyrovnávací paměti i -tého portu do vyrovnávací paměti j -tého portu by bylo možné zajistit, aby současně byly přeneseny dvě datové jednotky ze vstupu na výstup (když mají být přeneseny na odlišné výstupy). Rozšířením této myšlenky se dostaneme k centrální vyrovnávací paměti, do které budou vkládány všechny přijaté datové jednotky. Tato architektura je znázorněna na obr. 6.1. Využitím centrální paměti získáme další výhodu. V případě architektury na obr. 4.1



Obr. 5.6 Blokové schéma spojovacího pole s centrální pamětí

byl rozměr neuronové sítě dán celkovým počtem vstupních portů N , který se také rovná počtu výstupních portů. Po získání konfigurační matice dojde k propojení jednotlivých vstupů s vybranými výstupy. Prakticky však budou využity jen propojky, které vstup propojují s výstupem, kam skutečně směřuje datová jednotka. Z toho lze odvodit, že velikost neuronové sítě by se dala zredukovat na počet, daný různými východy kam směřují datové jednotky ze všech portů. Protože nárůstem velikosti neuronové sítě narůstá i doba potřebná k získání výsledku, redukcí velikosti neuronové sítě lze zvýšit rychlost zpracování vstupních informací. U architektury znázorněné na obr. 4.1 je zajištění seznamu o skutečně požadovaných výstupech značně obtížné. V případě centrální paměti je udržování seznamu o cílových výstupních portech mnohem jednodušší. Na základě seznamu můžeme určit, o kolik různých výstupů je skutečně zájem a podle toho dynamicky nastavit velikost neuronové sítě.

Dalším praktickým problémem, který musí být vyřešen, je situace, kdy ze stejného vstupního portu vedou dva datové toky na stejný výstupní port a tyto datové toky mají odlišnou prioritu. Uvedený algoritmus by v každém případě favorizoval datové jednotky s vyšší prioritou a datové jednotky s nižší prioritou by uvízly ve vyrovnávací paměti. Řešením této situace může být postupné zvyšování priority datových jednotek, které se v daném kroku nepřenesou na výstup. Tak lze zajistit, aby po určitém kroku i tyto datové jednotky měly dostatečně vysokou prioritu k tomu, aby byly vybrány.

6 ZÁVĚR

Cílem disertační práce bylo předvést praktickou ukázkou využití umělých neuronových sítí v oblasti optimalizace spojovacích procesů. Záměrně byla vybrána oblast implementace, kde se vykonává poměrně jednoduchá funkce, ale jsou kladeny velké nároky na rychlost provedení těchto úkolů. Optimalizace spojovacího pole těchto aktivních prvků je velmi aktuální problematikou. Jedná se o oblast, kde se paralelní zpracování dat dá využít velmi efektivně. V disertační práci je uvedena architektura přepínače, která obsahuje umělou neuronovou síť pro optimalizaci prioritního přepínání. Byl vypracován postup, jak danou problematiku modelovat matematicky a dále jak tento matematický model realizovat pomocí Hopfieldovy neuronové sítě.

Matematické zázemí prezentované v disertační práci lze také využít jako obecný popis, jak efektivně využívat Hopfieldovu neuronovou síť pro řešení dalších typů optimalizačních úloh. V práci je uvedeno, jakým způsobem se dají oddělit omezující kritéria od účelové funkce optimalizační úlohy a řešit tyto komponenty optimalizační úlohy tak, aby uvedené dva procesy nepůsobily proti sobě. Na druhé straně je však nutné zdůraznit, že v případě řešení optimalizačního problému pomocí neuronové sítě se jedná o stochastický proces, a proto síť nemusí v každém případě vybrat nejlepší řešení.

V rámci řešení disertační práce bylo také vyvinuto simulační prostředí v jazyce JAVA, pro ověření platnosti odvozených matematických vztahů. Jak ukazují výsledky rozsáhlého testování, získané teoretické výsledky splňují očekávání.

Vytýčené cíle disertační práce byly splněny. Byl navržen a ověřen s využitím softwarové simulace systém, který se dá využít pro řízení spojovacího pole určeného pro přepojování datových jednotek se zohledněnou prioritou jednotlivých datových jednotek. Možným pokračováním práce může být konkrétní hardwarová implementace uvedeného modelu, např. pomocí programovatelných logických obvodů.

POUŽITÁ LITERATURA

- [1] HOPFIELD, J. J.: Neural Networks and physical systems with emergent collective computational abilities, Proceedings of the National Academy of Sciences of the USA, USA, 1982
- [2] HOPFIELD, J. J.: Neurons with graded response have collective computational properties like those of two-state neurons, Proceedings of the National Academy of Sciences of the USA, USA, 1984
- [3] HOPFIELD, J. J., TANK, D. W: “Neural” Computation of Decisions in Optimization Problems, Biological Cybernetics, Springer-Verlag, 1985
- [4] SMITH, A. K: Neural Networks for Combinatorial Optimization: A Review of More Than a Decade of Research, School of Business Systems, Monash University, Clayton, Australia, 1999
- [5] SMITH, A. K: Solving combinatorial optimization problems using neural networks, University of Melbourne, Australia, 1996
- [6] WILSON, V., PAWLEY, G. S.: On the stability of the TSP Problem Algorithm of Hopfield and Tank, Biological Cybernetics 58, 1988
- [7] HAYKIN S.: Neural Networks – A Comprehensive Foundation, IEEE PRESS, 1994
- [8] RINPIPE, J., EULIANO, R. N., LEFEBVRE, W. C: Neural and Adaptive Systems, John Wiley & Sons, Inc., 1999
- [9] HIRSCH, SMALE: Introduction to dynamical system and linear algebra, Academic Press, New York, 1974
- [10] MACKAY, D., J., C.: Information Theory, Inference and Learning Algorithms, University of Cambridge, United Kingdom, 2001
- [11] HOPCROFT J. E., ULLMAN, J. D.: Introduction to Automata Theory Languages and Computation, Reading, MA: Addison-Wesley, 1979
- [12] McCULLOCH, W., A.; PITTS, W.: A logical calculus of the ideas imminent in nervous activity, Bulletin Mathematical Biophysics, vol. 5, 1943
- [13] AIYER, S., V., B.: Solving combinatorial optimization problems using neural networks with applications in speech recognition, University of Cambridge, United Kingdom, 1991
- [14] GEE, A., H.: Problem solving with optimization networks, University of Cambridge, United Kingdom, 1993
- [15] GREUB, W., H.: Linear Algebra, Springer – Verlag, Berlin, 1967
- [16] AIYER, S., V., B; NIRANJAN M., FALLSIDE, F.: A theoretical investigation into the performance of the Hopfield model, IEEE Transactions on Neural Networks, vol. 1, No. 2, 1990
- [17] GRAHAM, A.: Kronecker Products and Matrix Calculus with Applications, Ellis Horwood Ltd, Chichester 1981
- [18] GEE, A., H., PRAGER, R. W.: Polyhedral combinatorics and neural networks, University of Cambridge, United Kingdom, 1993

- [19] HAVEL, V.: Lineární algebra (matice, lineární prostory, analytická geometrie), Ediční středisko VUT Brno, Brno, 1990
- [20] KOSKO, B.: Neural Networks and Fuzzy Systems, Prentice-Hall International, Inc., London, 1992
- [21] PRINCIPE, J. C., EULIANO, N. R., LEFEBVRE, W. C.: Neural and Adaptive Systems: Fundamentals through Simulations, John Wiley & Sons, Inc., New York, 2000
- [22] SMITH, K., PALANISWAMI, M., KRISHNAMOORTHY, M.: Neural Techniques for Combinatorial Optimization with Applications, IEEE Transaction on Neural Networks, Vol. 9, No. 6, 1998
- [23] TROUDET, T. P., WALTERS, S. M.: Neural Network Architecture for Crossbar Switch Control, IEEE Transaction on Circuits and Systems, vol. 38, No. 1, 1991
- [24] ABE, S., GEE, A. H.: Global Convergence of the Hopfield Neural Network with Nonzero Diagonal Elements, IEEE Transaction on Circuits and Systems - II: Analog and Digital Signal Processing, vol. 42, No. 1, 1995
- [25] GEE, A. H., PRAGER, R. W.: Limitations of Neural Networks for Solving Traveling Salesman Problems, IEEE Transaction on Neural Networks, Vol. 6, No. 1, 1995
- [26] LAMMLE, T.: The Sybex CCNA Third Edition Study Guide, Sybex International, Littleton, USA, 2000
- [27] MOLNÁR, K.: Spojovací pole ATM přepínače řízené pomocí neuronové sítě, VUT, Brno, 1999
- [28] GRAND, M.: JAVA - referenční příručka jazyka, Computer Press, Praha, 1998
- [29] FAHMY, S.: A Survey of ATM Switching Techniques, The Ohio State University, 1995
- [30] SIEGMUND, G., PRAGER, E.: ATM technika pro širokopásmové sítě ISDN, Nakladatelství Hutig & Beneš, Praha, 1997

Životopis

Ing. Karol Molnár (1974) působil na Ústavu telekomunikací FEKT VUT v Brně nejprve jako doktorand. Postgraduální studium zahájil po absolvování inženýrského studia v oboru elektrotechnika a sdělovací technika v roce 1997.

V průběhu doktorandského studia byl zaměstnán na FEKT VUT v Brně jako odborný asistent a současně jako technický pracovník ve funkci správce počítačové sítě a technik pro zabezpečení výuky předmětů Spojovací systémy, Služby v integrovaných sítích a Účastnická koncová zařízení. Zabývá se problematikou ATM a využitím neuronových sítí v oblasti komunikačních sítí. Spolupracoval na vybudování multimediální sítě v laboratoři spojovací techniky a má značné zkušenosti s budováním vzdělávacího centra na FEKT VUT v Brně. Byl zodpovědným řešitelem grantu FVU VUT v Brně, řešitelem projektů FRVŠ Zpřístupnění databází v oborové knihovně VUT v Brně a Inovace předmětu Digitální ústředny. Dále byl řešitelem grantového projektu SPT Telecom, spoluřešitelem grantového projektu multimediální encyklopedie síťových technologií a podílel se na tvorbě dvou dílů znalostní databáze pro firmu Aliatel.

Seznam vybraných publikací

- MOLNÁR, K.: ATM in education. Konference ATM v praxi, Olomouc, 1998. Sborník konference str. 72-74.
- MOLNÁR, K.: Multimediální servery. Článek ve sborníku Telekomunikace a multimédia, ČES, FEI VUT v Brně, 1998, str. 25-28.
- MOLNÁR, K., Sadílek, R.: Distribution of Video Streams in Intranet, 3rd. International Scientific Conference Elektro'99, Žilina, 1999, pp. 46-49, ISBN 80-7100-603-3.
- MOLNÁR, K.: ATM Swirch with Neural Control, 22. International Conference Telecommunication and Signal Processing- TSP'99, Brno, 1999, pp. 29-32, ISBN 80-214-1154-6.
- MOLNÁR, K.: Neural-network-controlled Switch Fabric, RTT'99, Praha, 1999, pp. 21-24, ISBN 80-01-02124-6.

- MOLNÁR, K., Sadílek, R.: Multimediální prostředí ve výuce, Sborník prací studentů a doktorandů, FEI VUT v Brně, Brno, 1999, pp. 171-172, ISBN 80-214-1155-4
- KAPOUN, V., MOLNÁR, K.: Otimization of Switch Fabrics, International Conference Research in Telecommunication Technology 2000, Ostrava, pp. 21 – 25, 2000, ISBN 80-7078-817-8,
- KAPOUN, V., MOLNÁR, K.: Digital switch fields TST, 23th International Conference Telecommunications and Signal Processing TSP 2000, Brno, pp. 108 – 110, ISBN 80-7204-161-4,
- MOLNÁR, K. HTTP Support in Java In Research in Telecommunication Technology 2001. Research in Telecommunication Technology RTT 2001. Brno: FEI VUT, 2001, s. 289 - 292, ISBN 80-214-1938-5
- MOLNÁR, K., NAGY, Z. Multimedia distribution system, MOBIVAS Workshop, Greece, str. 103-111, sborník je přístupné přes adresu <http://mobivas.cnl.di.oua.gr>, (přístup k interním dokumentům mají jen členové řešitelského týmu) In . MOBIVAS 2001, 2001
- MOLNÁR, K. Developing Multimedia Applications in Java In Proceedings of the conference TSP'2001. Telecommunications and Signal Processing TSP -2001. BRNO: Foerster, 2001, s. 135 - 138, ISBN 80-214-1962-8
- MOLNÁR, K.: Switching Optimization by Neural Networks, Conference Telecommunications and Signal Processing TSP 2002, Brno
- MOLNÁR, K.: Neural Networks for Solving Optimisation Problems, International Conference Research in Telecommunication Technology 2002, Žilina

Abstrakt

Rapidním nárůstem multimediálních síťových aplikací se velmi výrazně projevuje velká nevýhoda klasických přenosových protokolů počítačových sítí, která spočívá v tom, že každá datová jednotka má stejnou prioritu. Chybějící podpora prioritního zpracování vybraných datových jednotek v aktivních síťových uzlech má velmi záporný vliv na aplikace vyžadující komunikace v reálném čase. Proto se hledají různé způsoby pro odstranění tohoto nedostatku z klasických přenosových protokolů. Tato podpora ovšem musí být implementována jak v síťové aplikaci, tak i aktivním síťovém prvku. Uvedená práce ukazuje možnost, jak provádět optimalizaci přepínání v aktivním síťovém uzlu při zohlednění priorit datových jednotek. Realizace je založena na využití umělých neuronových sítích, konkrétně na využití Hopfieldovy sítě.

Abstract

The mathematical background of artificial neural networks has been known for several decades. In spite of this fact, systems using artificial neural networks are still considered a kind of curiosity. This is probably caused by the hardware architecture required by neural networks. To take real advantage of neural networks, a parallel system architecture is required, which is made up of a great number of processor units that carry out relatively simple tasks. This architecture is quite different from processor-based architectures, which are used at this time but on the other hand neural networks are able to execute their functions faster. This is also the reason why the recent trends in digital integrated circuit and system development lead to parallel architectures. As a result of these trends, considerable expansion of neural network based systems can be expected.

The aim of this PhD thesis is to present a practical implementation of an artificial neural network for solving optimization problems. The field of implementation where a relatively simple function is executed but with the requirement for a very

high speed, was selected on purpose. Essential basics for the modern information society are communication networks able to transfer a huge amount of data with minimal transmission delay. The demands on the capacity of active network elements are still growing and this is the reason why the optimization of data processing inside these network elements will acquire high topicality. In addition, the function of multiport network elements is easy to realize with a parallel system architecture. In the thesis a switch architecture is developed that uses artificial neural networks to optimize the switching process based on the priorities assigned to data units.

Although the mathematical analysis is oriented to a concrete optimization problem several general mathematical formulae are derived first. The mathematical background presented also helps to solve other optimization problems by Hopfield neural network. The thesis contains a description of how to separate the confinement and the object function of the optimization problem and how to solve these two components of the optimization process such that they do not work one against another.

Together with the mathematical framework a simulation environment was created to evaluate the theoretical results. This simulation environment was used for detailed analysis of the neural network developed. The simulation results show that the theoretical results really satisfy the expectations.

The function of the neural network developed can be controlled by several parameters and the performance of the network is highly dependent on these parameters. This is the reason why a chapter with a detailed description of concrete implementation is added to this work.