

České vysoké učení technické v Praze  
Fakulta dopravní  
Ústav aplikované informatiky v dopravě



## **HOMEOSTATICKÁ NEURONOVÁ SÍŤ PRO MODELOVÁNÍ ROZHRANÍ ČLOVĚK-STROJ**

MARTIN RŮŽEK

Studijní program: INŽENÝRSKÁ INFORMATIKA  
Specializace: Inženýrská informatika v dopravě a spojích

Praha, Srpen 2015

---

**Školitel:**

doc. Dr. Ing. Tomáš Brandejský  
Ústav aplikované informatiky v dopravě

Fakulta dopravní  
České vysoké učení technické v Praze  
Konviktská 20  
110 00 Praha 1  
Česká republika

---

## Prohlášení

Předkládám tímto k posouzení a obhajobě disertační práci zpracovanou v rámci doktorského studia na Ústavu aplikované informatiky Fakulty dopravní ČVUT.

Prohlašuji, že jsem tuto práci vytvořil samostatně, s použitím literatury a zdrojů uvedených v seznamu a za pomoci legálních kopií řádě registrovaného nebo volně šiřitelného softwarového vybavení.

V Praze dne

Martin Růžek

---

## Anotace

V řadě dopravních úloh, jako je predikce poklesu pozornosti a nesprávného rozhodnutí řidiče, by bylo výhodné disponovat modelem schopným napodobit chování člověa. Doposud byly podobné modely obtížně realizovatelné z důvodů nedostatku znalostí o kognitivních a rozhodovacích procesech člověka. Postupné prohlubování vědomostí o rozhodovacích procesech a rostoucí výkon počítačů přibližují možnost reálného modelování rozhodovacích procesů a poklesu pozornosti. Tato práce se zabývá možnostmi využití umělých neuronových sítí pro modelování biologických neuronových sítí. Je navržena adaptace algoritmu zpětného šíření chyby, která je založena na principu hledání kompromisu mezi přesností popisu biologického neuronu a jednoduchostí výpočtu. Tento algoritmu pohlíží na neuronovou síť jakožto na systém autonomních jednotek soupeřících o zdroje, přičemž měřítkem úspěšnosti je užitečnost neuronu pro ostatní neurony, případně celý nadsystém. Navržená síť je otestována na u,ěle generovaných a reálných datech.

**Keywords:** umělá neuronová síť, predikce, modelování, homeostáze, HMI.

---

# Abstract

In a high variety of transport task, such as prediction of loss of attention or prediction of incorrect decision, it would be of great benefit to dispose of a model capable of simulation of the behaviour of real driver. Until now, similar models were difficult to realize because of a great complexity of a human brain and because of a lack of knowledge about its characteristics. The ongoing increase of knowledge about the decision processes and the growth of computational power make possible the realization of networks that can simulate the processes occurring during driving. This paper presents an adaptation of artificial neural networks for modelling of monotoneous activities. An improvement of the back propagation algorithm is proposed. This algorithm is based on the idea of compromise between simplicity of neuron and fidelity of the description to the biological sample. The idea is realized by the search of information homeostatical position. The proposed network is tested on artificial and real data.

**Keywords:** artificial neural network, prediction, modelling, HMI.

---

## Poděkování

Tato práce by nemohla vzniknout bez podpory mé rodiny a řady pedagogů a vědců z ČVUT a Ústavu informatiky AV ČR, z nichž bych chtěl poděkovat zejména mému školiteli doc. Dr. Ing. Tomáši Brandejskému za dlouhodobé vedení této práce a za množství nápadů, které jsem později využil při výzkumu, dále prof. Ing. Mirko Novákovi DrCs. za cenné připomínky a motivaci ke studiu umělých neuronových sítí a prof. MUDr. Josefu Faberovi, DrCs za ochotné předávání znalostí o biologických neuronových sítích.

---

# Obsah

<b>Seznam zkratek</b>	<b>xiii</b>
<b>1 Úvod</b>	<b>1</b>
1.1 Cíl práce . . . . .	1
1.2 Biologický neuron . . . . .	2
1.3 Biofyzikální modely neuronu . . . . .	4
1.3.1 Model Integrate and Fire (Louis-Lapicqueuv model) . . . . .	4
1.3.2 Hodgkin-Huxleyův model . . . . .	5
1.3.3 FitzHugh-Nagumův model . . . . .	7
1.4 Formální modely neuronu . . . . .	7
1.4.1 McCulloch-Pittsův perceptron . . . . .	7
1.4.2 Farley a Clarkův model . . . . .	9
1.4.3 Hopfieldova síť . . . . .	10
1.4.4 GMDH (Group Method of Data Handling) . . . . .	12
1.4.5 SONU (Second Order Neural Unit) . . . . .	15
1.4.6 HONNU . . . . .	18
<b>2 Simulace mozkových funkcí</b>	<b>19</b>
2.1 Blue Brain projekt . . . . .	19
2.1.1 Shoda mezi pozorovanými projevy mozku a modelem pomocí měření potenciálu . . . . .	21
2.1.2 Topologická analýza neuronové sítě . . . . .	21
2.2 SyNAPSE . . . . .	23

---

2.2.1	GENESIS . . . . .	27
2.2.2	NEST . . . . .	27
<b>3</b>	<b>Navržený model homeostatického neuronu</b>	<b>29</b>
3.1	Východiska navržené neuronové sítě . . . . .	29
3.1.1	Model rozhodovacích procesů lidského operátora . . . . .	29
3.1.2	Úrovně řízení . . . . .	30
3.1.3	Požadavky na neuronovou síť . . . . .	32
3.1.4	Kontinuita učení a odolnost vůči „catastrophic interference“ . . . . .	32
3.1.5	Výběr problémových oblastí vhodných pro realizaci modelu . . . . .	33
3.1.6	Problémy systémů pro automatické vyhodnocení funkce lidského operátora . . . . .	35
<b>4</b>	<b>Princip umělé homeostatické neuronové sítě</b>	<b>36</b>
4.1	Informační homeostáze . . . . .	36
4.2	Přenosová funkce neuronu . . . . .	37
4.3	Užitečnost neuronu . . . . .	37
4.4	Učení navržené neuronové sítě . . . . .	38
4.5	Citlivostní analýza neuronové sítě . . . . .	40
4.5.1	Současný výběr více vah na základě analýzy citlivosti . . . . .	41
4.6	Varianty učícího algoritmu . . . . .	41
4.6.1	Úprava náhodně vybrané váhy . . . . .	41
4.7	Postupná úprava vah . . . . .	42
4.8	Omezení vah . . . . .	43
4.9	Přehled variant učícího algoritmu homeostatického neuronu . . . . .	43
4.10	Ohodnocení kvality neuronové sítě . . . . .	44
<b>5</b>	<b>Realizace programu založeného na principu hledání rovnovážné polohy</b>	<b>45</b>
5.1	Topologie homeostatické neuronové sítě pro analýzu a predikci lidského operátora . . . . .	45
5.1.1	Program učení jediného neuronu . . . . .	47
5.1.2	Popis stavů neuronu . . . . .	48
5.2	Algoritmus učení neuronové sítě . . . . .	49
5.3	Velikost učícího kroku . . . . .	50
5.4	Problémy učení na principu informační homeostáze . . . . .	51



---

5.4.1	Zpoždění neuronové sítě . . . . .	52
5.4.2	Učení nejvyšší vrstvy neuronové sítě . . . . .	52
<b>6</b>	<b>Ověření principu učení homeostatického neuronu</b>	<b>54</b>
6.1	Použitá data . . . . .	54
6.1.1	Uměle generovaná data . . . . .	55
6.1.2	Reálná data . . . . .	55
6.2	Vyhodnocení kvality celé homeostatické neuronové sítě . . . . .	57
6.2.1	Statistické vyhodnocení výsledků . . . . .	58
<b>7</b>	<b>Příklad učení vybraného neuronu</b>	<b>59</b>
7.1	Příklad učení vybraného neuronu . . . . .	59
7.1.1	Porovnání učících algoritmů pro samostatný neuron z hlediska rychlosti učení . . . . .	59
7.1.2	Porovnání učících algoritmů pro samostatný neuron z hlediska schopnosti generalizace . . . . .	62
7.1.3	Porovnání učících algoritmů pro samostatný neuron z hlediska kontinuity učení . . . . .	63
7.1.4	Celkové srovnání učících algoritmů pro samostatný neuron . . . . .	64
7.2	Příklad učení neuronové sítě . . . . .	66
7.2.1	Příklad učení neuronové sítě umělými daty . . . . .	66
7.2.2	Příklad učení neuronové sítě s reálnými daty . . . . .	69
<b>8</b>	<b>Diskuze a nástin pokračování práce na tématu homeostatické neuronové sítě</b>	<b>72</b>
<b>9</b>	<b>Závěr</b>	<b>74</b>
	<b>Použitá literatura</b>	<b>77</b>
	<b>Publikace autora k tématu homeostatická neuronová síť</b>	<b>82</b>

---

## Seznam obrázků

1.1	Velmi zjednodušené schéma biologického neuronu . . . . .	4
1.2	Farleyův a Clarkův model skládající se z generátoru signálů, diskriminační jednotky, formátoru a pole neuronů (komplexu) rozděleného na 4 zmíněné části. . . . .	10
1.3	Ukázka topologie sítě MIA, jejíž uření bylo ukončeno po 4 krocích. . . . .	14
1.4	Typický průběh alfa vlny během mentace. Na začátku dochází k postupnému zapojování neuronů, poté se počet aktivních neuronů postupně snižuje. . . . .	15
1.5	Schéma SONU (Second Order Neural Unit) . . . . .	17
1.6	SONU se na rozdíl od McCullochova a Pittsova neuronu může naučit funkci XOR, ačkoliv nemá žádnou skrytou vrstvu. Existuje několik způsobů, jak může rozdělit vstupní prostor, např. elipsou, hyperbolou nebo parabolou. V tabulce 1.1 je uvedeno několik případů váhové matice $w$ , které dělí vstupní prostor způsobem odpovídajícím tomuto obrázku . . . . .	17
2.1	Modelové akční potenciály v okolí nervových buněk ze 3 různých vrstev, které byly vybuzeny 10 ms prodovým pulsem o napětí $400 \mu V$ . Soma a dendrity jsou šedé, axon je červený. Nejvýraznější odezvu má neuron 4 vrstvy. . . . .	21
2.2	Příklad shlukování neuronů mezi původním uspořádáním (A) a uspořádáním po přivedení signálu s proměnlivou frekvencí (B) zobrazené v projekci z třídimenzionálního prostoru . . . . .	23
2.3	Schéma neuronu vestavěného do 45nm chipu. V modelu se nachází celkem 8 nastavitelných parametrů. LFSR (linear feedback shift register) je generátor náhodných čísel nutný pro pravděpodobnostní učení parametrů neuronu. . . . .	25
2.4	Reprezentace neuronové sítě určené pro samoučení . . . . .	26

---

3.1	Příklad vstupů, vlastní neuronové sítě a výstupu při procesu rozhodování. V reálném světě je tento proces komplikovanější, např. velkou roli hrají minulé stavy (které mohou být uloženy v NN) . . . . .	31
6.1	Zjednodušené schéma informačního toku v reálném světě a v modelu. . . . .	56
6.2	Vztah použitého modelu a modelovaného systému. . . . .	57
7.1	Učení neuronu metodou back propagation pro jednoduchý případ se dvěma vstupy a jedním prahem (tj. neuron se třemi vahami). Červený bod značí počáteční náhodné nastavení vah a zelený bod konečné váhy (neuron v tomto bodě splnil učicí kritérium). . . . .	60
7.2	Učení homeostatického neuronu podle algoritmu úpravy nejvýznamnější váhy.	60
7.3	Učení homeostatického neuronu podle algoritmu úpravy postupně vybrané váhy. . . . .	61
7.4	Učení homeostatického neuronu podle algoritmu úpravy náhodně vybrané váhy. . . . .	61
7.5	Rychlost učení algoritmu back propagation a homeostatické neuronové sítě učící se na principu součtu výstupních vah. Z grafu je patrné, že s rostoucím počtem vstupů dosahuje metoda back propagation podobných výsledků. Pro přehlednost je rychlost učení vyjádřena logaritmičticky. Hodnoty byly získány zprůměrování 100 běhů výpočtu pro každý počet vstupů. . . . .	69
7.6	Porovnání schopnosti generalizace algoritmu back propagation a homeostatické neuronové sítě. Je zřejmé, že pro větší sítě dávají obě metody podobné výsledky.	69
7.7	Kontinuita učení pro homeostatickou neuronovou síť a síť s učením typu back propagation. . . . .	70
7.8	Příklad vrozků z databáze MNIST . . . . .	70

---

## Seznam tabulek

1.1	Váhy neuronu SONU, který rozděluje vstupní prostor na 2 lineárně neseparovatelné oblasti. Stejnou funkci může realizovat i klasická MLP, ale pouze za podmínky většího počtu neuronů uspořádaných do alespoň 3 vrstev. . . . .	18
5.1	Navržená topologie neuronových sítí . . . . .	46
7.1	Počet iterací nutných pro dosažení požadované přesnosti realizované funkce pro jednotlivé typy učících funkcí. . . . .	62
7.2	Výpočetní čas v milisekundách nutný pro dosažení požadované přesnosti realizované funkce pro jednotlivé typy učících funkcí. . . . .	63
7.3	Procentuální zhoršení střední kvadratické chyby po přivedení mírně změněného vzoru na vstup již naučeného neuronu. Pro vyloučení náhodného vlivu jsou uvedena čísla zprůměrována přes 100 simulací pro každou kombinaci. . . . .	64
7.4	Kontinuita učení vyjádřená jakožto počet iteračních cyklů nutných k opětovnému dosažení požadované MSE poté, co došlo k malé změně vstupního vektoru. . .	65
7.5	Rychlost učení homeostatické neuronové sítě pro uměle generovaná data. Porovnání s metodou back propagation . . . . .	67
7.6	Schopnost generalizace homeostatické neuronové sítě pro uměle generovaná data. Porovnání s metodou back propagation. . . . .	67
7.7	Kontinuita učení homeostatické neuronové sítě pro uměle generovaná data. . .	68
7.8	Rychlost učení homeostatické neuronové sítě pro reálná data. Porovnání s metodou back propagation. Topologie sítě je 748-1000-1. . . . .	71

---

## Seznam zkratek

<i>AIC</i>	Akaikeho informační kritérium
<i>DARPA</i>	Defense Advanced Research Projects Agency
<i>D</i>	Výstupní (požadovaná) data neuronové sítě
<i>D<sub>ext</sub></i>	Rozšířená matice výstupních dat
<i>GENESIS</i>	GEneral NEural SIMulation System
<i>HAN</i>	Homeostatický umělý neuron (Homeostatical Artificial Neuron)
<i>HANN</i>	Homeostatická umělá neuronová síť (Homeostatical Artificial Neural Network)
<i>MLP</i>	Multi layered perceptron, vrstevnatá neuronová síť
<i>MPI</i>	Message Passing Interface
<i>MSE</i>	Mean square error, střední kvadratická odchylka
<i>PCA</i>	Principal component analysis, analýza hlavních komponent
<i>q<sub>j</sub></i>	Užitečnost neuronu číslo j
<i>RBF</i>	Radial Basis Function, typ neuronové sítě
<i>SyNAPSE</i>	Systems of Neuromorphic Adaptive Plastic Scalable Electronics
<i>step<sub>norm</sub></i>	Konstanta pro normalizaci velikosti učícího kroku
<i>step<sub>offset</sub></i>	Minimální velikost učícího kroku
<i>w<sub>j</sub><sup>i</sup></i>	Vstupní váha neuronu číslo j
<i>w<sub>o</sub><sup>j</sup></i>	Výstupní váha neuronu číslo j
<i>W<sub>i</sub></i>	Input weights, množina vstupních vah
<i>W<sub>o</sub></i>	Output weight, množina výstupních vah
<i>w<sub>ext</sub></i>	Rozšířená matice vah
<i>X</i>	Vstupní data neuronové sítě

---

# Úvod

## 1.1 Cíl práce

Při řešení mnoha technických problémů je výhodné disponovat modelem lidského operátora.

Zatímco v určitých, zpravidla velmi specifických oblastech, je možné popsat a modelovat rozhodování člověka pomocí umělých systémů, v dopravě to často není reálné.

Příkladem obtížně modelovatelného systému je řidič automobilu nebo pilot letadla, který reaguje na širokou škálu vnějších podnětů a vnitřních stavů. Tyto typy interakce člověka se systémem nelze popsat jako jednoduchý regulátor, protože jednání člověka je zde ovlivněno mnoha faktory, jako např. metaforické a analogické uvažování, práce s neurčitými či neúplnými daty atp.

Přínos vytvoření modelu lidského operátora zde spočívá v návrhu technických prostředků pro podporu rozhodování a predikci situací s vyšší pravděpodobností chybného rozhodnutí.

Všechny pokusy o modelování duševních procesů však narážejí na potřebu porozumění a uchopení procesů probíhajících v lidském mozku.

Existuje celá řada postupů využitelných pro modelování mozkových funkcí, jako jsou rozhodovací stromy, celulární automaty, Bayesovské sítě a další.

Neuronové sítě mají oproti ostatním výhodu podobnosti s biologickým vzorem, proto byly mnou zde zvoleny jako nejvhodnější metoda. Druhou výhodnou vlastností neuronových sítí je schopnost řešit úlohy, které nejsou dostatečně přesně popsány. Umožňují modelovat řadu komplexních úloh a mají potenciál napodobit lidské vnímání, uvažování a intuici.

Cíle mé disertační práce lze shrnout do čtyř základních bodů.

- (a) Vytvořit přehled metod pro modelování a simulování duševních funkcí člověka se zaměřením na využití umělých neuronových sítí.
- (b) Popsat využití výpočetní techniky pro studium činnosti lidského operátora nebo případně jeho nahrazení. Tato oblast je předmětem intenzivního zájmu od druhé poloviny dvacátého století, kdy v důsledku rozvoje výpočetní techniky bylo možné realizovat funkce předpokládané u člověka. Neuronové sítě se během této doby střídavě dostávaly do středu zájmu, nicméně i jiné metody byly často využívány.
- (c) Ve vhodném programovém prostředí navrhnout umělou neuronovou síť schopnou modelovat duševní funkce řidiče se zaměřením na model popisující operátora dopravního prostředku. Tato síť bude silným zjednodušením biologické neuronové sítě, ale zároveň musí zachovávat její silné procesy. Je přípustné, aby navržená síť fungovala v diskretním čase, měla oddělenou paměť a výpočet či využívala generátoru náhodných čísel pro počáteční nastavení, zároveň však musí být schopna samoučení, adaptace na nové podmínky, a další vlastnosti typické pro biologickou neuronovou síť.
- (d) Na základě experimentálních dat posoudit vhodnost navržené neuronové sítě pro modelování požadovaných procesů a posouzení možností této sítě ve srovnání s jinými, existujícími metodami. Navrhnout metodiku pro ohodnocení neuronové sítě z hlediska funkčnosti a realizovatelnosti. Ověřit možnosti navržené sítě pro realizaci jiných úloh než je modelování duševních procesů, jako např. predikce nebo modelování signálu.

Ambicí této práce není vytvořit izomorfní model řidiče ani přesný popis procesů probíhajících v mozku řidiče, ale navrhnout zjednodušený model pro vybrané řidičovy rozhodovací procesy, které musí provádět při svém řízení vozidla.

## 1.2 Biologický neuron

Umělé neuronové sítě byly inspirovány uskupením biologických neuronů [1], které plní dvě základní funkce:

- trofickou funkci – tj. vyživování samotného neuronu

- o informační funkci – tj. přenos informace prostřednictvím elektrických impulsů a chemických látek.

Obě tyto funkce jsou spolu úzce svázány a není možné je oddělit, protože vyživování neuronů je závislé na informační funkci.

V rámci této práce budu prezentovat model neuronu, který zohledňuje obě funkce. Informační funkce samu popíšu podrobněji později.

Biologický neuron i jeho model, tak jak je znázorněn na obr. 1.1, lze rozdělit na 4 části:

- o tělo (soma),
- o dendrity,
- o axon a
- o synapse.

Tělo je největší částí neuronové buňky, ve které probíhá většina elektrochemických reakcí neuronu.

Dendrity jsou výběžky vedoucí vzruch do těla buňky, představují proto vstup do tohoto systému.

Synapse představuje rozhraní mezi dvěma neurony.

Axon je výstup z těla neuronové buňky, na který jsou prostřednictvím synapsí napojeny dendrity dalších neuronových buněk.

Jeden neuron má přibližně 10 000 synapsí a jeden axon.

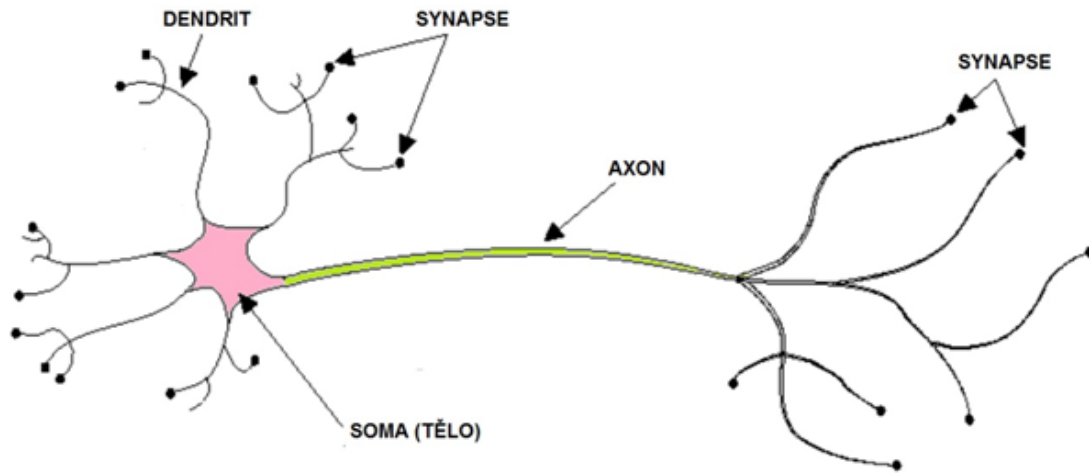
Z hlediska místa výskytu se synapse dělí na axosomatické (mezi axonem a tělem jiné buňky), axodendritické (mezi axonem jedné a dendritem druhé buňky) a axoaxonální (mezi dvěma axony).

Z hlediska způsobu přenosu informace lze synapse rozdělit na elektrické a chemické. Elektrické synapse přenášejí informace pomocí iontů. Tento způsob přenosu je nejjednodušší. Doba přenosu informace přes synapsi je přibližně 1 ms.

U chemických synapsí je doba přenosu podstatně delší a zároveň chemické přenašeče (mediátory, transmittery atd.) přenášejí větší množství informace.

Z pohledu umělých neuronových sítí je často neuron uvažován jako základní element, ve skutečnosti se ale jedná o komplikovanou strukturu. Uvnitř těla neuronu dochází k vnitřnímu přenosu mnoha informací, které jsou podstatně komplikovanější než přenos mezi jednotlivými neurony.





Obrázek 1.1: Velmi zjednodušené schéma biologického neuronu

Z hlediska cíle této práce se vnitřními přenosy informace uvnitř neuronů zabývat nebudu.

## 1.3 Biofyzikální modely neuronu

V této části práce budu popisovat některé vhodné základní modely biologického neuronu, a to:

- Integrate and Fire model,
- Hodgkinův-Huxleyův model a
- FitzHugh-Nagumův model.

### 1.3.1 Model Integrate and Fire (Louis-Lapicqueuv model)

Tento model, jehož autorem je francouzský neurolog Louis Lapicque (1907) [2, 3, 4], využívá analogie s jednoduchým elektrickým obvodem tvořeným odporem a kondenzátorem. Výstupní proud neuronu popisuje pomocí rovnice

$$I(t) = C_m \frac{dV_m}{dt}, \text{ kde} \quad (1.1)$$

$C_m$  je kapacita membrány,  $v_m$  je membránový potenciál a  $I(t)$  je výstupní proud neuronu. Pokud je na membránu oddělující neuron od vnějšího prostředí přiveden náboj, napětí na membráně roste až do hraniční hodnoty  $V_h$ , kdy dojde k výboji. Tento výboj je mode-

lován Dirackovou delta funkcí. Poté se neuron vrátí do původního stavu a proces se opakuje. Aktivační frekvence podle této teorie roste lineárně (a neomezeně) s proudem. Z důvodu postupné integrace náboje a následného výboje je model známý pod názvem Integrate and Fire. Tento model lze zpřesnit přidáním zpoždění  $t_r$ , což je časový úsek, během kterého se neuron nemůže aktivovat.

$$f(I) = \frac{I}{C_m V_h + t_r I}, \text{ kde} \quad (1.2)$$

$f(I)$  je aktivační frekvence (počet výbojů za vteřinu),

$V_h$  je hraniční napětí, při kterém dochází k výboji a  $I$  je konstatní budící proud.

Hlavním nedostatkem tohoto modelu je, že nepopisuje ztrátu napětí v čase. Pokud je jednou přiveden náboj na membránu, zůstane zde neomezeně dlouho dobu. Tento model lze považovat za jeden ze základních, historicky prvních modelů neuronu. Základní myšlenku tohoto neuronu (neuron se aktivuje po splnění určitých vnějších podmínek) budu v této práci také dále využívat.

### 1.3.2 Hodgkin-Huxleyův model

Zpřesněním modelu Integrate and Fire se zabývali fyziologové Alan Lloyd Hodgkin a Andrew Huxley. Jejich model z roku 1952 detailně popisuje proces nabíjení a výboje neuronu [5, 6]. Vychází z analogie s elektrickým obvodem, pomocí které modeluje vznik a šíření akčních potenciálů v neuronech. Pomocí nelineárních diferenciálních rovnic je možné přibližně popsat elektrické vlastnosti vzrušitelných buněk, jako jsou neurony nebo srdeční myocyty.

V tomto modelu se vyskytují dva typy kanálů: napěťově řízený kanál a pasivní kanál.

Napěťově řízené kanály mohou být otevřené nebo uzavřené v závislosti na potenciálu membrány.

Pasivní kanál je vždy otevřený a umožňuje průchod sodíkových a draslíkových iontů.

Všechny části neuronu mají svou vlastní elektrofyzikální analogii. Napěťově řízený iontový kanál je reprezentován nelineárními elektrickými rezistory  $g_n$ , kde  $n$  je číslo kanálu, ve kterých vodivost závisí na čase a na kanálu. Pasivní kanály jsou reprezentovány lineárními rezistory  $g_l$ . Elektrochemický gradient řídící tok iontů je reprezentován zdroji napětí  $E_n$  a  $E_l$ , iontové pumpy jsou reprezentovány zdroji proudu  $I_p$

Časová derivace potenciálu na membráně  $V_m$  je úměrná součtu proudů v obvodu:

$$\dot{V} = -\frac{1}{C_m} \sum_i I_i \quad (1.3)$$

kde  $I_i$  je iontový proud.

### Vlastnosti proudu iontů

Proud procházející iontovým kanálem lze popsat touto rovnicí:

$$I_i(V_m, t) = (V_m - E_i)g_i \quad (1.4)$$

kde  $E_i$  je potenciál  $i$ -tého iontového kanálu.

V napěťově řízeném kanálu je vodivost kanálu  $g_n(t, V)$  funkcí času a napětí, zatímco v pasivním kanálu je konstantní  $g_L$ . Proud generovaný iontovými pumpami závisí na druhu iontů, který je specifický pro každou pumpu.

### Napěťově řízené kanály

V Hodgkinův-Huxleyově modelu je vodivost napěťově řízených kanálů popsána takto:

$$g_n(V_n) = g_n \varphi^\alpha \chi^\beta \quad (1.5)$$

$$\dot{\varphi}(V_m) = \frac{1}{\tau_\varphi} (\varphi_\infty - \varphi) \quad (1.6)$$

$$\dot{\chi}(V_m) = \frac{1}{\tau_\chi} (\chi_\infty - \chi) \quad (1.7)$$

kde  $\varphi$  a  $\chi$  jsou řídicí proměnné pro aktivaci a deaktivaci reprezentující poměr maximální vodivosti při jakémkoliv čase a napětí.

$g_n$  je maximální hodnota vodivosti,

$\alpha$ ,  $\beta$  jsou reálné konstanty,

$\tau_\varphi$  a  $\tau_\chi$  jsou časové konstanty pro aktivaci a deaktivaci.

Pokud je membránový potenciál konstantní, lze řídicí rovnice popsat jako lineární diferenciální rovnice:

$$\varphi(t) = \varphi_0 - [(\varphi_0 - \varphi_\infty)(1 - e^{-\frac{t}{\tau_\varphi}})] \quad (1.8)$$

$$\chi(t) = \chi_0 - [(\chi_0 - \chi_\infty)(1 - e^{-\frac{t}{\tau_\chi}})] \quad (1.9)$$

Hodnota proudu platí pro libovolnou hodnotu membránového potenciálu:

$$I_n(t) = \bar{g}_n \varphi^\alpha \chi^\beta (V_m - E_n) \quad (1.10)$$

Pasivní kanály v modelu zastupují přirozenou průchodnost iontů membránami. Mají stejnou formu jako napěťově řízené kanály s konstantní vodivostí  $g_i$ .

Nevýhodou Hodgkin-Huxleyova modelu je jeho komplikovanost.

### 1.3.3 FitzHugh-Nagumův model

Podobně jako Hodgkinův a Huxleyův model je model FitzHugha a Naguma (1961) založen na myšlence, že neuron vysílá impuls, pokud hladina vnějších vstupů překročí určitou mez [7], následně se parametry neuronu vrátí do původního stavu.

Vznikl aplikováním fázové analýzy na redukovaný Hodgkinův a Huxleyův model o dvou proměnných. Rozlišuje část s rychlou a pomalou dynamikou, přičemž každá část je popsána jednou proměnnou. Systém je popsán rovnicemi

$$\frac{dv}{dt} = v(v - \alpha)(1 - v) - w + I, \quad (1.11)$$

$$\frac{dw}{dt} = \epsilon(v - \gamma w) \quad (1.12)$$

kde  $v$  je tzv. *rychlá proměnná* představující akční potenciál,  $w$  je tzv. *pomalá proměnná* reprezentující odpor hradla a  $\alpha$ ,  $\gamma$  a  $\epsilon$  jsou konstanty z intervalu  $(0,1)$ .

FitzHughův a Nagumův model je zjednodušením Hodgkinova a Huxleyova modelu s rozdílným způsobem aktivace neuronu. Silnou jeho stránkou je přesnější popis signálu v okolí výboje.

## 1.4 Formální modely neuronu

### 1.4.1 McCulloch-Pittsův perceptron

Jedná se o nejčastěji používaný formální model neuronu, který je oblíbený zejména pro svou jednoduchost. V následujícím textu bude použita nejrozšířenější definice dle [8, 9], v české terminologii je ale možné se setkat i s jinými definicemi.

Umělý neuron vznikl zjednodušením biologického neuronu. Má pevně daný (ale neomezený) počet vstupů, prahovou hodnotu danou reálným číslem a jeden výstup, který se může neomezeně větvit.

Vstup a výstup v původní verzi modelu byly binární hodnoty, v praxi se ale často používá spojité rozšíření.

Neuron realizuje dva procesy.

Prvním z nich je součet vážených vstupů. Vstupní signály jsou zde násobeny reálnými konstantami, které jsou „uloženy“ v synapsích. Synapse v tomto modelu plní funkci paměti [10].

Druhý proces modeluje přenosová funkce, tj. pokud součet působících signálů dosáhne alespoň prahové hodnoty, je výstup neuronu roven 1 (resp. u spojitěho modelu jisté reálné hodnotě), v opačném případě 0 (příp. -1).

Klasický perceptron zpracovává pouze binární signály a hodnota přenosové funkce  $S(x)$  je 1 pro  $x > 0$  a 0 pro  $x < 0$ .

V praxi se často používá spojité rozšíření vstupů a nelineární přenosové funkce, zejména ve tvaru sigmoidy:

$$S(x) = \frac{1}{1 + e^{-\alpha x}} \quad (1.13)$$

Funkci umělého neuronu lze pak vyjádřit vztahem

$$y = S\left(\sum_{i=1}^{i=n} (x_i w_i + \phi)\right) \quad (1.14)$$

kde  $\phi$  je práh (konstantní hodnota přiřazená neuronu, která nezávisí na vstupu) a  $S$  je přenosová funkce.

Robert Hecht-Nielsen dokázal praktickou využitelnost perceptronové sítě na základě Kolgorovova řešení 13. Hilbertova problému (ten formuloval tento význačný německý matematik jako výzvu k matematickému světu na světovém matematickém kongresu v Paříži v roce 1900. Jeho řešení dlouho odolávalo snaze světové matematické obce, až v roce 1953 uvedl jeho úplné řešení ruský matematik Kolmogorov. To pak opět upadlo v zapomnění, až v roce 1986 Američan R. Hecht-Nielsen ukázal, že na tomto matematickém základě je možno dokázat, že neuronovou sítí s vrstevnatým uspořádáním McCullochových a Pittsových modelů, která má alespoň 3 vrstvy, a dopředné - feed forward - propojení všech neuronových modelů v jedné vrstvě s neuronovými modely ve vrstvě vyšší, je možno, pokud neklademe omezení na počet neuronů, aproximovat s libovolnou přesností libovolnou funkci.

Tento Hecht-Nielsenův aproximační teorém je však pouze existenční (tj. ví se, že řešení vždy existuje), není však konstruktivní (tj. neříká, jak má být taková síť uspořádána).

McCullochovy a Pittsovy neurony jsou ve většině aplikací uspořádány do struktury označené jako MLP (Multi Layered Perceptron), což je vrstevnatá síť, ve které jsou neurony dané vrstvy propojeny se všemi neurony ve vyšší vrstvě. Nejsou propojeny se žádnou jinou vrstvou a nevyskytují se zde laterální vazby (vazby mezi neurony téže vrstvy). Všechny vazby jsou dopředné.

Pro učení MLP se nejčastěji používá algoritmus zpětného šíření chyby (back propagation [11]). Jedná se o heuristický iterativní algoritmus, který nezaručuje nalezení optimálního řešení. Je založen na principu postupného šíření chyby v opačném směru než v jakém se šíří signál. Zpětná propagace chyby vede ke změnám vah takovým způsobem, aby došlo ke snižování rozdílu mezi požadovaným a skutečným výstupem sítě.

Síť neuronů složená podle tohoto teorému z jednotlivých navzájem vrstevnatě uspořádaných modelů může realizovat libovolné zobrazení, neexistuje ale obecný algoritmus pro určení její topologie ani pro stanovení vah či prahů.

Výhodou tohoto vrstevnatého modelu je však jeho jednoduchost.

## 1.4.2 Farley a Clarkův model

Jednou z historicky prvních hardwarových realizací umělé neuronové sítě byl model Farleye a Clarka z roku 1954 [12], který je založen na Hebbově učícím principu.

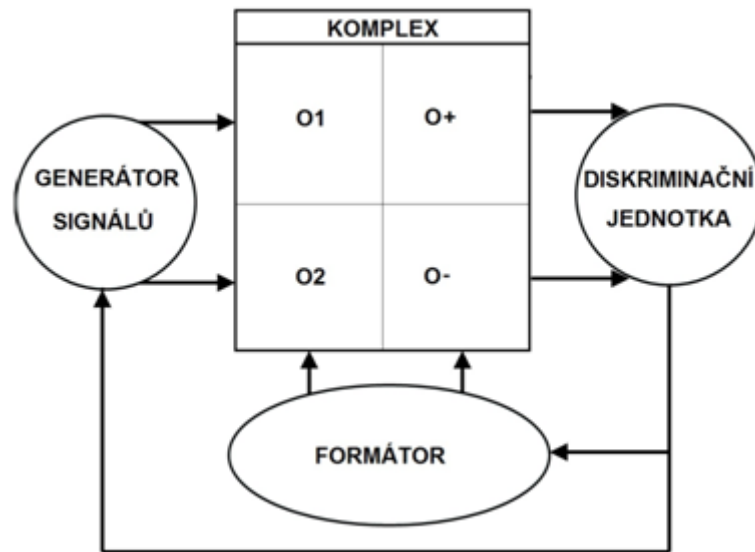
Tento princip tvrdí, že pokud se dva neurony aktivují současně, zvyšuje se pravděpodobnost existence vazby mezi nimi.

Farleyova a Clarkova síť se skládá z modelů neuronů plně propojených jednosměrnými vazbami. Tyto vazby násobí každý procházející signál reálnou konstantou, která je na počátku učení vybrána náhodně.

Každý uzel této sítě provádí sumaci vstupů a výstup vysílá, pokud hodnota vnitřního potenciálu překročí danou mez.

Celá síť je rozdělena do čtyř částí, přičemž jeden vzor vstupuje pouze jedné z nich (obvykle do levé horní, jak je naznačeno na obr. 1.2.) - část (O1), zatímco ostatní vzory vstupují do části (O2).

Požadavkem je, aby v případě, kdy je aktivní část O1, byla aktivní také pravá horní část O+ a zároveň neaktivní pravá dolní část O-. Pokud je aktivní část O2, měla by být aktivní pouze O-.



Obrázek 1.2: Farleyův a Clarkův model skládající se z generátoru signálů, diskriminační jednotky, formátoru a pole neuronů (komplexu) rozděleného na 4 zmíněné části.

Síť je vybavena „učitelem“ - formátorem, který posiluje změny vah v případě, kdy síť má požadované chování, a mění je tehdy, kdy se od cílového chování vzdaluje.

Důležitým rysem této sítě je schopnost tzv. samoorganizace, což je typická vlastnost také u biologických neuronových sítí, která je také požadována v navržených homeostatických neuronových sítích.

### 1.4.3 Hopfieldova síť

Jedná se o jednovrstvou síť, jejímž základem jsou McCullochovy-Pittsovy neurony [13, 14, 15] propojené podle vzoru „každý s každým“. Práh všech neuronů je na začátku nulový a výstup každého neuronu vede zpět na vstupy všech ostatních neuronů.

Z důvodu propojení typu každý s každým se váhy často vyjadřují pomocí matice vah. Váha na spojích mezi dvěma neurony je stejná v obou směrech, matice vah je proto symetrická.

Hopfieldova síť patří do skupiny asociativních pamětí, což znamená, že odpovědí na předložený vzor je přímo tento předložený vzor, a to i v případě, že byl vzor poškozen (za předpokladu, že síť byla na vzor předtím naučena a míra poškození vzoru nepřekročí určitou hranici).

Hopfieldova síť pracuje ve dvou fázích - učení a vybavování.

Ve fázi učení jsou na vstup přiváděny binární vzory. Váhy mezi neurony jsou nastaveny

iterativním (opakovaným) procesem, ve kterém se váha mezi dvěma neurony v čase  $t$  počítá podle vztahu

$$w_{ij} = w_{ij} * + (2v_i - 1)(2v_j - 1) \quad (1.15)$$

kde  $w_{ij}$  je váha mezi  $i$ -tým a  $j$ -tým neuronem v čase  $t$ ,  $w_{ij}^*$  je váha mezi  $i$ -tým a  $j$ -tým neuronem v čase  $t-1$  a  $v_i$  resp.  $v_j$  je hodnota výstupu  $i$ -tého resp.  $j$ -tého neuronu v čase  $t-1$ . Práhy jsou určeny dle vztahu

$$\theta_i = \frac{1}{2} \sum_{j=1}^N w_{ij} \quad (1.16)$$

jedná se tedy vždy o polovinu součtu vah do referenčního neuronu.

Ve vybavovací fázi, která je také iterativní, je na vstup přiveden vzor, který může být poškozený.

V případě vybavování neúplného vzoru se užívá tzv. energetická funkce  $E(t)$ , která určuje, kdy je možné proces vybavování ukončit.

$$E(t) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} y_i(t) y_j(t) - \sum_{i=1}^N \theta_i y_i(t) \quad (1.17)$$

kde  $w_{ij}$  je váha mezi  $i$ -tým a  $j$ -tým neuronem,

$y_i(t)$  je výstup neuronu číslo  $i$  v čase  $t$ ,

$y_j(t)$  je výstup neuronu číslo  $j$  v čase  $t$  a

$\theta_i$  je práh neuronu číslo  $i$ .

Během vybavování klesá hodnota tzv. energetické funkce. Pokud se pokles zastaví, znamená to, že se upravený vzor nachází v lokálním minimu a není možné dosáhnout dalšího zlepšení. Proces vybavování je v tom případě ukončen.

Bod zastavení poklesu energetické funkce může odpovídat globálnímu nebo lokálnímu minimu. Pokud se jedná o lokální minimum, nelze zaručit, že vybavování bylo zcela úspěšné. V určitých situacích může být odpovědí Hopfieldovi sítě inverzní vzor k předloženému vzoru.

Hopfieldova síť se používá např. pro rekonstrukci poškozeného obrazu.

Z hlediska modelů biologických neuronových sítí je zajímavé zejména propojení typu „všech se všemi“. Ačkoliv toto zapojení se v reálných sítích nevyskytuje, popisuje Hopfieldova síť důležitý princip laterálních vazeb.



Biologické neuronové sítě jsou také uspořádány do vrstev, přičemž mezi neurony ve stejné vrstvě existuje obvykle velké množství různých propojení.

#### 1.4.4 GMDH (Group Method of Data Handling)

Group Method of Data Handling je variantou neuronové sítě, u které není známa předem topologie. Tvorba sítě splývá s jejím učením, vazby mezi neurony, stejně tak jako nové vrstvy neuronů, vznikají až po přivedení vstupního signálu.

Tento typ sítě byl navržen ukrajinským matematikem A. G. Ivachněnkem v roce 1968 [16].

Jedná se o síť polynomiálního typu učící se s učitelem. Neurony jsou uspořádány do vrstev bez zpětných vazeb. Každý neuron má právě jeden výstup, který může být libovolně rozvětven (podobně jako u MLP).

Síť MLP se rovněž skládá z neuronových jednotek, rozdíl spočívá v přenosové funkci. Namísto váženého součtu neuron počítá ze vstupů polynom.

Výsledek polynomu je výstupem neuronu, nepoužívá se žádná nelineární přenosová funkce. Příkladem může být neuron se dvěma vstupy ( $x_1, x_2$ ) a výstupem  $y$ , jehož polynomiální funkce je:

$$y = ax_1^2 + bx_2^2 + cx_1x_2 + dx_1 + ex_2 + f \quad (1.18)$$

kde  $a$  až  $f$  jsou hledané parametry neuronů.

Nové vrstvy jsou přidávány, dokud síť nesplní kritérium pro ukončení učení, přičemž tímto kritériem obvykle bývá střední kvadratická odchylka (MSE) mezi skutečným a požadovaným výstupem. S každou přidanou vrstvou se zvyšuje stupeň výsledného polynomu (v případě učení podle rovnice (1.18) se s každou přidanou vrstvou zvyšuje stupeň polynomu o 2).

Učení lze popsat pomocí následujícího algoritmu: v prvním kroku jsou nastaveny parametry neuronů v první vrstvě, např. tak, že je přivedeno 6 vstupních vektorů a na jejich základě jsou určeny parametry neuronů (jedná se o 6 rovnic o 6-ti neznámých). Ve vyšší vrstvě je  $\binom{n}{2}$  neuronů, kde  $n$  je počet vstupů v předchozí vrstvě, případně počet vstupů pokud se jedná o první vrstvu. To znamená, že ve sledované (první) vrstvě jsou obsaženy všechny možné kombinace vstupů.

V druhém kroku dochází k eliminaci neuronů první vrstvy. Toto se provádí tak, že na vstup je přiveden jiný, testovací vektor, a je určena odchylka mezi požadovaným a skutečným

výstupem.

Následně jsou neurony seřazeny podle výsledku a nejhorší část je odstraněna. Poté je tato vrstva zakonzervována, její parametry se dále již nemění. Následně se postup opakuje s další vrstvou. V okamžiku, kdy některý neuron splní učicí kritérium, je trénování sítě ukončeno.

Důležitou součástí tohoto algoritmu je rozdělení vstupních dat na učicí a testovací část.

Validace modelu na jiných než trénovacích datech je běžná i u jiných metod, v tomto případě je ale samotnou podstatou učení.

Koeficienty neuronů (a-f v uvedeném příkladě) jsou vypočítány na základě velmi malé trénovací množiny. Posouzení toho, jak dobře tyto neurony aproximují požadovanou funkci, se uskutečňuje na jiné množině. Rozdělení dat na učicí a testovací množinu je možné udělat na začátku, ale často se testovací množina určuje v každém kroku náhodně.

### Typy GMDH sítí

Obecně můžeme GMDH sítě rozdělit do dvou skupin, a to na parametrické a neparametrické.

Parametrické sítě jsou takové, u kterých se během učení určují koeficienty (tj. parametry) jednotlivých neuronů. Nejčastějším příkladem je síť MIA (Multilayered Iterative Algorithm), která je naznačena na obr. 1.3. V tomto případě jsou ve všech vrstvách polynomy stejného stupně.

V případě neparametrických sítí nelze mluvit o neuronové síti v původním významu, ale o algoritmu, který využívá princip výběru nejlepšího modelu. Příkladem je algoritmus pro shlukovou analýzu OCA (Objective Cluster Analysis).

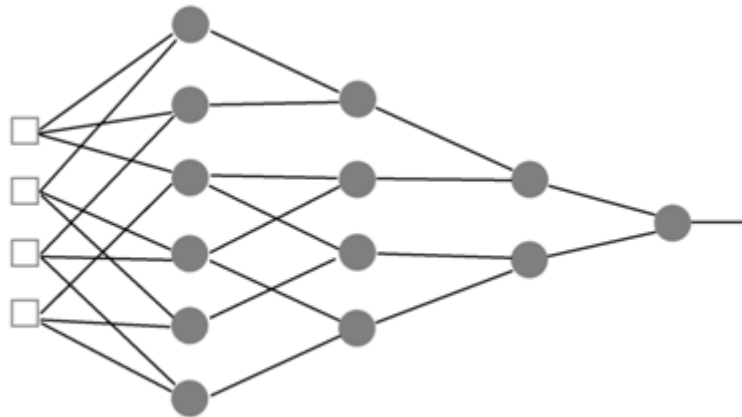
### Podobnost mezi topologií sítě GMDH a signálem EEG

Zajímavá je analogie mezi sítí MIA a činností mozku pozorované pomocí elektroencefalografu (EEG). O činnosti mozku je možné obecně získat jen velmi omezené informace.

Jednou z možností pro sledování funkce mozku je EEG.

V [17, 18] je popsáno tzv. vřeteno, což je častý útvar v záznamu EEG. Jedná se o zhruba 2 vteřiny dlouhý průběh v záznamu EEG, pro který je typické, že amplituda vln v první části narůstá až do určité hodnoty, následně po dosažení maxima dojde k jejímu postupnému poklesu. Průběh vřetena se blíží rovnici amplitudové mdulace.

K tomuto průběhu typicky dochází během procesu mentace (myšlení) a je pravděpodobně důležitý pro fixaci vzpomínek.



Obrázek 1.3: Ukázka topologie sítě MIA, jejíž uření bylo ukončeno po 4 krocích.

V síti MIA podobně dochází ke zvyšování počtu neuronů ve vrstvě, poté, co dosáhne maxima, počet neuronů klesá. Na konci procesu je naučená síť s jediným neuronem ve výstupní vrstvě.

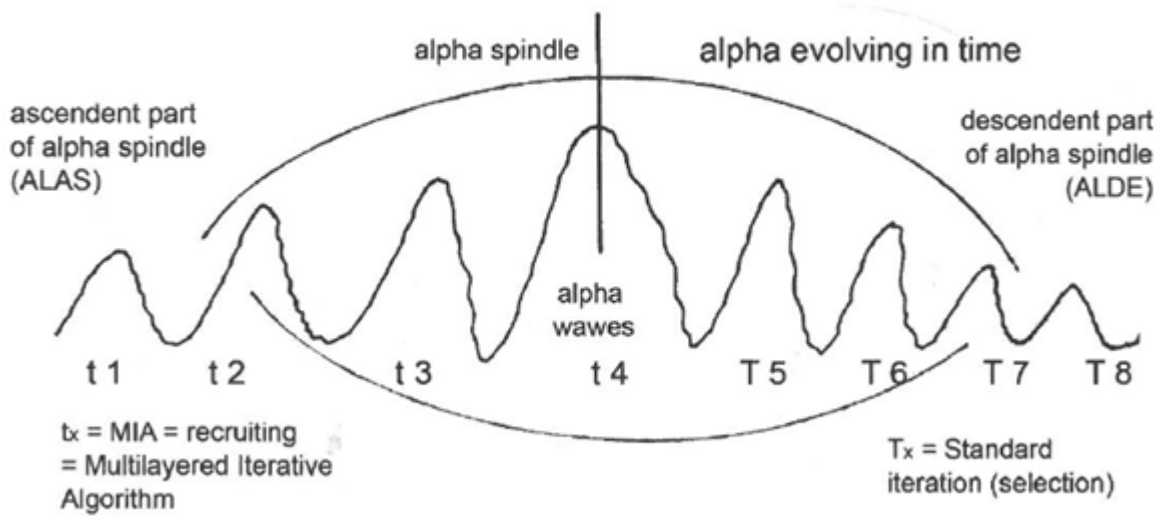
K podobnému procesu pravděpodobně dochází ve sloupcích kortexu (mozkových kolumnách). V každé alfa vlně se do učícího procesu zapojuje stále větší počet neuronů, což se v záznamu EEG projevuje narůstající amplitudou alfa vlny (obr. 1.4). Je proto možné, že by metoda GMDH mohla přispět k popisu některých mozkových funkcí.

### Využití sítí GMDH

V [19] je popsáno využití sítě typu GMDH pro identifikaci parametrů velkoplošného modelu znečištění ovzduší způsobeného větším množstvím zdrojů znečištění. Je použita modifikace sítě GMDH se zpětnou vazbou.

Tato modifikace spočívá v tom, že po prvním (částečném) naučení sítě je výstup této sítě vstupem pro učení v dalším kroku, čili dochází ke zvyšování dimenze vstupního vektoru. Kritériem pro ukončení procesu učení je hodnota Akaikeho informačního kritéria (AIC). Pokud se AIC zvyšuje, je trénování sítě zastaveno. Výsledky GMDH pro modelování znečištění jsou přesnější než v případě MLP nebo regresní analýzy.

Strukturu GMDH je možné kombinovat s jinými paradigmaty neuronových sítí. Spojení GMDH a radiálně bazických neuronů (RBF) je popsáno v [20, 21], Popsaná metoda je tam aplikována na analýzu lékařských obrázků se zaměřením na diagnostiku žaludku. Podle autorů má GMDH s radiálně bazickými neurony vyšší úspěšnost rozpoznávání než konvenční neuronové sítě s radiální bází.



Obrázek 1.4: Typický průběh alfa vlny během mentace. Na začátku dochází k postupnému zapojování neuronů, poté se počet aktivních neuronů postupně snižuje.

Kombinace GMDH a fuzzy neuronových sítí je popsána v [22]. Modifikace původního algoritmu spočívá v tom, že jednotlivé neurony nepočítají polynomiální funkce, ale funkce příslušnosti. Přenosové funkce neuronů zde nejsou polynomiálního typu, ale jsou použity radiálně bazické funkce (RBF). Tato síť byla aplikována v Japonsku na předpověď teploty. Výsledky fuzzy GMDH ve srovnání s klasickou RBF sítí byla pro tato data přesnější.

### 1.4.5 SONU (Second Order Neural Unit)

Výhodou sítí založených na McCullochových a Pittsových neuronech je jejich jednoduchost, ale tato jednoduchost je vykoupena tím, že výsledná síť je velmi komplikovaná. Z tohoto důvodu vznikla řada adaptací McCullochových a Pittsovových neuronů, ve kterých umělý neuron přebírá větší množství funkcí.

Jednou z možností je přidání zpětné vazby do neuronu, tj. přivedení výstupu neuronu zpět na vstup.

V [23] je představena myšlenka neuronových jednotek druhého řádu, tzv. SONU (Second Order Neural Unit). Tyto umělé neurony mají  $n$ -dimenzionální vstup  $x_n \in R_n$  a jediný výstup  $y \in R$  (zde  $R$  resp.  $R_n$  je množina reálných čísel). Vstupní vektor je  $x_n = (x_0, x_1, \dots, x_n)^T$ , kde  $x_0 = 1$  je konstantní vstup.

Funkce neuronu je popsána rovnicemi

$$u = x^T w x = \sum_{i=0}^n \sum_{j=i}^n x_i x_j w_{ij} \quad (1.19)$$

$$y = \theta(u) \in R^1 \quad (1.20)$$

kde  $\theta$  je nelineární přenosová funkce (např. sigmoida) a

$w$  je rozšířená matice vah. Jedná se o horní trojúhelníkovou matici, pod hlavní diagonálou jsou nuly.

$$w = \begin{pmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,n} \\ 0 & w_{2,2} & \dots & w_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_{n,n} \end{pmatrix} \quad (1.21)$$

McCullochův a Pittsův neuron je speciálním případem SONU, ve kterém je v matici  $W$  nenulový pouze první řádek. V tomto případě rovnice (1.19) přechází na

$$u = \sum_{i=1}^n x_i w_i \quad (1.22)$$

### Učení sítě SONU

V [24] je popsán algoritmus učení neuronu druhého řádu, který vychází z metody back propagation. Je použito stejné kritérium optimalizace pomocí střední kvadratické chyby, jaké se používá u MLP.

$$E(k) = \frac{1}{2} e(k)^2 = \frac{1}{2} [y(k) - y_d(k)]^2 \quad (1.23)$$

$y_d(k)$  je požadovaný výstup při vstupním  $k$ -tém vektoru

$y(k)$  je skutečný výstup při vstupním  $k$ -tém vektoru

$e(k)$  je hodnota chybové funkce.

Tato chyba je minimalizována pomocí změn váhové matice  $\Delta W$

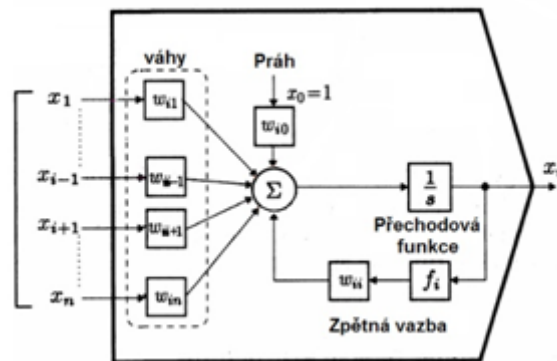
$$w(k+1) = w(k) + \Delta w(k) \quad (1.24)$$

Změna vah  $\Delta W$  je násobek ( $\mu$  je konstantní učící krok) gradientu chybové funkce:

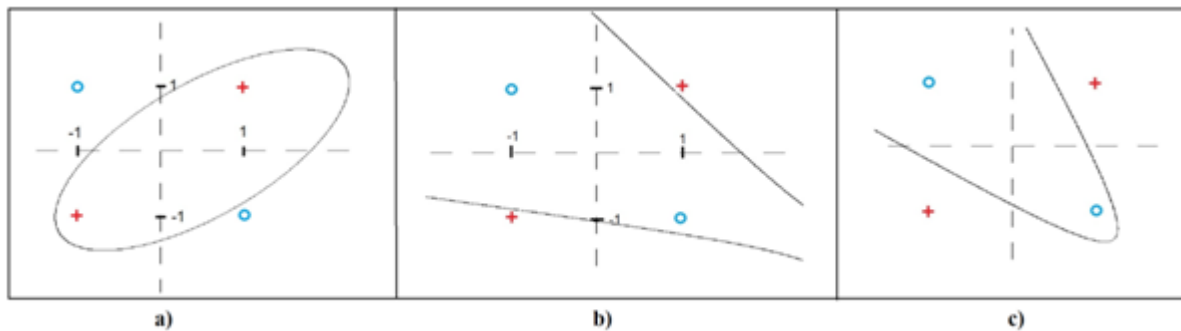
$$\Delta w = \mu \frac{\delta E(k)}{\delta w_n(k)} \quad (1.25)$$

Jeden neuron SONU může provádět stejné funkce jako několik McCullochových a Pittsovcových neuronů uspořádaných do vrstev. Neuron SONU má komplikovanější strukturu než McCullochův a Pittsov neuron, přičemž hlavní rozdíl spočívá v existenci zpětné vazby, jak je naznačeno na obr. 1.5. Díky tomu může SONU realizovat funkci XOR, což jednovrstvý perceptron neumí.

Na obr. 1.6 je ukázka tří možných způsobů, jak SONU naučená na funkci XOR rozdělí vstupní prostor do oblastí.



Obrázek 1.5: Schéma SONU (Second Order Neural Unit)



Obrázek 1.6: SONU se na rozdíl od McCullochova a Pittsova neuronu může naučit funkci XOR, ačkoliv nemá žádnou skrytou vrstvu. Existuje několik způsobů, jak může rozdělí vstupní prostor, např. elipsou, hyperbolou nebo parabolou. V tabulce 1.1 je uvedeno několik případů váhové matice  $w$ , které dělí vstupní prostor způsobem odpovídajícím tomuto obrázku

0,752	0,447	0,611	-0,921	0,511	0,348	-0,359	0,667	0,212
0	-0,706	-0,958	0	0,489	-0,698	0	0,428	-1,322
0	0	-0,630	0	0	0,010	0	0	0,850
<b>Elipsa</b>			<b>Hyperbola</b>			<b>Parabola</b>		

Tabulka 1.1: Váhy neuronu SONU, který rozděluje vstupní prostor na 2 lineárně neseperovatelné oblasti. Stejnou funkci může realizovat i klasická MLP, ale pouze za podmínky většího počtu neuronů uspořádaných do alespoň 3 vrstev.

### 1.4.6 HONNU

Rozšířením sítě SONU je síť HONNU (High Order Neural Network Unit), která je popsána v [25]. Na rozdíl od sítě SONU obsahuje více zpětných vazeb, proto je váhová matice  $w$  tří- nebo více rozměrná. Výstupní funkce sítě HONNU třetího resp. libovolného vyššího řádu je popsána rovnicí:

$$v_{cubic} = \sum_{i=0}^n \sum_{j=i}^n \sum_{k=j}^n w_{ijk} x_i x_j x_k \quad (1.26)$$

$$v_d = \sum_{i=0}^n \sum_{j=i}^n \dots \sum_{q=p}^n \sum_{r=q}^n w_{ij\dots qr} x_i x_j \dots x_q x_r \quad (1.27)$$

Výhodou sítí HONNU oproti klasickým neuronovým sítím pracujícím na principu zpětného šíření je nízký počet neuronálních vah a z něj vyplývající jednoduchost explicitního popisu struktury. Další výhodou je možnost implementace apriorních znalostí o modelovaném procesu do struktury sítě.

Neuron HONNU je komplikovanější než McCullochův a Pittsův neuron, proto síť tvořená těmito neurony obsahuje méně neuronů.

---

## Simulace mozkových funkcí

Snaha napodobit lidský mozek stála za vznikem a rozvojem neuronových sítí a i v dnešní době je tato metoda předmětem zájmu pro výzkumy týkající se modelování mozku. V minulosti byl hlavním omezením nedostatek výpočetního výkonu, přesto vznikalo mnoho odvážných teorií, jak modelovat složité duševní pochody, včetně intuice [26]. Růst výpočetního výkonu značně posunul možnosti neurosimulací, nicméně během posledních zhruba deseti let se stále více potvrzuje předpoklad, že ani výkonný hardware nezaručí vysvětlení takových jevů, jako je vědomí nebo uvažování.

V následující části jsou popsány 4 významné projekty zaměřené na modelování duševních procesů.

### 2.1 Blue Brain projekt

Projekt Blue Brain byl zahájen v roce 2005 jakožto spolupráce mezi profesorem Henry Markramem z Brain Mind Institute a IBM. Cílem tohoto projektu bylo detailní modelování kolumn (sloupců) neokortexu a následně celého mozku [27, 28]. Kolumna neokortexu představuje základní funkční jednotku mozkového kortexu.

Předstupněm projektu Blue Brain byla detailní (s přesností na milisekundy) simulace změn akčních potenciálů STDP (Spike-Timing-Dependent Synaptic Plasticity) provedená v roce 1993, díky které byl popsán princip komunikace mezi neurony, a následná simulace zjednodušené kolumny. Tyto experimenty probíhaly pod záštitou Brain Mind Institute (BMI).

Projekt Blue Brain (nyní často označován jako HBP – Human Brain Project) byl rozdělen do dvou fází, přičemž cílem první fáze bylo sestavit detailní model kolumny kortexu



savce (konkrétně byla vybrána 2 týdny stará krysa). Mozková kolumna je topologické a funkční seskupení neuronů, které má přibližně tvar válce o výšce 2 mm a průměru 0,5 mm. Obsahuje přibližně 10 000 neuronů u krysy a 60 000 neuronů u člověka. Cíl této první fáze byl dosažen v listopadu 2007.

Cílem druhé fáze byla simulace celého mozku, přičemž podle H. Markrama by měl být tento stav dosažitelný do roku 2018. V rámci této druhé fáze se rozlišují dvě větve, v první řadě je naplánována simulace na molekulární úrovni, která umožní studovat projevy genetické informace na funkci neuronu.

Druhá větev je zaměřena na paralelní simulaci velkého množství propojených kolumn. Ve výsledku by takto mělo být možné simulovat celý mozek, ve kterém je zhruba milion kolumn.

Dle H. Markrama budou v detailní simulaci celého mozku probíhat stejné procesy jako v lidském mozku, tj. myšlení, kreativita nebo vnímání. S využitím této simulace by pak bylo možné např. testovat vliv léků nebo zkoumat psychické poruchy. Tento názor je ale mnoha odborníky zpochybňován s poukazem na skutečnost, že pouhým „okopírováním hardwaru“ nedojde k emergenci duševních procesů.

Provedená simulace je natolik detailní, že jakožto základní prvky jsou v ní uvažovány jednotlivé molekuly. Důsledkem této podrobnosti je obrovská výpočetní náročnost, kvůli které bylo nutno použít pro výpočty jeden z nejvýkonnějších superpočítačů BlueGene/Q s 65 536 procesory a výkonem 715 teraflops. Tento superpočítač byl podle žebříčku TOP 500 v červnu 2015 v pořadí stým nejvýkonnějším počítačem na světě.

Podobná detailní a realistická simulace naráží na celou řadu vážných problémů. Jedním z nich je paralelismus výpočtu. Mozek sám je výrazně paralelní struktura, ale při simulaci na klasickém mnohajádrovém počítači vzniká problém se synchronizací a výměnou informace, neboť klasický počítač je ve své podstatě sekvenční zařízení [29, 30]. U multi-processorových systémů může výměna dat mezi procesory být kritickým místem.

V [31] je popsáno simulační prostředí zvané NEURON, které umožňuje téměř lineární urychlení výpočtu při paralelizaci simulace až do zhruba 100 procesorů, při více než 500 procesorech je již zrychlení s každým přidaným procesorem výrazně nelineárně menší. BlueGene přitom používá 65 536 procesorů, z čehož vyplývá, jak velké jsou komplikace při paralelizaci. Bylo navrženo několik řešení, jako např. konstantní doba synaptického přenosu. Nevýhodou tohoto řešení je, že vytvořený model se vzdaluje od své biologické předlohy.

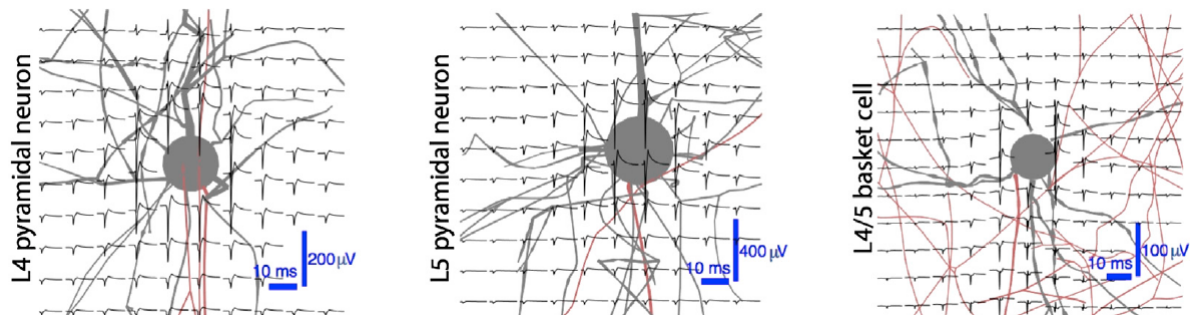
### 2.1.1 Shoda mezi pozorovanými projevy mozku a modelem pomocí měření potenciálu

Vzhledem ke skutečnosti, že projekt Blue Brain se snaží o detailní simulaci mozku, bylo nezbytnou podmínkou disponovat metodou pro porovnání modelu a vzoru. Vzhledem ke komplikovanosti mozku není možné použít elektroencefalograf (EEG), neboť tento signál je zprůměrován přes velké množství neuronů a neumožňuje proto detailně sledovat děje probíhající v jedné konkrétní buňce.

Metodou, která umožňuje detailně nahlédnout do dějů probíhajících v neuronu, jsou LFP (Local Field Potentials), což jsou změny elektrického potenciálu v okolí buněk. Toto měření je natolik přesné, že je možné změřit potenciál samostatného neuronu. S pomocí LFP pomocí je možná kalibrace navrženého modelu. Na druhou stranu s LFP není možné určit stav všech neuronů, nýbrž pouze velmi malé části neuronové sítě.

Druhou nevýhodou je, že se jedná o invazivní metodu (na rozdíl od EEG).

V [32] je popsána metoda projektu Blue Brain pro modelování LFP v malé části mozku, která se skládá jen z 12 000 modelů pyramidových neuronů a v níž je jen 15 000 000 synapsí.



Obrázek 2.1: Modelové akční potenciály v okolí nervových buněk ze 3 různých vrstev, které byly vybuzeny 10 ms prodovým pulsem o napětí  $400 \mu V$ . Soma a dendrity jsou šedé, axon je červený. Nejvýraznější odezvu má neuron 4 vrstvy.

### 2.1.2 Topologická analýza neuronové sítě

V biologické neuronové síti jsou neurony organizovány do různých skupin, přičemž nejlépe viditelnou strukturou jsou kolumny. Vazby mezi neurony jsou ale ve skutečnosti podstatně složitější a jejich poznání je podmínkou pro vytvoření realistického umělého modelu.

Toto téma je rozpracováno v [33, 24], kde je popsána organizace neuronů uspořádaných nejen do kolumn a vrstev, ale také do topologicky ohraničených shluků (clusterů). Neurony

uvnitř těchto shluků jsou výrazně více propojeny, než mezi shluky. Jedním z cílů projektu Blue brain bylo nasimulovat seskupování izomorfně organizovaných neuronů do shluků. Možným návodem je pravidlo společných sousedů (common neighbor rule), podle kterého pravděpodobnost existence synaptického spojení mezi dvěma neurony roste s tím, kolik mají společných sousedů (soused znamená buňku, která má s referenční buňkou synaptické spojení, nejedná se tedy o sousedství v topologickém smyslu).

V provedené simulaci se neurony nacházejí ve „virtuální 3D krychli“, přičemž každá strana krychle je spojena s protilehlou stranou. Každý neuron má díky tomuto uspořádání stejný počet sousedů.

V simulaci se vyskytují různé typy neuronů, přičemž základním typem je pyramidový neuron páté vrstvy[35]. Cílem je ověřit vliv tří parametrů – velikosti sítě, hustoty neuronů a morfologického dosahu – na vlastnosti shlukování (počet shluků a počet neuronů ve shluku).

Velikost sítě se pohybovala mezi 512 a 5832 neurony. Četnost velikosti shluku jakožto funkce velikosti sítě má přibližně logaritmicko-normální průběh, nejdříve roste a od určité hodnoty začíná klesat. Vliv velikosti sítě není příliš výrazný.

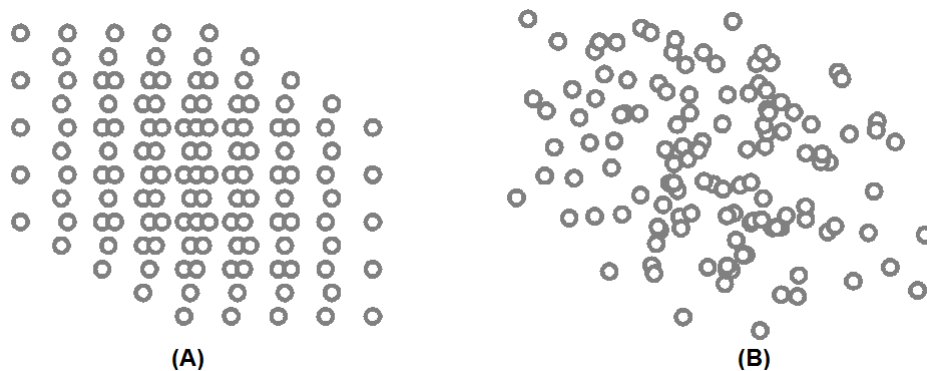
Morfologický dosah je veličina popisující rozvětvení neuronu. Čím větší je rozvětvení neuronu tím více má společných sousedů a tím větší je jeho morfologický dosah. Jedná se o abstraktní veličinu, jakožto referenční hodnota byl zvolen pyramidový neuron s morfologickým dosahem 1. Pokud je morfologický dosah v intervalu (0,5-1), počet neuronů ve shluku roste přibližně lineárně, v okolí 1 je přibližně konstantní a poté klesá. V simulaci se hodnota této veličiny pohybovala od 0,5 do 1.

Hustota neuronů má největší vliv tehdy, pokud je její hodnota nízká. V tom případě rychle roste počet shluků a zároveň klesá počet neuronů ve shluku. Pro vyšší hodnoty je organizace neuronů téměř nezávislá na hustotě.

Otázka shlukování neuronů má velký význam pro neurobiologické simulace. V takovýchto simulacích není účelem dosáhnout co nejrychlejší splnění dané funkce, ale provést jí způsobem, který je podobný procesům v mozku. To by nebylo možné bez studia konektivity neuronů jak v reální biologické neuronové síti, tak v simulace.

Tato simulace vychází z předpokladu, že na základě vlastností neuronů je možné určit počet neuronů ve shlucích. Je ale třeba podotknout, že shlukování neuronů je ovlivněno podstatně větším množstvím veličin, než uvedené tři parametry. Zajímavým poznatkem je, že velikost clusterů neroste neomezeně, ale směřuje k určitým limitním hodnotám. Tyto limitní hodnoty by zřejmě měly být dodrženy ve všech simulacích duševních funkcí. Ukazka shlukování je na Obr. 2.2.

Je třeba rozlišovat mezi shlukováním na makro a mikro úrovni. Makroskopická úroveň (tím je myšlena úroveň mozkových areí) se vyznačuje tím, že shluky jsou organizovány centrálně. Jednotlivé součásti těchto makro-shluků mají rozdílné funkce. Naproti tomu v mikro shlucích se vyskytují homogenní neurony. Nevyskytuje se mezi nimi žádná centrální nebo řídicí oblast. Předmětem zájmu této dizertace je zejména mikro úroveň.



Obrázek 2.2: Příklad shlukování neuronů mezi původním uspořádáním (A) a uspořádáním po přivedení signálu s proměnlivou frekvencí (B) zobrazené v projekci z třídímního prostoru

## 2.2 SyNAPSE

Projekt SyNAPSE (Systems of Neuromorphic Adaptive Plastic Scalable Electronics) je součástí programu vedeného institucí DARPA, který je zaměřen na vývoj nové počítačové architektury inspirované biologickými systémy. Jedná se o společný projekt několika univerzit a firem (Stanford University, Boston University, IBM a dalších), které jsou zastřešeny DAR-Pou.

Způsob zpracování informace v architektuře současných počítačů je zcela odlišný od biologických neuronových sítí. V biologické neuronové síti není rozdíl mezi pamětí a výpočtem, každý neuron má synaptické váhy představující paměť a zároveň realizovanou funkci.

Ve standardní umělé architektuře je naproti tomu paměť oddělena od procesoru. Toto je budoucí omezení růstu výpočetního výkonu, neboť i kdyby se rychlost procesoru nadále zvyšovala podle Moorova zákona, bude rychlost přenosu mezi pamětí a procesorem omezujícím faktorem rychlosti výpočtu.

Na rozdíl od projektu Blue Brain není úkolem projektu SyNAPSE modelování mozkových funkcí (ačkoliv autoři často přirovnávají velikost simulované sítě k části mozku určitého

živočicha – krysy, kočky, makaka [36], ale využití biologické inspirace pro vytvoření nové architektury počítačů, ve které:

- soma neuronu provádí výpočet (místo procesoru)
- synapse nahrazují paměť
- axony a dendrity slouží pro komunikaci (místo sběrnice).

SyNAPSE využívá superpočítač C2 z IBM pocházející z pobočky Almaden, který umožňuje simulovat neuronovou síť s 55 miliony neuronů a 442 miliard synapsí, odpovídající kortexu krysy. Použitý model neuronu je ale silně zjednodušen proti biologickému neuronu, proto nelze tvrdit, že se jím podařilo simulovat mozek krysy.

Tento systém má dva typy řízení:

- neurony jsou řízeny časem,
- synapse jsou řízeny událostmi.

Stav neuronů je počítán v každém kroku, synapse se aktivují pouze po přijetí signálu. Signál je na synapsi podržen po dobu odpovídající zpoždění signálu, která je závislá na vzájemné vzdálenosti neuronů. Zpoždění signálu v axonu může být 1 až 20 ms. Všechny vytvořené neurony pracují s dynamikou 1 ms. Modelový čas je přibližně 9 krát pomalejší než je reálný čas.

Kortex byl mezi ostatními částmi mozku vybrán proto, že se zde odehrává většina duševních procesů. Celková plocha kortexu krysy je  $600 \text{ mm}^2$ , počet neuronů na  $1 \text{ mm}^2$  je u ní zhruba 9200 a je přibližně konstantní. Přibližně 80 % neuronů je zde excitačních, ostatní jsou inhibiční. Inhibiční neurony mohou být spojeny pouze s excitačními neurony, zatímco excitační neurony mohou být spojeny s čímkoliv. Kortex je simulován na superpočítači typu BlueGene/L s 32 768 procesory [33]. V porovnání s mozkem vykazuje superpočítač C2 i BlueGene/L odlišnosti, z nichž nejvýraznější je fakt, že mozek pracuje ve spojitém čase, zatímco superpočítačové simulace jsou digitální.

Projekt je rozdělen do 5-ti fází, ve kterých:

**fáze 0** měla za cíl ověřit navrženou architekturu neuronové sítě, která by umožnila simulaci s modelovou frekvencí vyšší než 10 Hz, spotřebou energie méně než 10 J na jeden synaptický výboj a která by tyto parametry byla schopná zachovat až do velikosti sítě  $3 \times 10^8$  synapsí. Tato fáze byla ukončena v srpnu 2009.

**fáze 1** se týkala definice konkrétního čipu pro simulaci, ve které bude při velikosti sítě 1014 synapsí a 1010 neuronů spotřeba energie maximálně 1 kW.

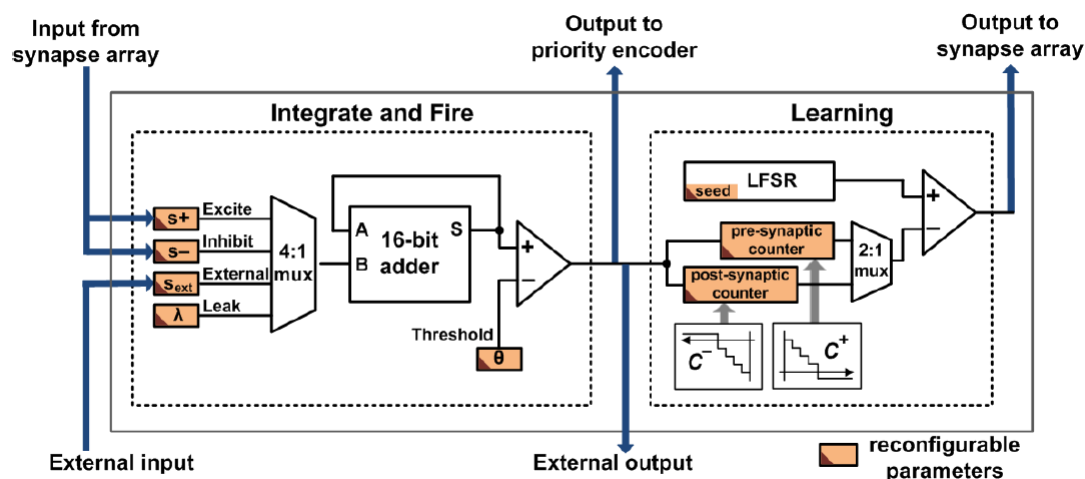
**fáze 2** vedla k návrhu čipu, ve kterém mělo být 1010 synapsí a 106 neuronů na  $cm^2$ . V rámci této části projektu byly studovány vlivy soupeření o prostředky mezi neurony na celkový výsledek neuronové sítě.

**fáze 3** se týkala návrhu neuronového systému s 1012 synapsemi a 108 neurony. Celý tento systém měl být integrován do jediného čipu.

**fáze 4**, která dosud neskončila, má za cíl instalovat čip s 108 neurony do robota, a ověřit funkčnost simulace na úrovni kočky.

Ve všech fázích je kladen důraz na praktickou využitelnost nově navržených velmi výkonných počítačů. Neočekává se využití v lékařství nebo pro základní výzkum mozku jako v projektu BlueBrain, ale pro návrh nové počítačové architektury.

Praktickým výsledkem je např. chip integrující neuronovou síť skládající se ze neuronů typu „Integrate and Fire“. Tento chip má spotřebu energie pouze 45 pJ na výboj [34]. Je v něm úzce spojen proces výpočtu a paměti, což má za následek kromě nízké spotřeby energie také vysokou škálovatelnost. V tomto čipu o rozměru 45 nm se nachází 256 neuronů a 64 000 synapsí. Na rozdíl od většiny předchozích modelů jsou synapse zde binární, nikoliv analogové. Princip neuronu na tomto čipu [35, 36] je ve stručnosti popsán na obr. 2.3.



Obrázek 2.3: Schéma neuronu vestavěného do 45nm čipu. V modelu se nachází celkem 8 nastavitelných parametrů. LFSR (linear feedback shift register) je generátor náhodných čísel nutný pro pravděpodobnostní učení parametrů neuronu.

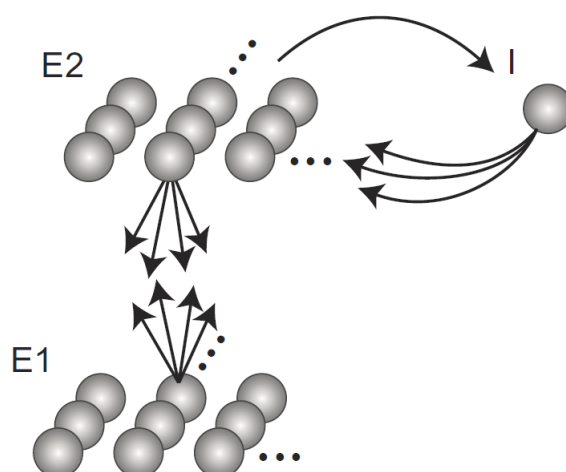
Tento chip byl otestován na úlohách klasifikace vzorů, rozpoznávání a auto-asociativních úlohách. Nejlepší výsledky tato síť přinesla v otázkách autoasociace, kde dosáhla téměř maximální teoretické kapacity Hopfieldovy sítě.

Zkušenosti s tímto chipem naznačují, že hardware založený na neuromorfologických principech může v budoucnu výrazně překonat von Neumanovu architekturu.

Na druhou stranu je třeba podotknout, že navzdory úspěchům projektu SyNAPSE je spotřeba energie u těchto nově vyvinutých počítačů stále výrazně vyšší než u lidského mozku (řádově milionkrát). Mozek má navíc podstatně komplikovanější strukturu.

V [37] je popsán návrh neuronové sítě pro vybavování vzorů na základě on-line učení bez učitele. Dlouhodobým cílem tohoto výzkumu je detailní popis toku informace v mozku. Živé organizmy disponují množstvím nejrůznějších senzorů, pomocí kterých získávají mnoho typů informací o okolí. Navzdory obrovské variabilitě smyslových vstupů jsou všechny tyto signály převedeny na jeden univerzální formát v podobě výboje neuronu. Mozek je schopen s takto širokou škálou vzorů pracovat a vytěžit smysluplné informace v situaci, ačkoliv se jedná o učení bez učitele a zpracování informace je paralelní.

Snaha o napodobení této funkce vedla k návrhu sítě, která se skládá ze dvou vrstev neuronů typu „Integrate and Fire“, přičemž tyto vrstvy jsou obousměrně propojené (obr. 2.4). Vstupem a zároveň výstupem sítě je vrstva excitačních neuronů E1, skrytá vrstva excitačních neuronů je označena E2 a inhibiční neuron jako I. Skrytá vrstva E2 obsahuje stejný počet neuronů jako vrstva E1.



Obrázek 2.4: Reprezentace neuronové sítě určené pro samoučení

Relativně jednoduchý učící algoritmus byl optimalizován pro rozpoznávání vzorů re-

prezentovaých řídkou maticí, ve vstupní vrstvě E1 bylo 100 neuronů, ale vzorů bylo jen 9. Problém spočíval v tom, že se vzory překrývaly a navíc byl na vstup přiveden bílý šum. Během 500 iterací se podařilo dosáhnout úspěšnosti vybavování přes 90 %. Zajímavým zjištěním, je, že úspěšnost není příliš ovlivněna šumem.

Přestože tato síť nebyla otestována s reálnými daty, výsledky naznačují, že se jedná o správný směr v oblasti neurovýpočtů. Důležitou vlastností sítě je škálovatelnost a modularita – tuto neuronovou síť lze použít jako stavební kostku pro výrazně komplikovanější sítě schopné řešit reálné aplikace.

### 2.2.1 GENESIS

GENESIS (GEneral NEural SIMulation System) je prostředí navržené pro simulaci neurobiologických procesů s důrazem na škálovatelnost. V projektu GENESIS je možné simulovat děje uvnitř buňky i velké neuronové sítě [43].

Zajímavé jsou pro nás hlavně tzv. realistické modely, což jsou modely založené na známých faktech o organizaci a fyziologii neuronů a o neuronových sítích. Zpravidla se zde používají různá rozšíření Hodgkinova a Huxleyova modelu. Vzhledem k typické náročnosti realistických simulací neuronových sítí je obvykle požadován paralelní výpočet.

Projekt PGENESIS (Parallel GENESIS) je rozšířením projektu GENESIS, který umožňuje spuštění na superpočítači s podporou MPI (multiprocessor interface) nebo PVM (Parallel Virtual Machine).

### 2.2.2 NEST

NEST je iniciativa, která byla založena za účelem dlouhodobé spolupráce výzkumů týkajících se simulací velkých neuronových sítí a sdílení znalostí [44].

Výsledkem je open-source program umožňující simulaci různě složitých typů neuronů. Velikost sítě v tomto programu je až jedna miliarda neuronů (stav v roce 2015), díky čemuž může mít každý neuron realistický počet vstupů, pro reálné vrstevnaté struktury, ale tato velikost sítě stále nestačí. Projekt NEST podporuje využití heterogenity neuronů a synapsí.

Jedním z požadavků projektu NEST je opakovatelnost pokusů, což je typický problém pro řadu simulací velkých neuronových sítí. Vzhledem k relativní jednoduchosti matematických abstrakcí biologického neuronu vzniká řada programů, které nejsou dostatečně zdokumentovány a u nichž není možné opakovat testy. U projektu NEST je výsledek testů stejný bez ohledu na počet procesorů, operační systém apod.



Velký důraz je zde kladen na uložení a obnovení stavu synapsí, protože v reálných sítích jejich počet řádově převyšuje počet neuronů, a proto je nutné jim vyhradit větší výpočetní prostor a mnohem větší paměť.

---

# Navržený model homeostatického neuronu

## 3.1 Východiska navržené neuronové sítě

### 3.1.1 Model rozhodovacích procesů lidského operátora

Lidský operátor je účinný a univerzální regulátor, který ve většině činnostech nemůže být nahrazen automatickým systémem. Na rozdíl od automatických regulátorů je např. schopen adaptovat se na proměnné podmínky, analyzovat velké množství odlišných scén a rozhodovat se v situacích, se kterými se nikdy předtím nesetkal. Ve srovnání s automatickým regulátorem má ale také nevýhody, zejména nedostatečnou odolnost vůči únavě, stresu, emočním stavům, agresivitě, nepředvídatelnost, pomalejší a méně přesnou reakci apod.

Pro popis faktorů ovlivňujících funkci lidského operátora a návrh asistenčních systémů by bylo výhodné disponovat modelem řidiče umožňujícím snadné otestování velkého množství technických řešení. V současné době existuje řada modelů, které se zpravidla soustředí na velmi specifickou oblast. Tyto modely se používají pro návrh praktických aplikací využívaných v dopravě, jako jsou regulátory pro udržování konstantní vzdálenosti od předchozího vozidla, pro vedení vozidla v jízdním pruhu nebo udržování kurzu letadla. Ve většině situací se ale řidič chová jinak než automatický regulátor. Vytvoření komplexního modelu řidiče by tak přispělo k porozumění vlivům ovlivňujícím rozhodování, a tím i bezpečnost, neboť většina nehod je způsobena chybou řidiče. Příkladem činností, které není možné jednoduše modelovat pomocí klasických metod je např. rozhodnutí o předjetí jiného vozidla, predikce úrovně pozornosti, volba jízdní strategie nebo volba trasy.

Tvorba modelu mentálních funkcí lidského operátora naráží na několik problémů. Prvním z nich je velké množství dat, které musí lidský operátor zpracovat. Největší je tok obrazových dat, neboť v lidském oku se nachází přibližně 100 milionů světločivných buněk s časovou rozlišovací schopností řádově 10 vjemů za vteřinu. Ve skutečnosti je ale možné tuto informaci výrazně zkomprimovat, což také lidský mozek provádí. Přesto se jedná o obrovský tok informací, který je obtížné zpracovat. Navíc kromě zrakových vjemů přijímá řidič zvuk, zrychlení a další jevy.

Druhým problémem je nedostatek znalostí o chování lidského operátora. Určité činnosti je sice možné popsat pomocí matematických rovnic (např. udržování odstupu), ale většinu rozhodnutí nejsme schopni formálně vyjádřit. Činíme komplikovaná a kvalifikovaná rozhodnutí, dokonce často správnější než v případě automatického systému, ale nejsme schopni popsat, proč jsme se tak rozhodli. Na rozhodování má vliv zkušenost i vrozené dispozice, přičemž často nejsme schopni odlišit, která část našich znalostí je vrozená, a která získaná. Na druhou stranu je možné získat dostatečné množství dat o rozhodovacích procesech pomocí sledování člověka v reálném provozu nebo na simulátoru. Z obou důvodů (velké množství dat a komplikovanost formálního popisu) se neuronové sítě jeví jako vhodná metoda pro modelování rozhodovacích procesů řidiče. Neuronové sítě se neprogramují, nýbrž učí na naměřených datech, a tím se blíží lidskému mozku. O procesu rozhodování lidského operátora můžeme poměrně snadno získat velké množství dat, ale extrakce znalostí z těchto dat je velmi obtížná.

Z důvodu komplikovanosti struktury lidského mozku nelze jednoduše popsat tok informace. Je ale možné se alespoň principiálně přiblížit procesům, k nimž dochází při zpracování informace a rozhodování. Velmi zjednodušený příklad toku dat v lidském mozku je na obr. 3.1.

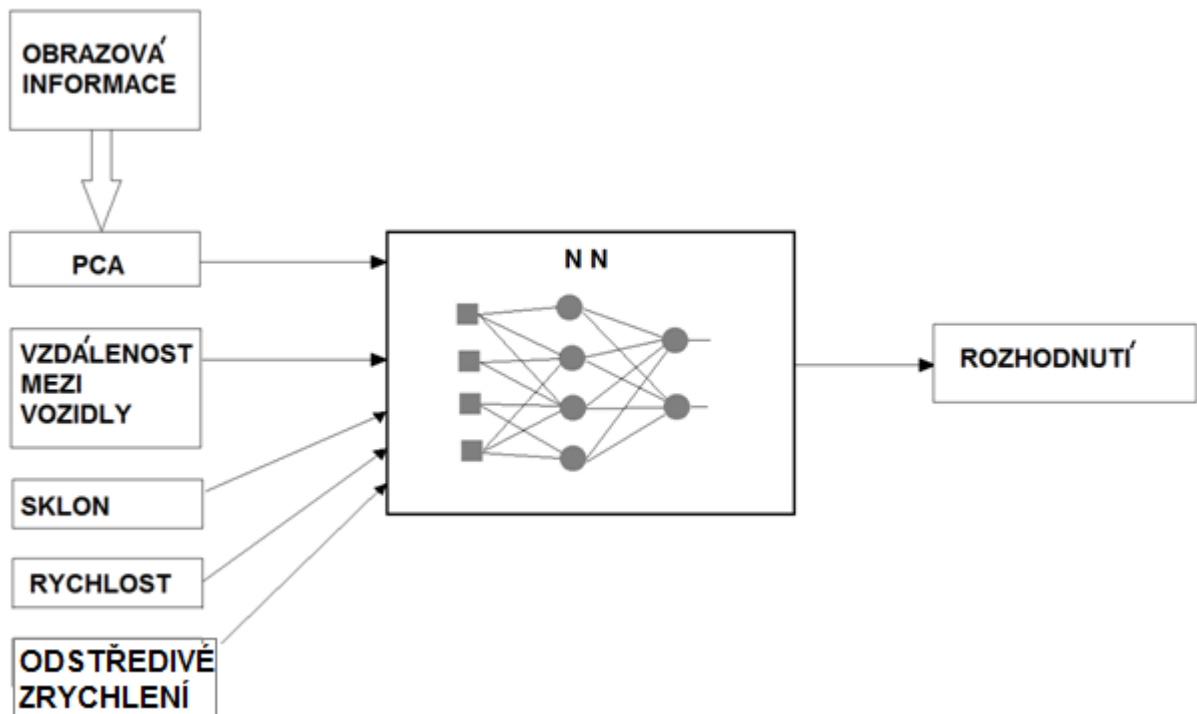
### 3.1.2 Úrovně řízení

Z hlediska modelovaných procesů je důležité definovat také hierarchickou strukturu.

Přestože cílem této práce není komplexní model řidiče, je výhodné alespoň rámcově vymežit oblast modelovaných jevů. Při tom je možné využít již prozkoumané hierarchické modely řidiče, které lze zobecnit na širokou škálu člověkem prováděných činností.

Při modelování řidiče se zpravidla užívá dělení činností do 3 hierarchických úrovní dle J. Rasmussena [30]:

1. naučené stereotypy, automatické činnosti, dovednosti



Obrázek 3.1: Příklad vstupů, vlastní neuronové sítě a výstupu při procesu rozhodování. V reálném světě je tento proces komplikovanější, např. velkou roli hrají minulé stavy (které mohou být uloženy v NN)

2. řízení založené na pravidlech

3. řízení založené na znalostech

Na první úrovni se nachází základní činnosti jako je rozjezd vozidla, udržování rychlosti, udržování odstupu. Tyto činnosti lze obecně dobře popsat a modelovat pomocí zpětnovazebních smyček.

Na druhé úrovni se nachází činnosti, které jsou nezbytné v případě interakce vozidla s jinými vozidly. V těchto případech se řidič rozhoduje podle dopravních předpisů, přičemž toto rozhodování lze zapsat ve formě logických implikací typu: nastane-li situace A, proved' akci B. Problémem je, že mohou nastat situace, které nejsou popsány pomocí pravidel. V takovýchto případech se uplatňuje nejvyšší úroveň, řízení pomocí znalostí.

Třetí úroveň popisující řízení na základě znalostí se týká činností jako je volba jízdní strategii, predikce dopravní situací apod. V těchto případech se většinou pracuje s neúplnými

daty a velkou roli zde hraje zkušenost a intuice, což jsou obtížně formalizovatelné procesy.

Rozdělení činností do 3 hierarchických úrovní není přesné a jednotlivé činnosti nelze jednoznačně přiřadit do kategorií.

#### 3.1.3 Požadavky na neuronovou síť

Ze záměru použití umělých neuronových sítí pro modely mentálních funkcí řidiče vyplývají požadavky na neuronovou síť, která musí být

- dostatečně jednoduchá, aby bylo možné jí simulovat na dostupném hardwaru.
- dostatečně podrobná, aby věrně korespondovala s biologickou předlohou

Oba požadavky jdou proti sobě, neboť biologická neuronová síť je velmi složitá struktura, kterou nelze přesně napodobit. Návrh neuronové sítě tak bude vždy hledáním kompromisu mezi jednoduchostí a přesností.

Z praktického hlediska je požadováno, aby program byl schopen zpracovávat data v reálném čase při použití dostupného procesoru (frekvence řádově do 4 GHz), např. při zpracování EEG dat by program měl být schopný zpracovat vstupy během 1 vteřiny.

Všechny pokusy o modelování mozkových funkcí narážejí na problém velikosti sítě, který vede ke kompromisu mezi přesností popisu neuronu a velikostí neuronové sítě. Pro realistické modely je nutné vytvořit model na detailní úrovni, kde vysoká složitost neuronu brání vytvoření modelu s dostatečným počtem výkonných prvků. Příkladem takového přístupu je projekt Blue Brain, který požaduje pro realistický model jediného neuronu výkon přibližně jednoho procesoru o 1 GHz. Na druhé straně stojí modely s velkým počtem neuronů, které ale kvůli své jednoduchosti nemohou věrně modelovat biologickou neuronovou síť. Příkladem je projekt SyNAPSE, který vytváří model neuronové sítě se srovnatelným počtem prvků jako má lidský kortex, ale neurony jsou silně zjednodušené.

Je zřejmé, že se v dohledné době nepodaří napodobením biologické neuronové sítě nahradit lidského operátora. Je ale možné využít umělé neuronové sítě ke studiu určitých, poměrně specifických činností. Jedná se např. o analýzu scény, identifikaci předmětů na scéně, volba trasy, detekci ztráty pozornosti apod.

#### 3.1.4 Kontinuita učení a odolnost vůči „catastrophic interference“

Neuronové sítě založené na učící metodě back propagation obvykle trpí problémem známým pod označením „catastrophic interference“, což je ztráta funkčnosti již naučené neuronové

sítě poté, co dojde k mírné změně vstupních vzorů. Tento problém se týká neuronových sítí, které byly naučeny na určité vzory. Pokud je k již naučeným vzorům přidán nový vzor, neuronová síť je schopná se na tento nový vzor naučit, ale ztrácí schopnost rozpoznávat dříve naučené vzory.

Odolnost vůči tomuto jevu sice není explicitně uvedena v požadavcích v bodu 3.1.3, ale vyplývá z druhé podmínky, neboť biologické neuronové sítě mají schopnost učit se nové vzory aniž by přitom zapomněli již naučené.

#### 3.1.5 Výběr problémových oblastí vhodných pro realizaci modelu

Vzhledem k zaměření této práce se detekce pozornosti lidského operátora jeví jako velmi zajímavá a užitečná aplikace navržené neuronové sítě. Detekce úrovně pozornosti lidského operátora je v současnosti předmětem intenzivního výzkumu mnoha oborů, které se ve většině případů snaží aplikovat lékařské znalosti pro technické využití.

Jednou z možností sledování úrovně pozornosti je měření elektrického potenciálu generovaného nervovými buňkami, což je velmi účinná metoda umožňující získat aktuální informace o stavu, v jakém se sledovaný jedinec nachází. Nevýhodou EEG (elektroencefalografie) je

- vysoká komplexnost vztahů mezi sledovaným signálem a skutečným stavem jedince
- makroskopický charakter tohoto signálu, ve kterém na každou elektrodu připadají miliony neuronů.

Přes tyto nevýhody je EEG slibnou metodou pro monitorování úrovně pozornosti, neboť se jedná o dostupnou metodu s velkou vypovídací schopností.

O vhodnosti využití neuronových sítí pro analýzu nebo modelování EEG svědčí intenzivní výzkum věnovaný tomuto tématu. V [47, 48] jsou popsány metody pro automatickou analýzu EEG, které se týkají lékařského využití, ale je možné, že by použité metody mohly být adaptovány i pro technické oblasti.

V [49] je popsána metoda pro hledání a rozpoznávání vzorů v EEG záznamu. Tato metoda, která je poměrně obecná a tudíž i univerzální, by mohla být použita pro automatickou analýzu funkce lidského operátora.

V dostupné literatuře se nejčastěji pracuje s využitím EEG pro lékařské účely. Tyto metody by zřejmě bylo možné upravit pro technické využití, ale je třeba mít na paměti,

že praktické využití analýzy EEG se od obvyklého použití v lékařství liší minimálně tím, že:

- předmětem zájmu je zdravý jedinec
- je vyžadována odpověď v reálném času. Nejsou k dispozici data z času budoucího, než je čas v němž je prováděna analýza.

Obecně lze říci, že analýza EEG pro lékařské využití je v současné době na vyšší úrovni než v technické oblasti. Je možné využít postupy používané v lékařství, ale je nutné je vždy výrazně upravit (tato úprava spočívá hlavně v zobecnění postupů, které musí být platné pro podstatně širší škálu stavů

Mým obecným cílem je modelování duševních procesů lidského operátora, ale v praxi jsem se soustředil zejména na řidiče. Důvodem je, že Fakulta dopravní ČVUT disponuje simulátorem vozidla umožňujícím pořizování současného záznamu EEG a jízdních parametrů. Navíc je zde možné čerpat z odborného zázemí a znalostí získaných dlouhodobým studiem interakce řidič – vozidlo.

Vzhledem k obtížnosti modelování a analýzy EEG signálu je ovšem zřejmé, že prvním krokem k využití navržené neuronové sítě musí být aplikace zaměřená na více specifickou oblast procesů řidiče. Z množství procesů, které připadali v úvahu (např. volba odstupů, příčné řízení nebo vyhodnocení situace na neřízené křižovatce) jsem si vybral analýzu obrazové scény. Pro tuto oblast jsem se rozhodl jednak díky snadné dostupnosti trénovacích a testovacích dat, a také proto, že je předmětem intenzivního výzkumu s využitím různých metod, a tudíž je možné získané výsledky porovnat.

V první fázi praktické realizace bylo navíc nutné pracovat s umělými daty, na kterých byly otestovány základní vlastnosti navrženého neuronu a neuronové sítě.

Oblasti, na kterých bude navržená síť testována, lze tedy rozdělit do 3 skupin:

- umělá data
- obrazová data
- EEG a případně další biologické signály

Konečným cílem by měla být praktická aplikace pro monitorování úrovně pozornosti nebo jiných parametrů řidiče. Pro monitorování funkce řidiče lze použít dva typy parametrů, a to

- přímé sledování řidiče, např. EEG, pohyby očí
- nepřímé sledování řidiče, např. sledování akcelerace, udržování v jízdním pruhu apod.

Mým cílem je síť, která bude schopna pracovat s oběma druhy vstupů.

Z hlediska učení sítě je zajímavá představa použít přímých indikátorů jakožto vstupů a nepřímých jakožto výstupu neuronové sítě.

#### 3.1.6 Problémy systémů pro automatické vyhodnocení funkce lidského operátora

V současné době již existuje mnoho automatických systémů pro vyhodnocování funkce lidského operátora, které naráží zejména na následující problémy:

- sledovaný systém je pokaždé jiný, rozdíly mezi jednotlivými jedinci jsou značné (např. rozdílná odolnost vůči únavě či náchylnost ke stresu)
- pro ohodnocení operátora v čase  $t$  je možné použít pouze hodnoty z času  $[t_n, t_{n-1}, \dots, t_{n-x}]$ .
- v praxi je potřeba problémy nejen detekovat, ale zejména je predikovat, tedy je potřeba ohodnocení alespoň v čase  $t_n + 1$ .

Řešení těchto problémů spočívá v

- nalezení takových charakteristik, které jsou společné pro všechny relevantní pokusné osoby (probandy). Toto je obtížně aplikovatelné při analýze EEG, neboť křivka elektroencefalografu je výrazně individuální
- individuální nastavení systému pro jedinou vybranou osobu. Ačkoliv toto řešení působí na první pohled neprakticky, je v případě řízení dopravních prostředků použitelné, neboť obvykle daný dopravní prostředek používá omezený počet uživatelů.
- pro samotnou detekci stavů řidiče nelze použít budoucí hodnoty (a naopak je obvykle potřeba budoucí hodnoty predikovat), ale je možné využít budoucí hodnoty při fázi trénování nebo při fázi adaptace parametrů systému pro danou osobu.

Je pravděpodobné, že pro tak náročný úkol jako je modelování a predikce lidského operátora bude třeba kombinovat všechna výše uvedená řešení.



---

# Princip umělé homeostatické neuronové sítě

## 4.1 Informační homeostáze

Hledání kompromisu mezi složitostí sítě a přesností popisu neuronu mne vedlo k návrhu neuronové sítě založené na principu informační homeostáze. Tento princip vychází z předpokladu, že neurony mezi sebou soupeří o zdroje, a kritérium pro přežití neuronu je realizace takové funkce, která je maximálně užitečná též pro ostatní neurony. Prostředkem pro nalezení této funkce je nastavení vah vybraného neuronu, přičemž pro nalezení této pozice lze použít znalosti o procesu homeostáze známého z biologie (např. udržování koncentrace solí nebo teploty). Na rozdíl obvyklého významu pojmu homeostáze se jedná o proces spojený s přenosem informace [A1, A2, A3].

V dalším textu bude použito označení *výstupní neuron* pro libovolný neuron sítě, který přijímá signál od referenčního neuronu, a *vstupní neuron* pro neuron, který svůj výstup předává *referenčnímu neuronu*. Analogicky k tomu existují vstupní  $W_i$  resp. výstupní váhy  $W_o$ , což jsou reálná čísla, kterými se násobí signál ze vstupního resp. do výstupního neuronu.

Základní jednotkou navržené neuronové sítě je tzv. homeostatický neuron. Učení je založeno na předpokladu, že neurony spolu soupeří o zdroje, přičemž měřítkem úspěšnosti je význam neuronu pro ostatní neurony. Každý neuron se snaží dosáhnout stavu, v němž bude jeho výstup požadován co největším počtem ostatních neuronů.

Jedná se o učení bez učitele, protože modelu není nadřazen žádný jiný systém, který by stanovoval úspěšnost neuronu. Neuron si sám určuje vlastní úspěšnost na základě vah dendritů patřících ostatním neuronům, které od něj odebírají výstup.

Tato myšlenka má své biologické opodstatnění, neboť reálný biologický neuron v závislosti na svém vnitřním stavu (excitaci) vysílá energii, jejíž část je akceptována ostatními neurony. Pokud tato energie není akceptována ostatními neurony, je to informace pro neuron, že jeho funkce není adekvátní. V tomto přiblížení je možné si představit homeostatický neuron jako inteligentní jednotku, která hledá takové nastavení, při kterém budou váhy neuronů na ní napojených maximální.

## 4.2 Přenosová funkce neuronu

Z hlediska dopředné fáze přenosu signálu od nejnižší po nejvyšší vrstvu je referenční neuron identický s klasickým perceptronem, tj. jedná se o jednotku provádějící nelineární zobrazení váženého součtu vstupů:

$$y = S \sum_{i=1}^{i=n} (x_i w_i + \phi) \quad (4.1)$$

kde  $\phi$  je práh neuronu,  $w_i$  je  $i$ -tá váha neuronu  $x_i$  je vstup  $i$ -tého neuronu a  $S$  je funkce sigmoidálního typu:

$$S(\chi) = \frac{1}{1 + e^{-\alpha\chi}} \quad (4.2)$$

Odlišnost spočívá v učící fázi, která vychází z principu hledání a udržování rovnovážné pozice, nikoliv ze zpětného šíření chyby.

## 4.3 Užitečnost neuronu

Pro učení neuronu je nutné definovat kritérium užitečnosti neuronu pro neuronovou síť (v dalším textu označeno jako  $q$ ).

Existuje několik variant výpočtu užitečnosti, které jsou popsány rovnicemi (4.3)-(4.9).

První krajní možností je optimalizace vah takovým způsobem, aby neuron byl užitečný pro všechny neurony ve výstupní vrstvě, tj. aby maximalizoval součet výstupních vah (4.3).

Opačným extrémem je optimalizace právě pro jeden výstupní neuron, tj. hledání maxima výstupních vah (4.4).

Mezi těmito dvěma krajními případy existuje mnoho kompromisních řešení, např. optimalizace podle určitého počtu nejvýznamnějších vah (4.5), ocenění nejdůležitějšího výstupu  $k$ -krát více než ostatních (4.6), kritérium hledající maximum medianu vazeb (4.7), nebo jiné části  $\alpha$  kvantilu množiny vazeb (4.8).

Nejblíže biologickému vzoru je varianta učení dle součtu výstupních vah (4.3), nejvíce se od něj vzdaluje maximum výstupních vah (4.9). Tyto varianty mají reálný vzor v biologickém neuronu, protože výstupní neurony jsou napojeny na axon referenčního neuronu v různých místech, což vede k různým mírám útlumu.

$$q = \sum_{i=1}^n w_i^o \quad (4.3)$$

$$q = \max(W^o); i = \{1, 2, \dots, n\} \quad (4.4)$$

$$q = \max(W^o) + \max\{W^o - \max(W^o)\} \quad (4.5)$$

$$q = k * \max(W^o) + \sum_{i=1}^n w_i^o \quad (4.6)$$

$$q = \text{med}(W^o) \quad (4.7)$$

$$q = Q_\alpha\{W^o\} \quad (4.8)$$

$$q = \sum_{i=1}^n w_i^o; w_i^o \in W : w_i^o = Q_\alpha(W) \quad (4.9)$$

kde  $W^o = \{w_1^o, w_2^o, \dots, w_n^o\}$  je množina vah výstupních neuronů spojených s referenčním neuronem

$q$  je užitečnost neuronu a

$\alpha$  je reálný parametr rozdělující množinu vah na dvě části, užitečné a neúžitečné.

Každé z těchto kritérií zřejmě povede k odlišným výsledkům.

Bude-li užitečnost daná součtem všech vah, bude učení pomalejší. V případě velkého množství výstupních vah (u biologického neuronu desetitisíce) by pravděpodobně učení vůbec neproběhlo. Naproti tomu v případě učení podle maxima výstupních vah je pravděpodobné, že na začátku bude alespoň jedna výstupní váha mít hodnotu blízkou jedné, a tím bude i užitečnost ihned po zahájení učení maximální. Jakožto nejvýhodnější se jeví nejobecnější  $\alpha$  kvantil (4.8), kde se bude počet výstupních neuronů postupně snižovat z  $\alpha_{poc}$  na  $\alpha_{kon}$ . Parametr  $\alpha$  by v tom případě mohl být řízen funkcí závislou na počtu kroků učícího cyklu, případně by mohl být závislý na výsledku neuronové sítě.

## 4.4 Učení navržené neuronové sítě

Otázku učení celé homeostatické neuronové sítě lze formulovat jako hledání takového nastavení vstupních vah  $W_i$ , aby užitečnost  $q = f(W^o)$  byla maximální.

Z tohoto principu vyplývá, že ve fázi učení je opačný směr toku informací než při dopředné fázi – výstupní váhy jsou vstupem do procesu učení a vstupní váhy jsou výstupem.

Pro určení změny vstupních vah existuje několik postupů. První postup spočívá v náhodných změnách vstupních vah a ve sledování efektu, který tato změna má na výstupní váhy. Pokud změna vede ke zvýšení užitečnosti, je potvrzena, pokud vedla ke snížení užitečnosti, je provedena změna opačná.

Tento postup by mohl dobře fungovat v případě malého počtu vah. Pokud však je vah větší množství (řádově tisíce v případě biologického neuronu), bude učení tímto způsobem velmi pomalé.

Druhou možností je vyhodnotit vliv jednotlivých vah a dále se soustředit pouze na ty váhy, které mají největší význam pro výstup neuronu. Vybrat váhy lze např. podle hodnoty a variability vstupního signálu – signál, který je blízký nule nebo je víceméně konstantní, nemůže výrazně zlepšit ohodnocení neuronu, a proto jej není třeba optimalizovat.

Základní metodu učení neuronu lze realizovat v těchto krocích:

1. počáteční nastavení vah na náhodná čísla
2. výpočet dopředné fáze
3. výpočet užitečnosti  $q(w)$  podle některého ze vztahů 4.4) až (4.9). Ohodnocení neuronu je vyjádřeno tím, že neurony ve vyšší vrstvě nastaví své váhy podle spokojenosti s výstupem referenčního neuronu
4. úprava váhy s indexem  $n$ , přičemž nový vektor vah je označen  $w_n$
5. výpočet dopředné fáze se stejnými vstupy
6. ohodnocení neuronu pomocí vyšší vrstvy stejným způsobem jako v bodu 3
7. určení užitečnosti stejně jako v bodu 3
8. uložení užitečnosti  $q(w_n)$
9. opakování kroků 4) až 8) pro všechny váhy
10. výběr váhy, jejíž změna vedla k největší změně užitečnosti  $w_{max}$
11. porovnání užitečností před  $q(w)$  a po změně  $q(w_{max})$ . Pokud změna vedla ke zvýšení užitečnosti, je ponechána, v opačném případě je provedena změna opačná. Pokud je užitečnost přibližně stejná jako před změnou, vrátí se síť do původního stavu.

Na rozdíl od algoritmu zpětného šíření chyby disponuje homeostatický neuron pamětí, pomocí níž může porovnávat své výsledky s předchozími stavy. Učení (které zde probíhá v bodu 7) bude výrazně efektivnější, pokud tento neuron nebude pouze potvrzovat nebo odmítat změny vah, ale pokud výsledná změna váhy bude proporcionální k derivaci užitečnosti (velké zlepšení užitečnosti=velká změna váhy).

## 4.5 Citlivostní analýza neuronové sítě

Pro zvýšení užitečnosti neuronu je klíčovou otázkou nalézt váhu s nejvyšší citlivostí na změnu užitečnosti neuronu pro vyšší vrstvu.

V rámci této práce je použita relativní (tzv. diferenciální) definice citlivosti dle Bodeho [?]:

$$s_{x_i}^F = \frac{\ln F(X, P)}{x_i} \quad (4.10)$$

kde

F je systémová funkce,

X je vektor parametrů a

P je vektor nezávisle proměnných.

Tuto funkci citlivosti lze aplikovat na homeostatickou neuronovou síť tak, že systémová funkce F je rovna spokojenosti s výstupem neuronu, X je vektor vah, případně strmostí nebo jiných parametrů, a P je vektor nezávislých parametrů (pozn. tento vektor je v případě z oddílu 2.2 prázdný, protože není uvažováno s nezávislými veličinami, např. s časem. Pro vytvoření realistického modelu řidiče je ale nutné s časem a dalšími nezávislými veličinami uvažovat.)

Využitím vztahu (4.10) lze určit váhu, jejíž změna má největší vliv na užitečnost neuronu, a která bude v dalším kroku upravena.

Pomocí citlivostní analýzy lze určit, zda je daná soustava

- kanonická (má právě nezbytný počet prvků)
- subkanonická (má menší než nezbytný počet prvků)
- redundantní (má více než nezbytný počet prvků).

Biologické i umělé neuronové sítě jsou zpravidla velmi redundantní, zůstává ale otevřená otázka, jak moc velká je míra jejich redundance. Intuitivně lze očekávat, že redundance modelu by měla být podobná jako redundance reálného systému.

Proto je vhodné stanovit alespoň přibližně nezbytný počet prvků.

### 4.5.1 Současný výběr více vah na základě analýzy citlivosti

Pomocí citlivostní analýzy lze nalézt váhu s největší citlivostí. Je ale možné nehledat jen jednu váhu s maximální citlivostí, ale skupinu vah, jejichž citlivosti překračují určitou mez. Následně budou modifikovány všechny tyto váhy. Jedná se o podobný princip jako u procesu back propagation v batch modu, ale s tím rozdílem, že se učení týká jen několika vah, ne všech. Příkladem je výběr jistého procenta vah (zde např. 95 %) vah s největší citlivostí:

$$\bar{s} = Q_{0.95} s_{x_i}^F, i \in \langle 0, n \rangle \quad (4.11)$$

## 4.6 Varianty učícího algoritmu

Výše popsaný postup učení neuronu úpravou váhy s nejvyšší citlivostí povede k největšímu zlepšení v následujícím kroku, není ale zaručeno, že se jedná z dlouhodobého hlediska o ideální změnu. Společný efekt několika změn vah, z nichž žádná není v daném kroku nejcitlivější, může být ve výsledku výraznější než neustálá úprava nejcitlivější váhy. Druhou nevýhodou algoritmu úpravy nejcitlivější váhy je skutečnost, že pro určení váhy s největším dopadem na výsledek je nutné spočítat citlivosti všech vah, což zpomaluje výpočet. Z obou důvodů byly navrženy dvě varianty k výše popsanému algoritmu, a to úprava náhodně vybraných vah a úprava postupně vybíraných vah.

### 4.6.1 Úprava náhodně vybrané váhy

Tento algoritmus učí síť v následujících krocích:

1. počáteční nastavení vah na náhodná čísla
2. výpočet dopředné fáze
3. výpočet užitečnosti  $q(w)$  - ohodnocení funkce vybraného neuronu pomocí neuronů ve vyšší vrstvě, které nastaví své váhy podle spokojenosti s výstupem vybraného neuronu

4. náhodný výběr indexu váhy  $n$
5. změna váhy s indexem  $n$  o  $\Delta w$ , nový vektor vah je označen  $w_n$
6. výpočet dopředné fáze
7. výpočet užitečnosti  $q(w_n)$
8. porovnání užitečností před  $q(w)$  a po změně  $q(w')$ . Pokud změna vedla ke zvýšení užitečnosti, je ponechána, v opačném případě je provedena změna opačná. Pokud je užitečnost přibližně stejná jako před změnou, vrátí se síť do původního stavu.
9. opakování kroků 1) – 5) až do dosažení požadované užitečnosti.

Pro tento algoritmus je důležitý parametr  $m$ , který určuje mez, při jejímž překročení je změna užitečnosti dostatečně významná na to, aby se potvrdila změna parametrů neuronu.

## 4.7 Postupná úprava vah

V případě tohoto algoritmu se váhy nevybírají náhodně, ale postupně. Algoritmus je identický s algoritmem 4.6.1 s tím rozdílem, že v prvním bodu se vybírá váha s indexem  $n$ , který se v každém kroku zvyšuje.

1. počáteční nastavení vah na malá náhodná čísla
2. výpočet dopředné fáze,
3. výpočet užitečnosti  $q(w)$ -ohodnocení neuronu pomocí neuronů ve vyšší vrstvě, které nastaví své váhy podle spokojenosti s výstupem referenčního neuronu
4. změna váhy s indexem  $n$  o  $\Delta w$ , nový vektor vah je označen  $w_n$
5. výpočet dopředné fáze
6. výpočet užitečnosti  $q(w_n)$
7. porovnání užitečností před a po změně. Pokud změna vedla ke zvýšení užitečnosti, je ponechána, v opačném případě je provedena změna opačná. Pokud je užitečnost přibližně stejná jako před změnou, vrátí se síť do původního stavu.

8. inkrementace indexu váhy  $n$  o 1. Pokud je  $n$  rovno počtu vah neuronu, pak je  $n=1$
9. opakovat kroky 1) –5) až do dosažení požadované užitečnosti.

## 4.8 Omezení vah

Biologický neuron přijímá signál pomocí synapsí a dendritů, které mění intenzitu signálu. Tato změna je v umělých neuronových sítích simulována násobením reálným číslem, což výrazně rozšiřuje výpočetní možnosti. Na druhou stranu se tím umělé neuronové sítě vzdalují od svého biologického vzoru, protože reálný neuron nemůže libovolně zvyšovat vstupující energii. Existuje několik možností, jak adaptovat proces změny signálu v synapsi tak, aby se více podobal skutečnosti. V rámci této práce byly prozkoumány tři z těchto možností, a to:

- váhy jsou omezeny na interval  $\langle -0; 1 \rangle$ . Tato varianta je nejbližší biologickému vzoru, protože dendrit jako takový nemůže dodávat energii zesilující přijatý signál, ani měnit jeho polaritu
- váhy jsou omezeny na interval  $\langle -1; 1 \rangle$ . Jedná se o kompromisní řešení, které je v určitém ohledu také věrné biologickému vzoru, neboť v biologických neuronových sítích se vyskytují inhibiční váhy, které jsou v modelu reprezentovány zápornou hodnotou váhy
- váhy nejsou omezeny. Tato možnost je nejvýhodnější z hlediska výpočtu nejrychlejší pokles střední kvadratické chyby - MSE, na druhou stranu není biologicky opodstatněná

## 4.9 Přehled variant učícího algoritmu homeostatického neuronu

V předchozích kapitolách byly definovány tyto varianty učení umělé neuronové sítě:

- Z hlediska výběru optimalizované váhy
  - učení nejvýznamnější váhy
  - učení náhodně vybrané váhy



postupné učení vah

- Z hlediska přístupu k neuronům ve větší vrstvě
  - maximalizace užitečnosti pro všechny výstupní neurony
  - maximalizace užitečnosti pro právě jeden výstupní neuron
  - maximalizace užitečnosti pro daný počet neuronů
- Z hlediska omezení vah
  - váhy neomezeny
  - váhy omezeny na interval  $\langle 0; 1 \rangle$
  - váhy omezeny na  $\langle -1; +1 \rangle$

## 4.10 Ohodnocení kvality neuronové sítě

Pro posouzení kvality neuronové sítě bude homeostatická neuronová síť porovnána s neuronovou sítí s učením typu back propagation. Navržená síť bude ohodnocena z hlediska:

- rychlosti učení
- schopnosti generalizace
- kontinuita učení

přičemž

- rychlost učení bude posouzena jako počet iterací nutných pro dosažení alespoň 95 % úspěšnosti rozpoznávání, případně střední kvadratickou odchylku menší než 0,01.
- schopnost generalizace bude ohodnocena jako poměr mezi účinností rozpoznávání trénovací množiny ku testovací množině
- kontinuita učení bude hodnocena jako počet běhů učícího algoritmu, při kterých došlo ke zhoršení střední kvadratické chyby.

---

## Realizace programu založeného na principu hledání rovnovážné polohy

V této části je popsán program učení neuronu na principech popsaných v kapitole 4. Jsou zde definovány hlavní funkce a proměnné. Detailní popis každé funkce je v komentáři funkce. Kód je napsán v jazyce Matlab.

### 5.1 Topologie homeostatické neuronové sítě pro analýzu a predikci lidského operátora

Všechny pokusy o modelování mozkových funkcí narážejí na problém velikosti použité sítě, který vede k nezbytnosti kompromisu mezi přesností popisu neuronů a velikostí neuronové sítě. Pro realistické modely je nutné vytvořit model na detailní úrovni, kde vysoká složitost použitých neuronů brání vytvoření modelu s dostatečným počtem výkonných prvků.

Příkladem takového přístupu je projekt HBP (Blue Brain), který požaduje pro realistický model jediného neuronu použití jednoho procesoru o pracovní frekvenci 1 GHz.

Na druhé straně stojí modely s velkým počtem neuronů, které ale kvůli své jednoduchosti nemohou věrně modelovat biologickou neuronovou síť. Příkladem je míněný projekt SyNAPSE, který modeluje neuronové sítě se srovnatelným počtem prvků jako má lidský kortex, ale neurony jsou silně zjednodušeny.

Při definici struktury neuronové sítě je nutné přihlídnout k mnoha aspektům, z nichž na prvním místě stojí kritérium určení počtu skrytých vrstev. Biologická neuronová síť je organizovaná do kolumn o 6 vrstvách, proto se s ohledem na požadavek podobnosti s

## 5.1. Topologie homeostatické neuronové sítě pro analýzu a predikci lidského operátora

biologickou neuronovou sítí jeví tento počet jako výhodný.

Na druhou stranu se v síti se 6 vrstvami (tzn. se 4 skrytými vrstvami) vyskytuje velmi obtížně řešitelný problém zpoždění, neboť informace o užitečnosti se bude zpožďovat vůči dopředné informaci až o 12 kroků.

Proto je nutné výsledky porovnat se sítěmi s menším počtem skrytých vrstev. Na základě zkušenosti byly navrženy sítě s počtem skrytých vrstev 1, 2 a 4.

Sítě s vyšším počtem skrytých vrstev nebyly testovány, neboť nemají biologické vzor v lidském mozku, na druhou stranu je ale možné, že by v určitých typech úloh mohly dosahovat lepších výsledků.

Druhým důležitým parametrem je počet neuronů v jednotlivých vrstvách. Zkušenosti se standardními neuronovými sítěmi ukazují, že nejlepších výsledků dosahují sítě, ve kterých počet neuronů ve vrstvách roste tak, že v první skryté vrstvě je přibližně dvojnásobný počet neuronů než ve vstupní vrstvě, atd. až přibližně do vrstvy ve středu neuronové sítě, odkud se počet neuronů ve vrstvě opět snižuje.

Nejedná se ale o univerzální pravidlo a počet neuronů se v praxi určuje spíše testováním a na základě empirických pouček. Topologie sítí testovaných v rámci této práce jsou uvedeny v tab. 5.1.

Počet skrytých vrstev	Počet neuronů ve vrstvách I	Počet neuronů ve vrstvách II	Počet neuronů ve vrstvách III
1	1-2-1	4-4-4	100-100-100
2	1-2-2-1	4-10-4-4	100-200-150-10
4	1-2-4-4-2-1	100-200-400-400-200-100	500-3000-2000-3000-1000-500

Tabulka 5.1: Navržená topologie neuronových sítí

Nejblíže skutečné kolumně je varianta III se 4 skrytými vrstvami.

Dalším krokem je definice parametrů učicího algoritmu, konkrétně např. učicí krok, délka zapomínacího okénka (tj. počet minulých hodnot, které si neuron pamatuje), paměť minulých hodnot apod.

### 5.1.1 Program učení jediného neuronu

Program umělé neuronové sítě s homeostatickým způsobem učení byl realizován v prostředí Matlab. Požadavkem je, aby všechny varianty byly spustitelné jediným programem, ve kterém je typ učení zadán volbou.

Celkem jsem zkoušel 6 variant učení. Z hlediska kritéria posouzení užitečnosti se jedná o:

- učení pro všechny výstupní neurony (maximalizace součtu výstupních vah)
- učení pro právě jeden výstupní neuron (maximalizace maxima výstupních vah)

Z hlediska výběru upravované váhy

- váha s největším vlivem na výsledek
- náhodně vybraná váha
- postupný výběr vah

Program se spouští funkcí HAN (Homeostatic Artificial Neuron) s parametry ( $X, D, \text{learning\_type}, \text{criterion}$ ), kde

- $X$  je matice vstupů, ve které každému vstupnímu vektoru odpovídá jeden řádek.
- $D$  je matice požadovaných výstupů (v případě, kdy je neuron součástí neuronové sítě, je tento vektor nahrazen vyšší vrstvou neuronové sítě): Počet řádků matice  $D$  musí být stejný jako u matice vstupů  $X$ , neboť každý řádek odpovídá kombinaci vstupu a výstupu. Počet sloupců je roven počtu simulovaných výstupních neuronů.
- **learning\_type** je typ učicí funkce. Tento parametr může nabývat dvou hodnot, a to buď učení pro všechny výstupní neurony, nebo učení pro právě jeden výstupní neuron.
- **criterion** optimalizační kritérium. Tento parametr může nabývat 3 hodnot, a to optimalizace váhy s největší citlivostí (tj. váhy s největším zvýšením užitečnosti v příštím kroce), nebo optimalizace náhodně vybrané vhy, nebo optimalizace postupně vybrané váhy.

Celý vytvořený program je uveden na přiloženém CD disku a je strukturován do následujících šesti podprogramů volaných z hlavního programu:

- **Forward\_Phase** – výpočet dopředné fáze neuronu
- **Utility** - výpočet užitečnosti neuronu  $q$ , na základě této hodnoty se činí rozhodnutí, jak změnit váhy sítě
- **Weight\_Update**-změna vah
- **Weight\_ext**-rozšíření matice vah. Z důvodu učení zvyšováním užitečnosti pro vyšší vrstvu se namísto vektoru vah  $w = [1, n_x]$  pracuje s maticí  $w_{ext} = [2 * n + 1, n_x]$ , kde v prvním řádku je původní vektor  $w$ , v dalších  $n$  řádcích je vždy jeden (subdiagonální) prvek zvýšen o  $\delta w$  a v dalších  $n$  řádcích je subdiagonální prvek zmenšen o  $\delta w$ .
- **Step\_Size**-výpočet velikosti kroku, o který se změní váhy mezi dvěma iteracemi
- **Desired\_Extended**-úprava matice požadovaných výstupů  $D$ . Tato matice je rozšířena z dimenze  $[m, n]$  na  $[2n_x + 1, mn]$ , kde  $n_x$  je počet vstupních neuronů v případě učení nejvýznamnější váhy, nebo na dimenzi  $[3, mn]$  v případě učení náhodně vybrané váhy nebo v případě postupného učení vah.

### 5.1.2 Popis stavů neuronu

Vstupem učicího algoritmu pro jednotlivý neuron jsou:

- matice vstupů  $X(m, n_1)$ , kde  $m$  je počet vzorů, které se neuron učí a  $n_1$  je počet dendritů neuronu (počet vstupů neuronu)
- matice požadovaných výstupů  $D(m, n_2)$ , kde  $m$  je počet vzorů a  $n_2$  je počet neuronů napojených na výstup referenčního neuronu (počet výstupních neuronů)
- vektorem *evaluate* definujícím spokojenost všech neuronů v téže vrstvě. Na základě tohoto vektoru vybraný neuron nastavuje učicí krok. Velikost učicího kroku je větší, pokud je dne tohoto vektoru užitečnost vybraného neuronu nízká ve srovnání s ostatními neurony.

Vybraný neuron je charakterizován vektorem vah  $w$  a prahem  $t$ .

- sumaci vstupů pro každý vzor s indexem  $n$ :  $s = w * x(n, :)^T$
- přenosovou funkci:  $y = \frac{1}{1 + e^{-\alpha s}} + t$

Výstupní neurony nastavují své váhy podle spokojenosti s výstupem neuronu. Z důvodu stručnosti popisu je namísto výstupních neuronů použita matice požadovaných výstupů  $D$ . Učení lze popsat takto: v prvním kroku je vektor vah  $w$  rozšířen na matici

$$w_{ext} = \begin{pmatrix} w_1 & w_2 & \dots & w_{n-1} & w_n \\ w_1 + \Delta & w_2 & \dots & w_{n-1} & w_n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ w_1 & w_2 & \dots & w_{n-1} - \Delta & w_n \\ w_1 & w_2 & \dots & w_{n-1} & w_n - \Delta \end{pmatrix} \quad (5.1)$$

V prvním řádku této matice je původní vektor vah  $w$ , pod tímto vektorem je čtvercová submatice, ve které je k hlavní diagonále přičten krok  $\Delta$ . Pod ní je druhá čtvercová submatice, od jejíž diagonály je odečten krok  $\Delta$ . Váhy jsou omezeny na interval  $\langle -1, 1 \rangle$ . Dimenze matice je  $\langle 2 * n + 1, n \rangle$ .

Pro každý vektor vah (tj. pro každý řádek v matice  $w_{ext}$  a pro každý vzor (tj. pro každý řádek v matice  $X$ ) je spočítán výstup neuronu. Výsledkem je matice výstupů

$$y = \begin{pmatrix} y_{1,1} & \dots & y_{1,m} \\ \vdots & \ddots & \vdots \\ y_{2*n+1,1} & \dots & y_{2*n+1,m} \end{pmatrix} \quad (5.2)$$

kde  $y_{i,j}$  je odpověď neuronu s nastavením vah dle  $i$ -tého řádku na  $j$ -tý vzor. Skutečné výstupy neuronu jsou porovnány s požadovanými výstupy. Vzhledem ke skutečnosti, že ve výstupní vrstvě se nachází více neuronů s různými požadovanými funkcemi, je každý skutečný výstup porovnán se všemi požadovanými výstupy. Toto porovnání je realizováno pomocí tří-dimenzionální matice  $Q$ , kde prvek  $q(i,j,k)$  má význam spokojenosti výstupního neuronu číslo  $k$  na  $i$ -tou variantu referenčního neuronu vybuzeného vstupem číslo  $j$ .

## 5.2 Algoritmus učení neuronové sítě

Proces učení samostatného neuronu lze zobecnit na celou neuronovou síť. Referenční neuron nastavuje své parametry na základě vstupů od neuronů v nižších vrstvách, přičemž tyto parametry jsou učícím kritériem pro vstupní neurony. Z praktického hlediska je hlavní překážkou realizace rozhraní mezi neurony. Toto rozhraní je provedeno jakožto vektor  $l$  o  $n$

prvcích, ve kterém  $n$ -tý prvek udává spokojenost všech výstupních neuronů s  $n$ -tým neuronem. Tento vektor je získán z matice užitečnosti  $Q$  tak, že  $l(n)=f(g(Q))$ , kde  $g$  je funkce posuzující vhodnost daného nastavení vah podle určeného kritéria (kvadratické kritérium, absolutní vzdálenost od požadované funkce apod.),  $f$  je funkce posuzující vhodnost nastavení neuronu pro vyšší vrstvu. Tato funkce závisí na typu učení, tj zda se jedná o učení pro všechny výstupní neurony, pro pouze jeden výstupní neuron atd.

Pro dvouvrstvou síť platí následující učící algoritmus:

1. referenční neuron provede dopřednou fázi a pošle své výsledky do vyšší vrstvy
2. neurony ve vyšší vrstvě na základě spokojenosti s výstupem referenčního neuronu nastaví své váhy  $w$ .
3. referenční neuron spočítají svou užitečnost pro jednotlivé výstupní neurony a výsledek uloží do matice  $Q$
4. neuron upraví své parametry dle daného postupu (úprava nejcitlivější váhy, úprava náhodně nebo postupně vybrané váhy) a provede dopřednou fázi
5. neurony ve vyšší přepočítají své vstupní váhy  $w'$
6. referenční neuron spočítá novou užitečnost  $Q'$  podle stejného postupu jako v bodě 3
7. referenční neuron vybere z vah  $w$  a  $w'$  takové nastavení, které má vyšší užitečnost
8. neurony ve vyšší vrstvě nastaví váhy identickým způsobem
9. celá síť opakuje body 1 až 8 až do dosažení požadované MSE

## 5.3 Velikost učícího kroku

Neurony ve stejné vrstvě se snaží maximalizovat svůj význam v porovnání s ostatními neurony téže vrstvy. Význam neuronu je relativní hodnota, která popisuje vztah neuronu k vyšší vrstvě, analogicky k principu soupeření o zdroje mezi neurony téže vrstvy. Pokud je význam neuronu ve srovnání s ostatními neurony vysoký, není nutné příliš měnit jeho parametry.

Učící krok  $\Delta w$  je hodnota, o kterou se změní váha během jedné iterace. Velikost učícího kroku je závislá na relativním významu neuronu.

Prakticky je stanovení významu realizováno pomocí vektoru evaluate, ve kterém prvek na  $i$ -té pozici má význam užitečnost  $q$  neuronu číslo  $i$ . Referenční neuron provádí seřazení vektoru evaluate od nejnižší po nejvyšší hodnotu. V závislosti na tom, kde se nachází jeho vlastní index, normuje velikost učícího kroku konstantou

- 0,1, pokud spadá mezi 10 % nejlépe ohodnocených neuronů
- 0,3, pokud je horší než 10 % nejlepších neuronů a lepší než 10 % nejhorších neuronů
- 1 pokud spadá mezi 10 % nejhorších neuronů.

Tyto hodnoty vycházejí z praktického testování a je možné, že pro jiná data či jiné typy sítí by byly vhodnější jiné konstanty.

Existuje mnoho jiných způsobů jak spočítat velikost učícího kroku, podstatný je princip rostoucí velikosti kroku se zhoršující se funkcí neuronu. Neuron, který jsou dobře hodnoceny vyšší vrstvou, mění své váhy jen o malý krok, zatímco málo úspěšné neuron mění své parametry výrazně.

Možné modifikace výpočtu učícího kroku jsou:

- lineární závislost kroku na významu
- spojitá nelineární funkce závislostí kroku na významu, např. sigmoida
- skoková prahová funkce s jiným počtem stupňů a jiným odstupňováním prahů, než je uvedeno v předchozím odstavci.

Pro různé úlohy se mohou osvědčit jiné způsoby určení učícího kroku. Ideální metoda závisí na použitých datech, velikosti sítě, výpočtu užitečnosti a funkci pro výpočet užitečnosti. Neexistuje univerzální návod pro výběr ideální metody, je vždy nutné využít zkušenost a odhad.

## 5.4 Problémy učení na principu informační homeostáze

Popsaný princip učení homeostatického neuronu přináší dva hlavní problémy dané rychlostí přenosu informace a uzavřeností učícího cyklu.



### 5.4.1 Zpoždění neuronové sítě

Předpokládejme vrstevnatou neuronovou síť bez laterálních vazeb. V každém kroku je na vstup přiveden jiný vzor. První vrstva předá svůj výstup druhé vrstvě, druhá vrstva třetí, atd. Poté dojde k aktualizaci vah, což obnáší vyčíslení užitečnosti neuronu dle výstupních vah a následnou úpravu vstupních vah. Následně je na vstup sítě přiveden jiný vzor. Výsledkem je, že informace o užitečnosti vzoru z času  $t$  se setkává v neuronu ve vzoru z času  $t+1$ . S rostoucím počtem vrstev se zvyšuje odstup mezi vzorem a informací o odezvě na daný vzor.

Existují dvě řešení tohoto problému.

První možností je nastavit dynamiku procesu na dostatečně pomalou úroveň, aby ke změně vstupu docházelo až poté, co se dopředná informace dostane do nejvyšší vrstvy, a informace o užitečnosti zpět do nejnižší.

Druhou možností je vybavit neurony pamětí, která jim umožní zapamatovat si vzor, který byl na jejich vstupu v okamžiku příslušejícím současné užitečnosti.

Nevýhodou prvního přístupu je pomalost učení a možná nestabilita, v druhém případě se trénování vzdaluje od biologického vzoru.

### 5.4.2 Učení nejvyšší vrstvy neuronové sítě

Popsaný postup učení může fungovat pro všechny neurony, na které je napojen další neuron, tj. na všechny neurony s výjimkou výstupní vrstvy, neboť tyto neurony nemají výstupní váhy a tudíž pro nelze použít princip optimalizace vstupních vah dle vah výstupních. Z tohoto vyplývá nemožnost použít uvedený učící princip pro neurony v nejvyšší vrstvě.

V reálném světě tento problém nevzniká, neboť výstupní vrstva je napojena, byť nepřímou na aktuátory, jejichž zásahy se přes okolí dostávají zpět na smyslové vstupy neuronové sítě. V případě modelu ale nedisponujeme touto uzavřenou smyčkou.

Řešení tohoto problému vychází ze skutečnosti, že v bližším pohledu lze na mozek pohlížet jakožto na soustavu relativně samostatných neuronových sítí, uvnitř kterých probíhá velmi intenzivní komunikace, ale přenos dat mezi těmito sítěmi má podstatně nižší frekvenci.

V případě homeostatické neuronové sítě popsané v této práci se lze spokojit s tím, že předmětem modelování je vlastní rozhodování a řízení, vyhodnocení spokojeností s funkcí řízení se odehrává v jiné síti, která se nemusí řídit stejným principem.

Problém učení nejvyšší vrstvy v modelech studovaných v této práci jsem tady vyřešil tak, že pro neuronu v nejvyšší vrstvě jsem aplikoval jinou učící metodu (back propagation) než pro ostatní neurony.

---

# Ověření principu učení homeostatického neuronu

## 6.1 Použitá data

Testování homeostatické neuronové sítě je obecně rozděleno do dvou fází:

- testování samotného vybraného neuronu
- testování vrstevnaté homeostatické neuronové sítě

Účelem první fáze je otestovat realizovatelnost navrženého učicího principu.

Cílem druhé fáze je ověřit možnosti vybrané homeostatické neuronové sítě pro modelování reálných situací, se kterými se setkává operátor dopravního prostředku.

Z hlediska dat použitých pro testování je experiment rozdělen do dvou částí. Vybraný neuron i navržená homeostatické neuronová síť jsou testovány na těchto typech dat:

- uměle vytvořená data,
- experimentálně získaná data.

V první části se jednalo o ověření realizovatelnosti samotného učicího principu pro učení jediného neuronu. V tomto případě nemá smysl používat experimentálně získaná data, neboť navržený homeostatický neuron je v dopředné fázi shodný s jednovrstvým perceptronem a může tudíž nalézt řešení jen pro lineárně separovatelné množiny. Takové množiny ale v praktických úlohách, které jsou zajímavé z hlediska zaměření této práce, nemají význam, proto jsou použity uměle navržené množiny.

Pro testování vícevrstevných homeostatických neuronových sítí s architekturou popsanou v kapitole 4 je možné však použít jak uměle generovaná, tak experimentálně získaná data,

### 6.1.1 Uměle generovaná data

Pro učení jednovrstvé neuronové sítě jsou použity 4 lineárně separovatelné třidimenzinální množiny. Tyto vektory byly vygenerovány náhodně. Hodnoty prvků učících vektorů jsou z intervalu  $\langle -1, 1 \rangle$ . Pro učení vícevrstvé neuronové sítě není nutné, aby data byla lineárně separovatelná (toto vyplývá z Hecht-Nielsenova teorému, podle kterého neuronová síť s alespoň jednou skrytou vrstvou umí s libovolnou přesností aproximovat libovolnou funkci). Délka vstupního vektoru vychází z topologií navržené neuronové sítě a je rovna 3, 4, 100 nebo 500 prvků. Výstupní vektory mají délky 1, 2 a 10 prvků. Vstupní i výstupní vektor jsou omezeny intervalem  $\langle -1, 1 \rangle$  a jsou vygenerovány náhodně.

### 6.1.2 Reálná data

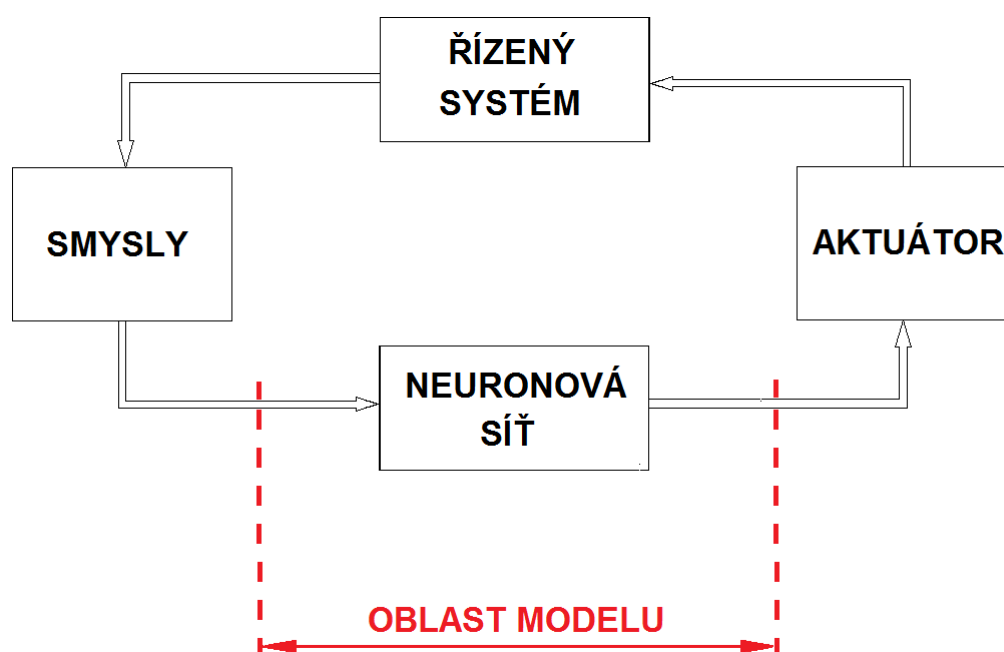
Pro ověření praktické využitelnosti navržené homeostatické neuronové sítě pro modelování procesů lidského operátora je nutné otestování pomocí reálných dat. V ideálním případě by byla na vstup přivedena stejná data, s jakými pracuje operátor dopravního prostředku, a výstupem by byly očekávané řídicí povely. V praxi takto rozsáhlé modelování není možné pro nedostupnost dat a dostatečného výpočetního výkonu. Je ale možné omezit model na děje odehrávající se „blíže“ k oblasti rozhodování.

V takovém případě není nutné, aby vstup neuronové sítě byl totožný s vjemy operátora dopravního prostředku, ale může se jednat o data do určité míry předzpracovaná. Stejně tak výstupem nemusí být impulsy ke svalům, ale rozhodnutí o plánovaných akcích, přičemž toto rozhodnutí může mít i poměrně abstraktní formu.

Vzhledem k nemožnosti vytvořit detailní model ani pracovat se stejnými daty jako operátor dopravního prostředku se oblast modelu zúžila na část biologické neuronové sítě s předzpracovanými vstupy a výstupy, jak popisuje obr. 6.1.

Nejvíce informací získáváme pomocí zraku, proto byl vstup modelu omezen na zrakové informace.

Zraková informace má obrovskou (a obtížně vyjádřitelnou) dimenzi, neboť v lidském oku se nachází přibližně 130 milionů světločivných buněk, které mají různou časovou rozlišovací frekvenci. Zraková informace je výrazně komprimována, a to částečně již v oku, částečně ve zrakovém kortexu. Bylo by velmi obtížné a z hlediska modelu rozhodovacích



Obrázek 6.1: Zjednodušené schéma informačního toku v reálném světě a v modelu.

procesů málo užitečné se snažit tuto kompresi namodelovat, je však možné použít některou z existujících metod strojového učení pro kompresi obrazu.

Jakožto vhodná metoda pro kompresi obrazové informace byla vybrána metoda PCA (Principal Component Analysis). Tato transformace umožňuje snížení dimenze vstupních dat s co nejmenší ztrátou informace.

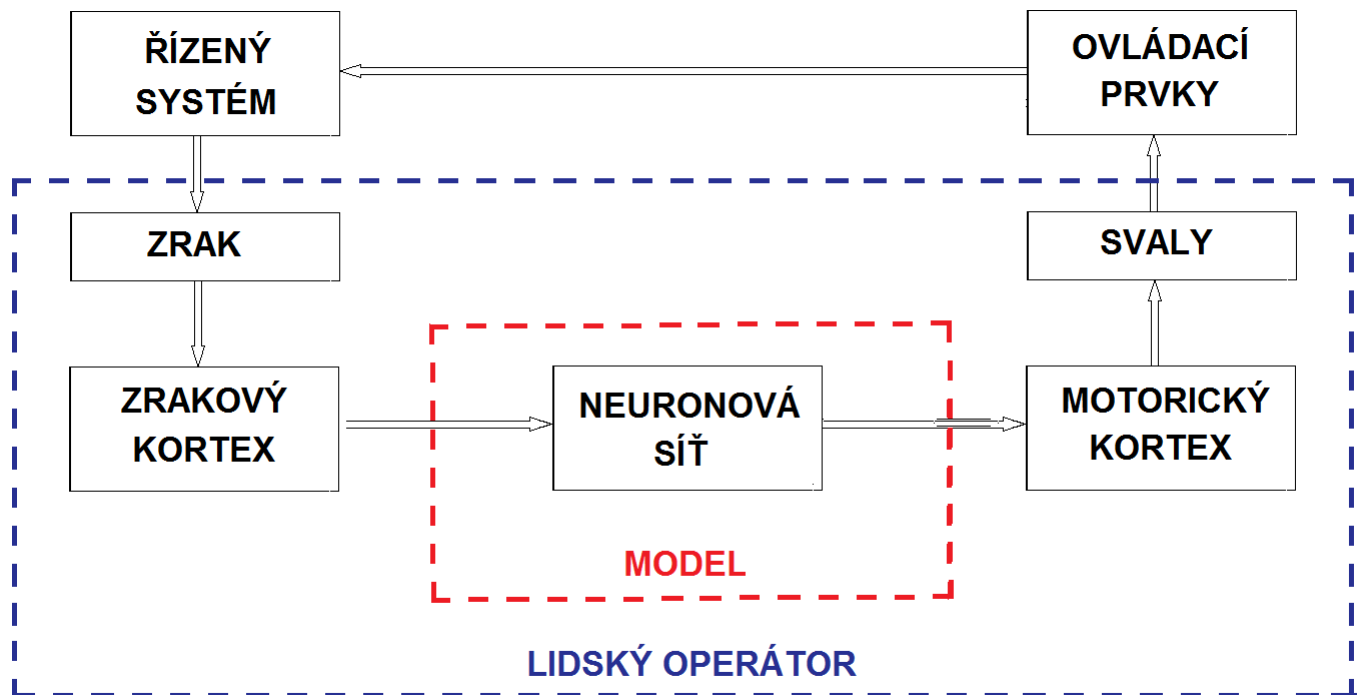
Výstupem je v případě lidského operátora akční zásah, který se projevuje změnou působící síly na ovládací prvky řízeného systému. Tento akční zásah je realizován pomocí svalů. Pro přesný model by tedy výstupem měly být impulsy pro svaly.

Vzhledem k zaměření této práce ale není nutné uvažovat přímo impulsy pro aktuátory, ale je možné zúžit výstup na povel nadřazený konkrétním svalovým impulzům.

V současné době neumíme dát přesnou odpověď na otázku, jakou formu má takovýto povel, ale je možné na základě měření akčních potenciálů alespoň přibližně určit aktivované části mozku.

Podrobnější popis modelované oblasti obsahuje obr. 6.2. Zrak a zrakový kortex jsou nahrazeny zpracováním obrazu a operace metodou PCA, motorický kortex a svaly jsou nahrazeny abstraktním vektorem popisujícím stav systému. Ovládací prvky a řízený systém leží mimo oblast lidského operátora, a proto nejsou předmětem modelu ani jeho aproxi-

mace.



Obrázek 6.2: Vztah použitého modelu a modelovaného systému.

## 6.2 Vyhodnocení kvality celé homeostatické neuronové sítě

Samotné vyhodnocení dat je i v tomto případě poměrně komplikované a neexistuje pro ně univerzální postup. Jakožto vstup do sítě jsou použity uměle generovaná data a obrazová informace.

Výstupem sítě je vektor stavu řidiče, který popisuje míru pravděpodobnosti různých typů špatného rozhodnutí (pozdní reakce, vyjetí z jízdního pruhu, nedodržení odstupu od vpředu jedoucího vozidla a další) pomocí

- intenzity pozornosti
- koherence míry fokusace na sledovanou činnost

### 6.2.1 Statistické vyhodnocení výsledků

Při vyhodnocování kvality nastavení neuronových sítí je důležité počáteční nastavení, které je v prvním kroku náhodné. Důsledkem náhodného nastavení počátečních vah je, že každý běh simulace může skončit s jiným výsledkem. Pro vyloučení vlivu náhodného počátečního nastavení je každá topologie a každá učící metoda neuronové sítě testována ve 100 bězích simulačního programu, ze kterých je spočítán průměr.

Při vyhodnocení jsou sítě samostatně posuzovány z hlediska **rychlosti učení**, tj.

- počet iterací nutných k dosažení požadované funkce
- výpočetní čas ve vteřinách

**generalizace** - tj. schopnost rozpoznávat vzory, na které síť nebyla naučena. Numericky je generalizace vyjádřena jakožto poměr mezi přesností modelu testovacího a trénovacího vektoru

**kontinuita učení** - pokud je síť naučena na daný vzor a změní se vstupní vektor, je určeno, po kolika iteracích dojde k přeučení na nová data tak, aby MSE sítě byla stejná před změnou dat.

---

## Příklad učení vybraného neuronu

### 7.1 Příklad učení vybraného neuronu

Následující část popisuje učení jediného neuronu na jednoduchém vstupním vektoru o 2 dvou prvcích.

Délku vstupního vektoru jsem vybral se zřetelem k lepšímu zobrazení výsledku, neboť váhy neuronu se dvěma vstupy a jedním prahem lze zaznamenat do jediného třídímního grafu.

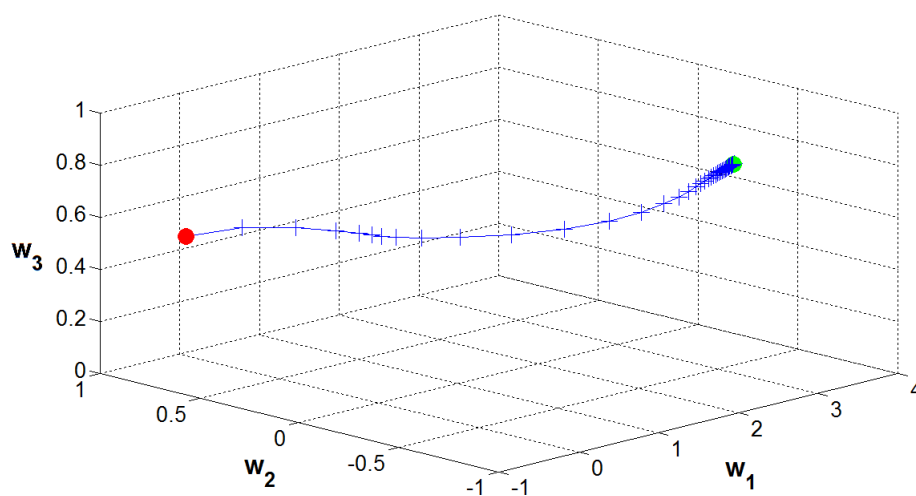
Porovnal jsem učení na základě metody back propagation, jehož ukázka je na obr. 7.1, s učením homeostatického neuronu využívajícího nejvýznamnější váhu (obr. 7.2), náhodně vybranou váhu (obr. 7.3) a postupně vybranou váhu (obr. 7.4).

#### 7.1.1 Porovnání učících algoritmů pro samostatný neuron z hlediska rychlosti učení

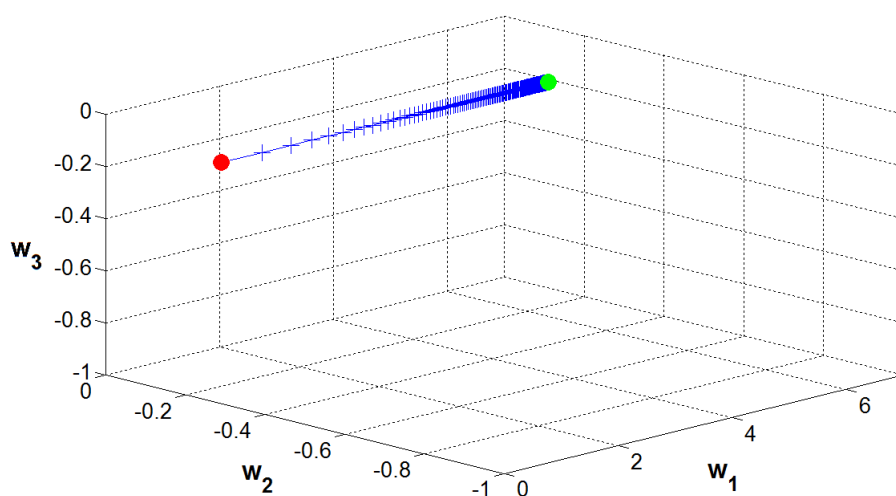
Tabulky 7.1 a 7.2 zobrazují statistické porovnání uvedených 4 typů neuronů pro různý počet vstupů. Jednotlivé statistiky výsledků učícího algoritmu (počet učících cyklů a výpočetní čas) byly získány na základě 100 simulací provedených pro každou kombinaci učícího algoritmu a délky vstupního vektoru, ze kterých byl spočítán průměr. Opakování simulací bylo nutné pro vyloučení vlivu počátečního náhodného nastavení.

Nebylo by „spravedlivé“ porovnávat jednotlivé algoritmy pouze na základě počtu iterací nutných k naučení, neboť v různých algoritmech se v jednotlivých iteracích provádějí různé složité operace. Např. v algoritmech učení náhodně nebo postupně vybrané váhy jsou výpočty v rámci jedné iterace poměrně jednoduché, proto je potřeba větší množství iterací





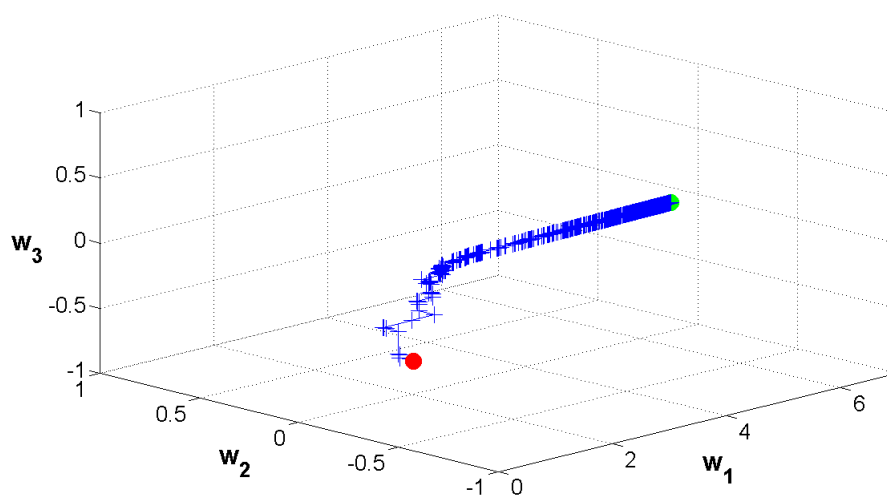
Obrázek 7.1: Učení neuronu metodou back propagation pro jednoduchý případ se dvěma vstupy a jedním prahem (tj. neuron se třemi váhami). Červený bod značí počáteční náhodné nastavení vah a zelený bod konečné váhy (neuron v tomto bodě splnil učící kritérium).



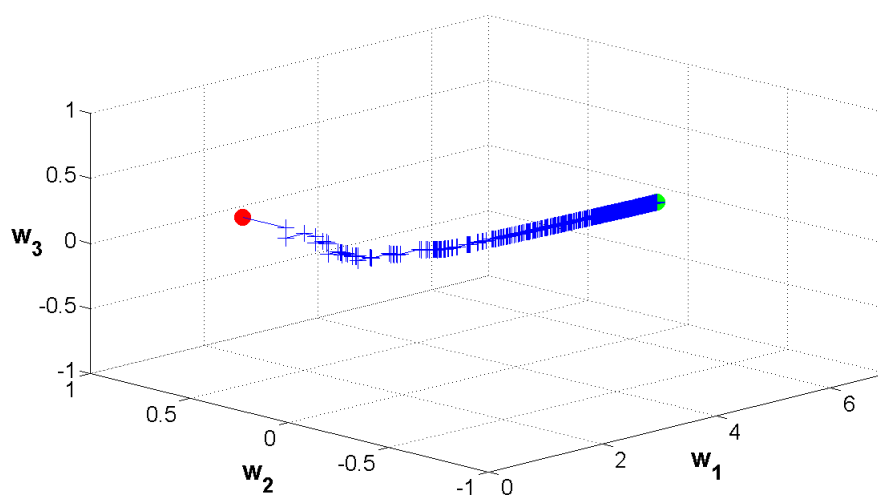
Obrázek 7.2: Učení homeostatického neuronu podle algoritmu úpravy nejvýznamnější váhy.

pro dosažení požadovaných vlastností neuronové sítě. Na druhou stranu výpočetní čas pro provedení jedné iterace je v tomto případě nižší.

Vzhledem ke skutečnosti, že v konečném důsledku je výpočetní čas důležitým kritériem pro praktické využití dané neuronové sítě, využil jsem tento čas jakožto sekundární kritérium vhodnosti testovaného algoritmu.



Obrázek 7.3: Učení homeostatického neuronu podle algoritmu úpravy postupně vybrané váhy.



Obrázek 7.4: Učení homeostatického neuronu podle algoritmu úpravy náhodně vybrané váhy.

Z tabulek 7.1 a 7.2 vyplývá, že z hlediska počtu iterací i výpočetního času lze dosáhnout nejlepších výsledků s algoritmem back-propagation, avšak že tento rozdíl se s přibývajícím počtem vstupů snižuje.

Mezi homeostatickými neurony dosahuje nejlepších výsledků algoritmus učení nejcitlivější váhy z hlediska počtu nutných iterací, z hlediska výpočetního času je tento algorit-

Rychlost učení dle počtu iterací				
Počet vstupů neuronu	Back propagation	Učení nejvýznamnější váhy	Učení postupně vybrané váhy	Učení náhodně vybrané váhy
3	29	31	233	138
4	26	34	683	179
5	31	44	3584	5700
6	32	47	7052	2559
7	34	56	8352	2233
8	32	79	3694	1605
9	43	82	7441	1589
10	38	83	3865	2319

Tabulka 7.1: Počet iterací nutných pro dosažení požadované přesnosti realizované funkce pro jednotlivé typy učících funkcí.

mus jen mírně lepší. S rostoucím počtem vstupů se snižuje relativní rozdíl mezi algoritmem učení nejvýznamnější váhy a algoritmy učení náhodně a postupně vybrané váhy.

### 7.1.2 Porovnání učících algoritmů pro samostatný neuron z hlediska schopnosti generalizace

Generalizace je definována jakožto schopnost neuronové sítě rozpoznávat vzory, na které tato síť nebyla předtím naučena. Tato vlastnost má zásadní význam pro využití neuronové sítě, neboť v dopravě i jinde musí lidský operátor prakticky neustále činit rozhodnutí v situacích, se kterými se dříve nesetkal. Stejná schopnost je proto požadována i od navržené neuronové sítě.

Prakticky jsem schopnost generalizace ověřoval tak, že jsem na vstup neuronové sítě přivedl vektor, který se lišil o 10 % od trénovacího vektoru. Sledovaným parametrem bylo, jak se změna vstupního vektoru projeví na střední kvadratické odchylce neuronu.

Kvůli vyloučení náhodného vlivu jsem tuto simulaci 100 krát opakoval, stejně jak jsem to učinil při měření rychlosti učení.

Výsledky jsou v tab. 7.3.

Z uvedené tabulky je vidět, že nejnižší zhoršení vykonávané funkce má učící metoda

Rychlost učení dle výpočetního času				
Počet vstupů neuronu	Back propagation	Učení nejvýznamnější váhy	Učení postupně vybrané váhy	Učení náhodně vybrané váhy
3	5	11	38	22
4	5	19	136	35
5	6	34	829	1310
6	6	50	1874	675
7	7	77	2488	660
8	6	137	1230	532
9	8	175	2714	577
10	7	217	1543	922

Tabulka 7.2: Výpočetní čas v milisekundách nutný pro dosažení požadované přesnosti realizované funkce pro jednotlivé typy učících funkcí.

back propagation, a že toto zhoršení je téměř nezávislé na délce vstupního vektoru. Mezi metodami homeostatického učení dává nejnižší citlivost na změnu vstupního vektoru učení pomocí úprav nejvýznamnější váhy. Učení postupně přidávané váhy a učení náhodně vybrané váhy dává přibližně stejný výsledek.

Zajímavým zjištěním je, že procentuální odchylka od požadované funkce v případě homeostatického neuronu roste s počtem prvků vstupního vektoru, zatímco v případě metody back propagation je přibližně konstantní.

### 7.1.3 Porovnání učících algoritmů pro samostatný neuron z hlediska kontinuity učení

Kontinuita učení je definována jakožto počet iterací, které jsou nutné k přeučení neuronové sítě, pokud dojde ke změně vstupních nebo výstupních vektorů.

Tato vlastnost má značný význam pro modely mentálních funkcí, neboť lidský operátor pracuje v proměnném prostředí, kde se musí adaptovat na měnící se podmínky. Fáze učení tak splývá s fází vykonávání dané funkce.

Prakticky byla kontinuita učení testována tak, že jsem zafixoval stav neuronové sítě po dosažení požadované střední kvadratické odchylky. Poté jsem provedl malé změny všech

Schopnost generalizace				
Počet vstupů neuronu	Back propagation	Učení nejvýznamnější váhy	Učení postupně vybrané váhy	Učení náhodně vybrané váhy
3	6	55	86	83
4	5	64	90	99
5	6	83	114	75
6	5	86	111	85
7	5	120	119	107
8	4	138	143	112
9	5	251	144	105
10	7	226	92	94
Průměrné zhoršení výsledku sítě	5	127	112	95

Tabulka 7.3: Procentuální zhoršení střední kvadratické chyby po přivedení mírně změněného vzoru na vstup již naučeného neuronu. Pro vyloučení náhodného vlivu jsou uvedena čísla zprůměrována přes 100 simulací pro každou kombinaci.

prvků vstupního vektoru, přičemž tato změna nikdy nepřesáhla hodnotu 0,1. Následně jsem zopakoval učící proces až do opětovného dosažení požadované střední kvadratické odchylky.

Tabulka 7.4 znázorňuje výsledky, které jsem získal zprůměrováním 100 běhů simulačního algoritmu pro vyloučení vlivu náhodné chyby.

Z tabulky 7.4 je vidět, že dosažení nejlepší kontinuity umožňuje algoritmus back propagation. Z hlediska kontinuity učení je pořadí učících algoritmů podobné jako z hlediska rychlosti učení. Je ale zároveň vidět, že rozdíl mezi nejlepším algoritmem (back propagation) a nejhorším (učení náhodně vybrané váhy) je menší než v případě porovnání rychlosti učení.

#### 7.1.4 Celkové srovnání učících algoritmů pro samostatný neuron

Jako první krok k ověření funkčnosti navrženého algoritmu jsem použil porovnání tří modifikací učících algoritmů a algoritmu zpětného šíření chyby. Tyto algoritmy jsem porovnal

Kontinuita učení				
Počet vstupů neuronu	Back propagation	Učení nejvýznamnější váhy	Učení postupně vybrané váhy	Učení náhodně vybrané váhy
3	4	5	92	44
4	3	6	237	63
5	4	6	2418	2968
6	3	6	2795	2034
7	4	7	1382	5149
8	3	12	2040	1077
9	4	11	2411	1005
10	4	11	2940	1783

Tabulka 7.4: Kontinuita učení vyjádřená jakožto počet iteračních cyklů nutných k opětovnému dosažení požadované MSE poté, co došlo k malé změně vstupního vektoru.

ze tří hledisek, a to z hlediska rychlosti učení, schopnosti generalizace a kontinuity učení.

Z hlediska kritéria rychlosti učení dává nejlepší výsledky algoritmus back propagation, a to jak co do počtu iterací, tak co do výpočetního času. Na druhém místě je algoritmus úpravy nejcitlivější váhy homeostatického neuronu. Algoritmus úpravy náhodně vybrané váhy a algoritmus úpravy postupně vybíraných vah dávají přibližně stejné výsledky.

Rychlost učení jsem sledoval nejen na základě počtu iterací, ale také dle výpočetního času. Výpočetní čas je závislý na použitém hardwaru, proto jsem všechny simulace provedl na stejném počítači. Z hlediska výpočetního času je nejúspěšnější algoritmus back propagation, ostatní algoritmy mají přibližně podobné výsledky.

Z hlediska generalizace umožňuje dosažení nejlepších výsledků metoda back propagation. Na druhém místě je metoda učení homeostatického neuronu pomocí nejvýznamnější váhy.

Z hlediska kontinuity učení jsou výsledky podobné jako z hlediska rychlosti učení, tj. nejlepších výsledků dosahuje metoda back propagation. Rozdíl mezi touto metodou a homeostatickým neuronem ovšem již není tak výrazný.

## 7.2 Příklad učení neuronové sítě

Neuronovou síť jsem testoval na dvou typech dat, a to na uměle generovaných datech a na reálných datech.

Umělá data byla získána náhodným generováním vstupních vektorů v programu Matlab. Délka vstupního vektoru se pohybovala od 5 do 100 prvků, délka výstupního vektoru od 1 do 5-ti prvků. Jedinou podmínkou bylo, aby se ve vstupní matici nevyskytovaly dva stejné vstupy s různými výsledky (v takovém případě by nebylo možné neuronovou síť naučit na všechny vektory).

Reálná data mohou mít mnoho podob, v rámci této práce jsem vybral obrazovou informaci databázi MNIST, která obsahuje 60 000 vzorků ručně psaných číslic.

Ve všech případech jsem použil princip úpravy nejvýznamnější váhy, který se nejvíce osvědčil při testech samotného neuronu. Úpravu náhodně vybrané váhy a úpravu postupně vybrané váhy jsem také testoval, ale pro určitě konkrétní kombinace parametrů, nikoliv systematicky jako pro úpravu nejvýznamnější váhy.

Pro učení neuronové sítě jsem využil počítač ferret5 z Centra pro intenzivní výpočty ČVUT, který má 12 procesorů o frekvenci 2,67 GHz a paměti cache 12 MB.

Jedním z požadavků na neuronovou síť je sice to, aby pracovala na běžně dostupném hardwaru, ale ve fázi vývoje a testování je výhodnější využít rychlejší výpočet. Pro vyloučení náhodného vlivu jsem všechny simulace 100 krát opakoval. Výsledky v následujících tabulkách mají význam průměru ze 100 běhů simulačního programu.

### 7.2.1 Příklad učení neuronové sítě umělými daty

Neuronová síť byla trénována a testována na datech, které byly generovány stejným způsobem jako data pro testování samostatného neuronu, tj. jednalo se o vektory s hodnotami  $\{-1;1\}$ .

V případě testování odolnosti vůči degradaci byly tyto vektory navíc upraveny tak, že jednotlivé prvky byly vynásobeny náhodně vygenerovaným číslem, které představuje šum (stejně jako u dat pro samotný neuron platí, že relativní rozdíl mezi původním a upraveným vzorem nesmí přesáhnout 10 %).

Počet neuronů ve vstupní a ve skryté vrstvě je roven délce vstupního vektoru, ve výstupní vrstvě byl vždy jeden neuron.

Tabulka 7.5 obsahuje výsledky pro rychlost učení vyjádřenou počtem iterací pro 4 funkce optimalizace (výpočet užitečnosti jakožto součtu výstupních vah, maxima výstupních

vah, medianu výstupních vah a poloviny nejvyšších vah, definice v kapitole 3.2.2) a pro referenční metodu back propagation. Tato síť měla vždy 1 skrytou vrstvu.

Rychlost učení dle výpočetního času				
	Délka vstupního vektoru			
Kriterium optimalizace	5	10	50	100
$q = \text{sum}(w_o)$	<b>795</b>	304	355	233
$q = \text{max}(w_o)$	931	293	328	<b>212</b>
$q = \text{med}(w_o)$	808	<b>247</b>	385	293
$q = \text{sum}(w_o : w_o > \text{med})$	1108	269	<b>314</b>	238
back propagation	356	183	236	238

Tabulka 7.5: Rychlost učení homeostatické neuronové sítě pro uměle generovaná data. Porovnání s metodou back propagation

Schopnost generalizace neuronové sítě				
	Délka vstupního vektoru			
Kriterium optimalizace	5	10	50	100
$q = \text{sum}(w_o)$	254	<b>55</b>	21	<b>32</b>
$q = \text{max}(w_o)$	<b>150</b>	66	19	33
$q = \text{med}(w_o)$	209	76	22	39
$q = \text{sum}(w_o : w_o > \text{med})$	368	73	<b>17</b>	33
back propagation	30	32	22	43

Tabulka 7.6: Schopnost generalizace homeostatické neuronové sítě pro uměle generovaná data. Porovnání s metodou back propagation.

V tabulkách 7.6 a 7.7 je porovnání stejných učících metod z hlediska schopnosti generalizace a kontinuity učení.

Výsledky naznačují, že z hlediska rychlosti učení je nejvýhodnější optimalizace pro



Kontinuita učení neuronové sítě				
	Délka vstupního vektoru			
Kriterium optimalizace	5	10	50	100
$q = \text{sum}(w_o)$	1561	<b>301</b>	98	<b>101</b>
$q = \text{max}(w_o)$	1446	314	69	162
$q = \text{med}(w_o)$	<b>1259</b>	307	49	182
$q = \text{sum}(w_o : w_o > \text{med})$	1722	472	<b>27</b>	121
back propagation	255	110	39	123

Tabulka 7.7: Kontinuita učení homeostatické neuronové sítě pro uměle generovaná data.

všechny výstupní váhy, tj., užitečnost je rovna součtu výstupních vah.

$$q = \sum_{i=1}^n w_i^o \quad (7.1)$$

Z hlediska schopnosti generalizace je také součet výstupních vah nejvýhodnější. Maximum a median výstupních vah dávají přibližně stejné výsledky. Nejhorší generalizace dosahuje součet horní poloviny výstupních vah.

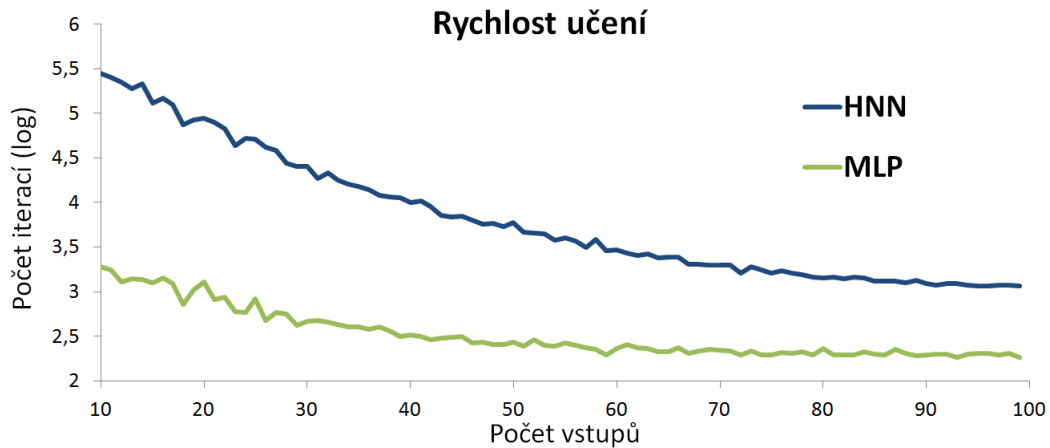
Nejlepší kontinuitu učení má metoda součtu výstupních vah, ostatní 3 metody jsou přibližně podobné.

Učící algoritmus back propagation dosáhl vždy vyšší rychlosti učení. V případě malých sítí (5 a 10 vstupů) má také lepší generalizaci. Pro největší síť se 100 vstupy je schopnost generalizace lepší u homeostatické neuronové sítě než u algoritmu back propagation. Kontinuita učení algoritmu back propagation je v případě malého počtu vstupů výrazně lepší než u homeostatické neuronové sítě, pro větší síť je přibližně konstantní.

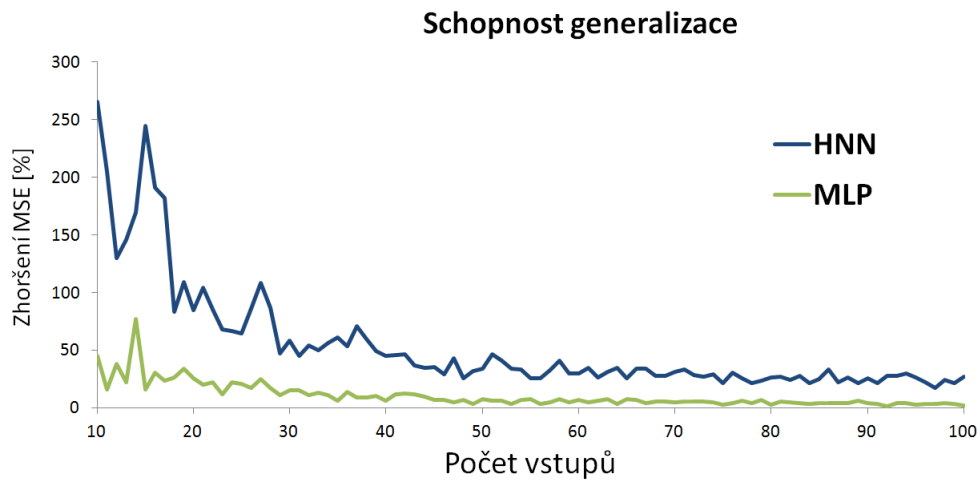
Se zvyšujícím se počtem vstupů se výsledky jednotlivých metod přibližují.

Na základě testů s jednou skrytou vrstvou jsem dospěl k závěru, že nejvýhodnější je výpočet užitečnosti dle součtu výstupních vah.

Pro porovnání vlastností učících algoritmů jsem provedl simulaci, při které jsme počet vstupů zvyšoval od 10 do 100. Při každém počtu vstupů jsem porovnal rychlost učení homeostatické neuronové sítě s výpočtem užitečnosti dle součtu výstupních vah vůči metodě back propagation. Z grafu 7.5, ve kterém jsou výsledky, je patrné, že s rostoucím počtem vstupů se snižuje rozdíl mezi homeostatickou neuronovou sítí a metodou back propagation.



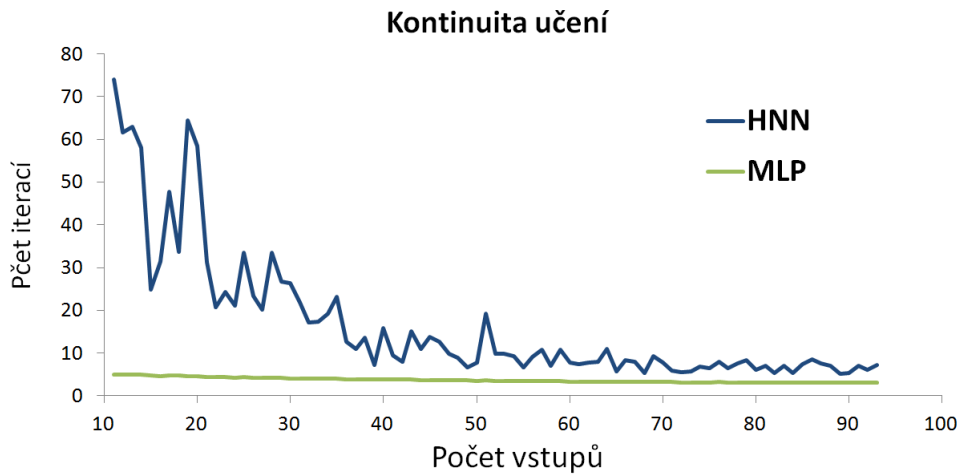
Obrázek 7.5: Rychlost učení algoritmu back propagation a homeostatické neuronové sítě učící se na principu součtu výstupních vah. Z grafu je patrné, že s rostoucím počtem vstupů dosahuje metoda back propagation podobných výsledků. Pro přehlednost je rychlost učení vyjádřena logaritmicky. Hodnoty byly získány zprůměrování 100 běhů výpočtu pro každý počet vstupů.



Obrázek 7.6: Porovnání schopnosti generalizace algoritmu back propagation a homeostatické neuronové sítě. Je zřejmé, že pro větší sítě dávají obě metody podobné výsledky.

## 7.2.2 Příklad učení neuronové sítě s reálnými daty

K otestování konceptu homeostatického neuronu jakožto součásti neuronové sítě jsem použil známou databázi MNIST (obr. 7.8). Tato databáze obsahuje 60 000 digitalizovaných vzorků ručně psaných číslic, které poskytlo American Census Bureau. Tato data jsou volně



Obrázek 7.7: Kontinuita učení pro homeostatickou neuronovou síť a síť s učením typu back propagation.

k dispozici na internetu, např. v rámci soutěže KAGGLE ([www.kaggle.com](http://www.kaggle.com)).



Obrázek 7.8: Příklad vrozků z databáze MNIST

Pomocí těchto dat jsem porovnal výsledek neuronové sítě složené z navržených neuronů vůči učící metodě back – propagation. Kritéria pro srovnání jsou stejná jako v případě porovnání samotných neuronů.

Pro ohodnocení kvality neuronové sítě jsem použil procentuální úspěšnost rozpoznávání vzorů, nikoliv střední kvadratickou odchylku jako v případě učení samotného neuronu. Požadována byla alespoň 95 % úspěšnost rozpoznávání. Důvodem, je že v tomto případě mají data reálnou reprezentaci a proto vyjádření úspěšnosti rozpoznávání poskytuje lepší představu o výsledku sítě než střední kvadratická chyba.

Ohodnocení neuronové sítě pro reálná data			
	Kritérium ohodnocení sítě		
Kritérium optimalizace	Rychlost učení	Schopnost generalizace	Kontinuita učení
$q = \text{sum}(w_o)$	<b>3584</b>	36	<b>215</b>
$q = \text{max}(w_o)$	4625	54	645
$q = \text{med}(w_o)$	6748	<b>24</b>	856
$q = \text{sum}(w_o : w_o > \text{med})$	5486	36	436
back propagation	1163	11	317

Tabulka 7.8: Rychlost učení homeostatické neuronové sítě pro reálná data. Porovnání s metodou back propagation. Topologie sítě je 748-1000-1.

---

## Diskuze a nástin pokračování práce na tématu homeostatické neuronové sítě

Provedené experimenty prokázaly schopnost učení neuronové sítě na základě principu hledání homeostatické pozice. Navržená síť dosahuje z hlediska rychlosti učení horších výsledků než jiné metody umělé inteligence, přesto je její použití opodstatněné, neboť v jiných ohledech jsou její výsledky srovnatelné.

Důležitou vlastností je podobnost s biologickou neuronovou sítí, a z ní vyplývající vlastnosti jako je kontinuita učení, odolnost vůči poškození vstupního vzoru nebo odolnost vůči přeučení.

V rámci časových možností práce na této dizertaci jsem nemohl detailně prozkoumat aplikace popsaného učícího algoritmu pro řešení reálných úloh. S ohledem na zkušenosti získané při testování se ale domnívám, že homeostatickou síť lze využít zejména v těch situacích, kde je požadována kontinuita a stabilita učícího procesu. Příkladem může být sledování úrovně pozornosti řidiče – v tomto případě se sledovaný objekt mění s časem, ale přitom zachovává určité vlastnosti. Systém pro sledování řidiče proto musí být schopen adaptovat se na měnící se projevy řidiče.

Podařilo se mi otestovat neuronovou síť na databázi MNIST, která se principiálně podobá zpracování obrazové informace při dopravních procesech. Druhá část, tj. rozpoznávání reálných obrazů s nimiž se setkává člověk, není dosud dokončena.

Druhou zajímavou vlastností je odolnost vůči fenoménu známému pod označením „catastrophic interference“. Tento problém způsobuje degradaci funkce neuronové sítě, která je přeučena na nové vzory. Síť, která byla již dříve úspěšně naučena na určité vzory, a ke které je přiveden nový vzor, při učení tohoto nového vzoru ztratí schopnost rozpoznávat

---

původní vzor, tedy zapomene již naučené vzory. Toto je jeden z typických problémů sítí založených na principu učení back-propagation. V tomto ohledu je značný rozdíl oproti lidskému mozku, který má schopnost pamatovat si nové vjemy a zároveň nezapomenout dříve naučené.

Kvantitativní vyjádření odolnosti vůči katastrofic interference je poměrně obtížné a závisí na použitých datech, nicméně dosažené výsledky naznačují, že homeostatické neuronové sítě v tomto směru dosahují lepších výsledků než sítě naučené metodou back propagation.

Navržená neuronová síť, stejně jako i jiné neuronové sítě, byla učena heuristickou metodou, která nezaručuje nalezení optimálního řešení, ani předem nejsou známy parametry hledané sítě. Z této skutečnosti vyplývá nutnost hledání parametrů (např. topologie nebo rychlosti učení) pomocí rozsáhlého testování.

Domnívám se, že pro budoucí využití této metody je potřeba se zaměřit zejména na optimalizaci sítí pro uměle vytvořená data.

Práce s reálnými daty se sice na jednu stranu zajímavější, na druhou stranu nemůže být úspěšná, dokud nebudou dostatečně prozkoumány teoretické vlastnosti tohoto druhu neuronových sítí. Zejména důležitá je závislost výsledku na parametrech sítě (např. počet vrstev, počet neuronů ve vrstvách atd.).

Vzhledem k této skutečnosti bych rád zaměřil budoucí výzkum právě na učení sítí s umělými daty.

---

## Závěr

Cílem této dizertační práce bylo prozkoumat možnosti modelování duševních funkcí lidského operátora pomocí neuronových sítí a navrhnout umělé neuronové sítě schopné modelovat určitou činnost. V první fázi jsem provedl analýzu podobných projektů zabývajících se modelováním biologických neuronových sítí. Na základě výsledků těchto projektů a praktických zkušeností s neuronovými sítěmi jsem navrhl úpravu učicího algoritmu. Na tento učicí algoritmus jsou kladeny tyto dvě základní požadavky:

- učení umělé neuronové by mělo probíhat na podobném principu jako u biologické neuronové sítě
- učicí pravidlo by mělo být dostatečně jednoduché, aby bylo možné provádět simulace na běžně dostupném hardwaru

Z první podmínky vyplývá nemožnost využít běžné algoritmy využívající učení s učitelem, neboť v tom případě by se učení ve svém principu vzdalovalo od biologického vzoru.

Na základě druhé podmínky není možné vytvářet detailní simulaci biologického neuronu.

Ve druhé fázi jsem navrhl učicí algoritmus, který byl pracovně nazván „homeostatický neuron“. Důvodem k tomuto pojmenování je analogie procesem hledání homeostatické polohy, což je fenomén dobře známý z biologie. Proces učení (tj. nastavování svých vlastních vah) probíhá na základě sledování parametrů ostatních propojených neuronů. V tomto algoritmu není potřeba účasti žádného učitele ani jiného nadřazeného systému, který by modifikoval váhy daného neuronu.

---

Navrhl jsem několik způsobů, jak tento homeostatický neuron realizovat. Některé z těchto myšlenek se časem ukázaly jako slepé cesty a dál jsem se jimi nezabýval, zato jiné varianty se jeví jako nadějně.

V průběhu práce se jakožto hlavní překážka a zároveň výzva ukázala obrovská komplexnost sledovaných neuronových sítí, která je ještě znásobena při pokusech o praktickou aplikaci. Neuronové sítě jsou vždy závislé na použitých datech, proto daná neuronová síť či učící algoritmus může přinášet dobré výsledky s určitými daty, ale horší výsledky s jinými daty. Uvážíme-li obrovské množství typů dat, které zpracovává lidský mozek, bylo nemožné ověřit všechny navržené varianty neuronové sítě pro všechny typy dat, které by mohly být zajímavé v dopravních aplikacích.

Z tohoto důvodu jsem se soustředil pouze na ty typy neuronových sítí, které v úvodním testování přinášeli dobré výsledky.

Jednalo se konkrétně o sítě, které neomezuji váhy a které maximalizují svou užitečnost pro omezený počet neuronů ve vyšší vrstvě.

Tato úvaha ale neznamená, že by ostatní metody byly špatné – je možné, že v určitých situacích mohou být zamítnuté metody výhodnější než ty, které jsem ověřoval.

Oproti původnímu plánu se nepodařilo optimalizovat síť pro zpracování dat z elektroencefalogramu (EEG). Důvodem k tomu byla vysoká komplexnost těchto dat. Proto jsem se soustředil na návrh samotné sítě a její testování s uměle generovanými daty nebo s obrazovými daty, jejichž získání jakož i reprezentace jsou výrazně jednodušší. Zpracování obrazové informace je velmi specifická oblast, která je předmětem intenzivního zájmu mnoha jiných metod.

Navzdory očekávání se jakožto hlavní problém ukázala nikoliv rychlost učení, nýbrž otázka zpoždění. Navržený učící algoritmus totiž vede k tomu, že neurony má v daný okamžik dopředný signál a informaci o užitečnosti, které si časově neodpovídají.

Jakožto nejvýhodnější řešení tohoto problému se ukázalo nastavení dynamiky vstupů na výrazně nižší úroveň, než jaká je rychlost komunikace uvnitř neuronové sítě. Vůči tomuto řešení existuje oprávněná námitka, že se učící princip rozchází s biologickou realitou. Na druhou stranu je toto řešení podstatně věrnější než druhá možnost, tj. vybavení neuronu pamětí. Navíc ve většině praktických úloh je skutečně dynamika vnějších procesů výrazně pomalejší, než komunikace uvnitř mozku.

Zajímavá a dosud nezodpovězená zůstává otázka, zda bude možné použít síť i pro jiné úlohy, než pro jaké byla navržena (tj. pro modelování duševních funkcí člověka).

Vzhledem ke skutečnosti, že neuronové sítě již mnohokrát prokázaly svou využitelnost



---

pro zcela jiné obory, než pro které byly navrženy, je možné, že i tento učící algoritmus bude možné použít pro jiná data.

Navzdory řadě nevýhod, které má navržená síť oproti existujícím modelům neuronu a umělým neuronovým sítím, má tato síť potenciál přispět k modelům duševních funkcí díky novému přístupu k procesu učení. Zejména pro vyšší mozkové funkce, jako je rychlé rozhodování při nedostatku informací, není možné vytvořit deduktivní model stylem „shora“, ale je možné vytvořit model na základě pozorovaných jevů. Nevýhodou tohoto přístupu je, že získaná znalost zůstane „uložena“ v neuronové síti v podobě, ve které bude velmi obtížné jí porozumět, bude se jednat o černou skříňku (s výjimkou velmi jednoduchých sítí, což ovšem není tento případ). Toto je ale nevýhoda pouze zdánlivá, protože s modelem je možné pracovat, i když není možné explicitně vyjádřit pravidla v něm obsažená. Navíc se jedná spíše o „šedou“ skříňku - strukturu sítě lze částečně dekodovat, pouze to je obtížnější než u jiných metod umělé inteligence.

Model lidského operátora založený na homeostatických neuronových sítích má potenciál přispět k testování nejrůznějších situací, návrhů asistenčních systémů, ohodnocení dopravních řešení apod. a tím zjednodušit bezpečnostní analýzu.

---

## Použitá literatura

- [1] Novák, M *Umělé neuronové sítě: teorie a aplikace*. Praha, C. H. Beck 1998 ISBN 80-7179-132-6
- [2] Abbott L. F *Lapique's introduction of the Integrate-and-Fire model neuron (1907)*. Brain Research Bulletin 50 (5/6) 1999 s. 303-304.
- [3] Feng, J *Is the integrate-and-fire model good enough?—a review*. In: Neural Networks Vol. 14 2001 s. 955-975
- [4] Burkitt A *A Review of the Integrate-and-fire Neuron Model: I. homogeneous Synaptic Input*. In: Biological Cybernetics 2006 DOI 10.1007/s00422-006-0068-6
- [5] Cronin J *Mathematical aspects of Hodgkin-Huxley neural theory*. Cambridge University Press USA, 1987 ISBN 0 521 33482 9
- [6] Hodgkin A.L., Huxley A.F. *A quantitative description of membrane current and its application to conduction and excitation in nerve*. In: J. Physiology 1952 s. 500-544
- [7] Rocsoreanu, C., Georgescu, A., Giurgiteanu, N. *The Fitzhugh-Nagumo model Bifurcation and Dynamics* . Nizozemí 2000 ISBN 0-7923-6427-9
- [8] McCulloch, W. S., Pitts, W *A Logical Calculus of the Ideas Immanent in Nervous Activity*. In: Bulletin of Mathematical Biophysics Vol 5 1943 s. 115-133
- [9] Šnorek M., Jiřina M *Neuronové sítě a neuropočítače*. ČVUT 1996
- [10] Bishop Ch. *The Perceptron*. In Neural Networks for Pattern Recognition Oxford University Press 2008 ISBN 978-0-19-853864-6

- 
- [11] Levine D S *The Back Propagation Networks and its Variants*. In Introduction to neural and cognitive modeling, Lawrence Erlbaum Associates 2000 ISBN 0-8058-2005
- [12] Farley, B.G.; Clark, W. *Simulation of self-organizing systems by digital computer*. Transactions of the IRE Professional Group on Information Theory Vol 4 1954 s. 76-84
- [13] Hopfield, J. J *Neural Networks and physical systems with emergent collective computational abilities*. Neural Networks and physical systems with emergent collective computational abilities, Proceedings of the national academy of sciences of the United States of America-Biological sciences Vol 79 1982 s. 2554-2558
- [14] Hopfield, J. J. *Artificial neural networks*. IEEE Circuits and Devices Magazine Vol. 4 1988 s. 3-10
- [15] Freeman J *The discrete Hopfield network*. In Simulating neural networks with Mathematica Addison-Wesley 1994 ISBN 0-201-56629-X
- [16] Ivakhnenko, A. G *Polynomial Theory of Complex Systems*. IEEE Transactions on Systems, Man and Cybernetics Vol 4 1971 s. 364-378
- [17] Faber J., Novak M *Thalamo-cortical reverberation in the brain produces alpha and delta rhythms as iterative convergence of fuzzy cognition in an uncertain environment*. In Neural Network World Vol. 2 2011 s. 169-192
- [18] Faber J. *Bulbo-thalamo-cortical structures and their activities simulated by "Formator complex" and "multilayered iterative algorithm" Models*. Univerzita Karlova v Praze 2015 Monografie CLX ISBN 978-80-246-3101-1
- [19] Kondo T., Pandya A.S *GMDH-type Neural Networks with a Feedback Loop and their Application to the Identification of Large-spatial Air Pollution Patterns*. In Proceedings of the 30th SICE Annual Conference International Session Papers (The Society of instrument and Control Engineers) Iizuka Japonsko, 2000 s. 1-6
- [20] Kondo T *Revised GMHD-type neural networks with radial basis functions and their application to medical image recognition of stomach* In System Analysis Modelling Simulation 2003 Vol. 43, No. 10 s. 1368-1376

- 
- [21] Kondo T *Identification of Radial Basis Function Networks by using Revised GMDH-type Neural Network with a Feedback loop*. In Proceedings of the 41st SICE Annual Conference 2002 Vol 5 s 2672 - 2677, DOI: 10.1109/SICE.2002.1195514
- [22] Miyagishi K., Masami O., Ichihashi H *Temperature Prediction From Regional Spectral Model by Neurofuzzy GMDH*. INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH 33 Vol 5 1995, s 1229-1240
- [23] Homma N., Gupta M.M *Study on general second-order neural units (SONUs)*. MULTIMEDIA, IMAGE PROCESSING AND SOFT COMPUTING: TRENDS, PRINCIPLES AND APPLICATIONS, 2002, Vol. 13, s. 177-182
- [24] Gupta M.M. et al *Static and Dynamic Neural networks: From Fundamentals To Advances Theory* . IEEE Press and Wiley-Interscience, 2003
- [25] Bukovsky I. *Modeling of Complex Dynamic Systems by Nonconventional Artificial Neural Architectures and Adaptive Approach to Evaluation of Chaotic Time Series* . ČVUT FS, dizertační práce, 2007
- [26] Hoon-Heng Teh *What is Human Intuition*. In: Neural Logic Networks, World Scientific, ISBN 981-02-1844-3
- [27] Markram H. et al *Introducing the Human Brain Project* . In: The European Future Technologies Conference and Exhibition, 2011, Vol. 7, s. 39-42
- [28] Peck Ch et al *Network-Related Challenges and Insights from Neuroscience* . In Bio-Inspired Computing and Communication, BIOWIRE -FirstWorkshop on Bio-Inspired Design of Networks, 2007 Cambridge, s. 67-78
- [29] Tauheed F, Biveinis L, Heinis T, Schürmann F, Markram H *Accelerating Range Queries For Brain Simulations* . In IEEE 28th International Conference on Data Engineering, 2012, s. 941-952
- [30] Hines M, Markram H., Schürmann F *Fully implicit parallel simulation of single neurons* . In Journal of Computational Neuroscience, 2008, s. 439-448, DOI 10.1007/s10827-008-0087-5
- [31] Migliore M et al *Parallel network simulations with NEURON* . In Journal of Computational Neuroscience 2006, s 119-129, DOI 10.1007/s10827-006-7949-5

- 
- [32] Michael W. et al *A Biophysically Detailed Model of Neocortical Local Field Potentials Predicts the Critical Role of Active Membrane Currents* . In: *Neuron*, Vol 79, 2013, s. 375-390
- [33] Lasserre et al *A Neuron Membrane Mesh Representation for Visualization of Electrophysiological Simulations* . In *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, VOL. 18, NO. 2, 2012, s. 214-227
- [34] Perin R, Telefont M., Markram H. *Computing the size and number of neuronal clusters in local circuits*. *Frontiers in Neuroanatomy*, Vol 7, 2013
- [35] Ramaswamy S., Markram H *Anatomy and physiology of the thick-tufted layer 5 pyramidal neuron* . *Frontiers Cell Neuroscience*, 2015
- [36] Modha D. S., Raghavendra S *Network Architecture of the Long Distance Pathways in the Macaque Brain* . In: *Proceedings of the National Academy of the Sciences, USA*, 2010, s. 13485-13490
- [37] Modha D. S., Ananthanarayanan R., Esser S. K. et al *Cognitive Computing*. In *COMMUNICATIONS OF THE ACM*, Vol 8, 2011, s. 62-71, DOI :10.1145/1978542.1978559
- [38] , P., Arthur, J., Akopyan, F., Imam, N., Manohar, R., Modha, D. S *A Digital Neurosynaptic Core using Embedded Crossbar Memory with 45pJ per spike in 45nm*. In: *IEEE Custom Integrated Circuits Conference*, 2011, IEEE, s. 1-4, DOI: 10.1109/CICC.2011.6055294
- [39] Imam N *A Digital Neurosynaptic Core Using Event-Driven QDI Circuits* . *IEEE 18th International Symposium on Asynchronous Circuits and Systems*, 2012, s. 25 – 32, DOI 10.1109/ASYNC.2012.12
- [40] Seo, J. et al *A 45nm CMOS neuromorphic chip with a scalable architecture for learning in networks of spiking neurons* . *IEEE Custom Integrated Circuits Conference (CICC)*, 2011, s: 1 - 4, DOI: 10.1109/CICC.2011.6055293
- [41] Arthur, J.V et al. *Building block of a programmable neuromorphic substrate: A digital neurosynaptic core* . In: *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, s: 1 - 8, DOI: 10.1109/IJCNN.2012.6252637

- 
- [42] Esser, S.K. Ndirango, A., Modha, D.S. *Binding sparse spatiotemporal patterns in spiking computation* Neural Networks (IJCNN), The 2010 International Joint Conference on, s. 1 - 9, DOI: 10.1109/IJCNN.2010.5596925
- [43] Rodriguez, A.L., Cornelis, H., Beeman, D., Bower, J.M. *Multiscale modeling with GENESIS 3, using the G-shell and Python* In BMC Neuroscience 2012, 13(Suppl 1) doi:10.1186/1471-2202-13-S1-P176
- [44] Kunkel S. et al *Spiking network simulation code for petascale computers* In Frontiers in neuroinformatics, Vol 8, 2012 s. 1-23,. doi:10.3389/fninf.2014.00078
- [45] Mead, C *Neuromorphic electronic systems* In: Proceedings of the IEEE, 1990, Vol 78, Is 10, s. 1629-1636, DOI: 10.1109/5.58356
- [46] Rasmussen, J *Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models* IEEE Transactions on Systems, Man and Cybernetics, Vol 13, Issue 3, 1983 s. 257-266, DOI: 10.1109/TSMC.1983.6313160
- [47] D’Rozario, Dungan et al *An automated algorithm to identify and reject artefacts for quantitative EEG analysis during sleep in patients with sleep-disordered breathing* In: Sleep and Breathing, Září, 2014
- [48] Buteneers P., Verstraetena D., Mierloa Pieter van, et al *Automatic detection of epileptic seizures on the intra-cranial electroencephalogram of rats using reservoir computing Artificial Intelligence* In Medicine, Vol 53, Issue 3, 2011 s. 215-223
- [49] Prochazka A., Kukul J *Transform Use for Feature Extraction and EEG Signal Segments Classification,* Communications, Control and Signal Processing, 2008, ISBN: 978-1-4244-1687-5
- [50] Ripley B D *Feed Forward Neural Network* In: Pattern Recognition and Neural Networks, Cambridge University Press, 2009, ISBN 978-0521-71770-0
- [51] Bode, H.W *Network Analysis and Feedback Amplifier Design* Van Nostrand, New York, 1951

---

## Publikace autora k tématu homeostatická neuronová síť

- [1] Růžek M, Brandejský T *Artificial Neuron with Homeostatic Behaviour* In The 2010 European Simulation and modelling conference, Ghent, Belgie 2010, ISBN 978-90-77381-57-1, s. 139 - 141
- [2] Růžek M, *Artificial neuron with homeostatic learning* In 16th International Conference on Soft Computing, Brno 2010, ISBN 978-80-214-4120-0, s. 242 - 245
- [3] Růžek M, , Brandejský T *Model of homeostatic artificial neuron* in Neural Networks, Fuzzy Systems & Evolutionary Computing, ISBN 978-960-474-195-3, Athens 2010, s. 145 - 148
- [4] Růžek M, Brandejský T *Model of biological ANN based on homeostatic neurons* In 12th WSEAS International Conference on Neural Networks (NN'11), ISBN 978-960-474-292-9, Athens 2011, s. 66 - 69
- [5] Růžek M, *Artificial Neural network inspired by homeostasis in biological networks* In Proceedings of 17th International Conference on Soft Computing (MENDEL 2011), ISBN 978-80-214-4302-0, s. 232 - 235
- [6] Růžek M, *Artificial Neural Networks for Models of Driver's Brain Functions* In: 20th Anniversary of the Faculty of Transportation Sciences, Czech Technical University in Prague - Selected Papers, ČVUT v Praze, 2013, ISBN 978-80-01-05320-1, s. 207 - 211

- [7] Růžek M, *Modeling of EEG Signal with Homeostatic Neural Network* In: Nostradamus 2013: Prediction, Modeling and Analysis of Complex Systems, Ostrava 2013, ISBN 978-3-319-00541-6 s. 175 - 180