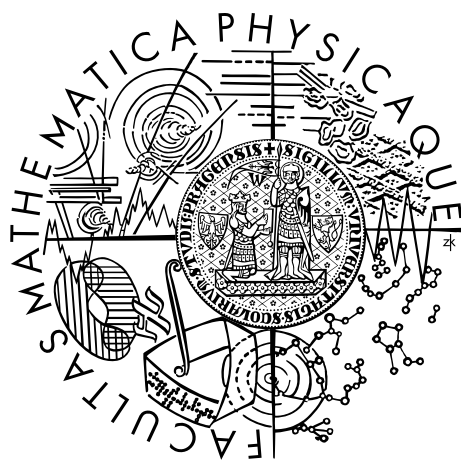


Univerzita Karlova v Praze  
Matematicko-fyzikální fakulta

# DIPLOMOVÁ PRÁCE



Jakub Krchák

Rozpoznávání strukturovaných zvuků pomocí synchronizace v  
neuronové síti

Katedra teoretické informatiky a matematické logiky

Vedoucí diplomové práce: Doc. RNDr. Petr Maršálek, PhD., Ústav patologické  
fyziologie, 1. lékařská fakulta UK

Studijní program: Informatika, teoretická informatika, umělá inteligence

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 13. prosince 2006

Jakub Krchák

# Obsah

<b>1</b>	<b>Předmluva</b>	<b>1</b>
<b>2</b>	<b>Úvod</b>	<b>2</b>
<b>3</b>	<b>Neuron</b>	<b>4</b>
3.1	Biologický neuron . . . . .	4
3.1.1	Dendrity . . . . .	4
3.1.2	Soma . . . . .	5
3.1.3	Axon . . . . .	5
3.1.4	Akční potenciály . . . . .	5
3.2	Umělý neuron . . . . .	9
3.3	Spiking neuron . . . . .	11
3.3.1	Integrate and fire neuron . . . . .	13
3.3.2	Leaky integrate and fire neuron . . . . .	13
3.3.3	Pyramidová buňka . . . . .	14
<b>4</b>	<b>Původní neuronová síť</b>	<b>15</b>
4.1	Vrstva A . . . . .	17
4.2	Vrstva W . . . . .	18
4.2.1	$\alpha$ neurony . . . . .	19
4.2.2	$\beta$ neurony . . . . .	20
4.2.3	$\gamma$ neurony . . . . .	20
<b>5</b>	<b>Prostor vstupů - přirozená řeč</b>	<b>22</b>
5.1	Lidská řeč . . . . .	22
5.2	Zvuk v digitální podobě . . . . .	23
5.3	Zpracování přirozené řeči . . . . .	24
5.4	Kortikální reprezentace zvuku . . . . .	25

<b>6</b>	<b>Řešení</b>	<b>26</b>
6.1	Postup hledání parametrů řešení . . . . .	26
6.2	Algoritmus propojení neuronů . . . . .	28
6.3	Vlastnosti paralelního propojení . . . . .	29
6.4	Doplňování vzorů . . . . .	32
6.5	Nastavování paralelních vah . . . . .	34
6.6	Problém prahů neuronů výstupní vrstvy . . . . .	35
6.7	Kapacita sítě . . . . .	36
<b>7</b>	<b>Technická implementace</b>	<b>38</b>
7.1	Třída CSynapse . . . . .	38
7.2	Třídy CNeuron a CLIAFNeuron . . . . .	40
7.3	Hlavní program . . . . .	42
<b>8</b>	<b>Porovnání se současnou technologií rozpoznávání řeči</b>	<b>43</b>
<b>9</b>	<b>Závěr</b>	<b>45</b>
9.1	Diskuse řešení . . . . .	45
9.2	Možná zlepšení . . . . .	46
9.3	Závěr . . . . .	47

**Název práce:** Rozpoznávání strukturovaných zvuku pomocí synchronizace v neuronové síti

**Autor:** Jakub Krchák

**Katedra:** KTIML MFF UK

**Vedoucí diplomové práce:** Doc. RNDr. Petr Maršálek, PhD.

**e-mail vedoucího:** marsalek@cesnet.cz

**Abstrakt:** Tato práce studuje fenomén rozpoznávání zvuků pomocí synchronizace v neuronové síti neuronů s akčním potenciálem. Používá vstupní vrstvu pro rozpoznávání příznaků v krátkém zvuku, který odpovídá slabice. Prostřední vrstva je nastavená tak, aby v určitém okamžiku konvergovala frekvence aktivování neuronů, odpovídajících naučenému vzoru. To následně vybudí výstupní neuron, který přísluší danému vzoru.

**Klíčová slova:** umělá neuronová síť, spiking neuron, rozpoznávání řeči.

**Title:** Recognition of structured noises by neural network synchronization

**Author:** Jakub Krchák

**Department:** Department of theoretical informatics and mathematical logic at MFF UK, Prague

**Supervisor:** Doc. RNDr. Petr Maršálek, PhD.

**Supervisor's e-mail address:** marsalek@cesnet.cz

**Abstract:** This work studies the phenomenon of sound recognition through spiking neuron network synchronization. The input layer fires on specific features in the input sound, which resemble syllable. The neurons in the middle layer are interconnected in such a way that they prolong their firing rates if the firing frequency is similar. This causes the firing of the output neuron of the corresponding pattern.

**Keywords:** artificial neural network, spiking neuron, voice recognition.

# 1 Předmluva

Sluch lze považovat za nepostradatelný lidský smysl, minimálně proto, že jako hlavní formu mezilidské komunikace používáme řeč. Právě sociální interakce, postavená na rozvoji mluveného slova, se stala tím největším důvodem k tak markantnímu zvětšování objemu mozkovny u historických lidí - tedy vzestupu druhu homo vůbec. Proto je zřejmé, že lidský mozek se musel naučit mluvenému slovu dobře rozumět. Zpracovává samozřejmě všechny zvuky, dráždící naše ušní bubínky, ale při použití moderních zobrazovacích metod zjistíme největší aktivaci mozku právě při zpracování řeči. Pochopit přesně, jak funguje proces poznávání jednotlivých slov sluchovými centry, je důležitou otázkou na cestě za dalším pochopením mozku. Jako možný prostředek našeho sebezkoumání a pokus porozumět jedné z nejkomplexnějších struktur v rámci známého vesmíru - lidskému mozku - nabízí současná věda k použití počítačové simulace. I když se jedná o zjednodušené modely, jsou při zkoumání mozku velmi užitečné a díky nim se podařilo nahlédnout některé souvislosti. My se pokusíme pomocí umělých neuronových sítí vybudovat systém, schopný rozpoznávat lidská slova, inspirovaný biologickým mozkiem.

## 2 Úvod

V práci jsme vypracovali řešení problému, zadaného C. Brodym a J. Hopfieldem v člancích [1, 2]. Článek [1] popisuje jednoduchý "organismus", který provádí rozpoznávání složité časové sekvence, která s postupem času vyžaduje integraci informace - v tomto případě rozpoznávání jednoslabičných slov. Výpočet sítě lze pochopit aplikací jednoduchých principů popisujících nervovou aktivitu. Organismem je síť jednoduchých neuronů a synapsí. Článek popisuje síť tak, jak bychom popisovali biologický organismus. Popis principů za chování sítě je pozdržen (vydán následně v [2]) a článek tak má tvořit jakousi mentální výzvu čtenáři, aby z popsáných vlastností organismu vydedukoval podstatu principů.

Řešení [1] pracuje na principu časové synchronizace neuronů a její výstupní neurony používají funkci, kterou je možné popsat jako "MANY\_ARE\_NOW\_EQUAL". Pracovali jsme na několika vylepšeních původní práce. Předkládáme některá z nich i nápady na jiná. Dále zdůvodňujeme, proč se některými cestami z původní práce dost dobře vydat nedá.

V první kapitole popisujeme cestu od biologického neuronu - pochopitelné to inspirace všech neuronů umělých/matematických. Cesta vede přes perceptron - jednoduché, ale elegantní matematické zjednodušení biologického konceptu. Končí pak modelem, který používáme - spiking neuronem. Pochopitelně existuje množství různých implementací tohoto typu umělých nervových jednotek, my popisujeme některé z nich.

V další kapitole následuje detailní popis neuronové sítě, jak ji používáme pro naši úlohu. Rozdělení do jednotlivých vrstev dle jejich významu a detaily jednotlivých typů spiking neuronů. Rovněž zde nalezneme popis mechanismů, které dávají síti schopnosti rozpoznávat zvuky. Je zde její rozdělení na příznakové kanály a také to, jaké obsahuje skupiny stejných neuronů s různými délkami odezvy na podnět.

Následuje popis metod řešení. Jakým způsobem se nám podařilo vypracovat

vat řešení soutěže vypsané v [1] následované našimi vylepšeními tohoto řešení. Jednotlivé podseky popisují vlastnosti paralelního propojení neuronů vedoucí k synchronizaci, způsob algoritmicke provázání neuronů odpovídajících učenému vzoru, dále rozbor schopnosti doučit síť další předlohu již známého vzoru. Popisujeme také metodu, kterou lze dosáhnout kýžených vah paralelní množiny algoritmickeým způsobem. Rovněž zde lze nalézt popis práce na zjištění přibližné kapacity sítě (související s vlastnostmi paralelního propojení).

Další je kapitola o technické implementaci. Implementace byla provedena v programovacím jazyce C++ s důrazem na použití objektů pro modelování neuronu a synapse. Rovněž v této části práce je popsán prostor vstupů sítě. Rozebíráme zde digitalizaci zvuku a způsoby jeho zpracování pro účely informatiky, počítačového zpracování zvuku a především metody zpracování řeči. Rovněž popisujeme moderní metody, pokoušející se popsat zvuk modely blízkými neuronální reprezentaci.

Jako poslední lze nalézt diskusi dosažených výsledků a dalších možností výzkumu našeho tématu. Práce je uzavřena seznamy obrázků a literatury.

Autor by zde rovněž rád poděkovat všem, kteří ho při vypracovávání diplomové práce podporovali. Největší dík pak patří vedoucímu, doc. Petru Maršálkovi, za jeho skvělé vedení, množství výborných a věcných připomínek a odpovědi na množství otázek.



## 3 Neuron

### 3.1 Biologický neuron

Lidský mozek se skládá z nervových buněk - neuronů. Těch je v mozku obrovské množství - až  $10^{11}$ . Neurony spolu komunikují. Mezi nimi nalezneme řádově  $10^{14}$  spojů - tzv. synapsí. Jednotlivé neurony jsou spolu těmito synapsemi pospojovány do vyšších logických celků, ve kterých pracují na svých specifických úkolech. Takovými vyššími celky jsou: centrum sluchu, centrum zraku a podobně. Neuron jakožto lidská buňka samozřejmě obsahuje jádro s DNA v chromosomech, mitochondrie, ribosomy, prostě všechno, co má eukaryotická buňka obsahovat. Dále vykazuje většina nervových buněk obdobný tvar. Typický neuron lze rozdělit na tři základní části (viz obr. 1). 1. Vstupní výběžky - **dendrity**, 2. oblast, ve které se nachází většina organel a jádro - **tělo**, neboli soma a 3. konečně dlouhý výstupní výběžek - **axon**. Postupně si tyto části popíšeme podrobněji.

#### 3.1.1 Dendrity

Dendrity jsou výběžky neuronu určené k příjmu signálů. Na jejich koncích jsou synaptické receptory, které dokáží přijímat množství chemických látek - neurotransmiterů (neuropřenašečů). Ty mohou být aktivační (vybuzují neuron k činnosti), nebo inhibiční (jeho činnost naopak utlumují). Na základě zachytávání těchto neurotransmiterů receptory na svém konci pak každý dendrit pošle malou změnou proudu signál tělu neuronu. Na jednom neuronu se nachází až tisíce dendritů přijímajících signál. Každý neuron tedy může získávat vstupní informace až z tisíců jiných jednotek.

### 3.1.2 Soma

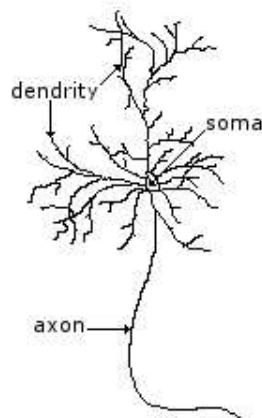
Soma - tělo neuronu, ve kterém se nachází jádro neuronu, a do kterého přichází všechny dendrity a vychází z něj axon. Integruje se v něm napětí ze všech dendritů. Pokud toto napětí překročí určitou mezní hodnotu (práh), aktivuje neuron a vyšle akční potenciál axonem. Probíhá v něm také syntéza proteinů a řízení procesu narůstání spojů s jinými neurony - vytváření, udržování a změna těchto spojů. Tyto spoje jsou základem vyšších logických obvodů.

### 3.1.3 Axon

Axon je jeden speciální výběžek neuronu, kterým z něj odchází nervový vzruch. Bývá zpravidla mnohonásobně delší než tělo neuronu. Po většinu délky je pokryt myelinovými pochvami, které zlepšují a zrychlují vedení vzruchu na konec axonu. V mezerách mezi nimi jsou situovány iontové kánály, které umožňují šíření akčního potenciálu. Z každého axonu vybíhá spousta presynaptických výběžků předávajících signál dalším neuronům. Na nich se nachází speciální struktury, které vypouští neurotransmitery a komunikují tak s navazujícími neurony. Každý axon tak oslovuje pomocí transmiterů dendrity jiných neuronů.

### 3.1.4 Akční potenciály

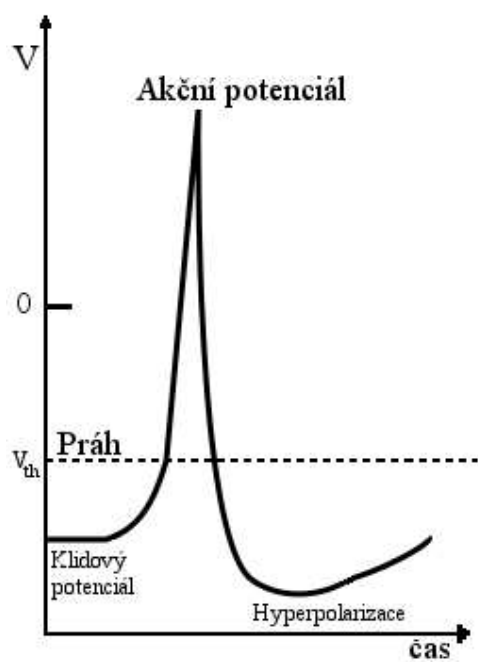
Mezi neurony se šíří informace pomocí vzruchů. Vzruchy jsou dvojího druhu: elektrické - uvnitř neuronu a chemické - ve štěrbinách mezi neurony, v synapsích. Elektrický děj probíhá uvnitř nervové buňky. Je založen na rozdílu koncentrace iontů uvnitř a vně buněčné membrány. Uvnitř neuronu je mnohem vyšší koncentrace draselných kationtů -  $K^+$  než ve vnějším prostředí a naopak v mezibuněčném prostoru se vyskytuje mnohem více sodných kationtů  $Na^+$  než v buňce. Povrch buňky je pokryt velkým množstvím iontových kanálů, které umožňují buňce udržovat nerovnovážený stav mezi ní a vnějším - mezibuněčným - prostředím.



Obrázek 1: **Biologický neuron.** Množství dendritů přivádí elektrické signály do těla neuronu (soma), které je integruje a při přesáhnutí prahu vysílá akční potenciál axonem na synapse, které vedou vzruchy na postsynaptické (naslouchající) neurony.

Jiné kanály jsou potřeba právě při akčním potenciálu (přenosu informace elektrickým proudem), neboť jejich otevřením se prudce změní koncentrace iontů, a tedy i napětí v buňce - odtud anglický termín *spike* používaný pro akční potenciál (viz obr. 2). Pokud takto vznikne postupující vlna, udržovaná výše zmíněnými myelinovými pochvami, přenesení se změna napětí až na konec dlouhého axonu. Po odeznění akčního potenciálu se buňka dostane naopak pod své klidové hodnoty napětí, což otevře opět další iontové kanály, které ji uvedou zpět do klidového, ale nerovnovážného stavu.

Chemický děj probíhá v synapsích (mezi jednotlivými neurony) pomocí látek zvaných neurotransmitery. Když akční potenciál dosáhne konce axonu - přechodu na synapsi - uvolní se do synaptické štěrbiny určité množství neurotransmiteru. Jeho molekuly se na druhém konci naváží na receptory dendritu a modifikují prostřednictvím otevření iontových kanálů napětí na postsynaptickém neuronu. Již také bylo zmíněno, že nervové přenašeče mohou být dvou typů. Excitační - vyvolávající postsynaptický neuron a inhibiční - utlumující činnost postsynaptického



Obrázek 2: **Akční potenciál.** Na obrázku je schématicky znázorněn průběh napětí na neuronu, který překročí práh, vyše akční potenciál a projde refrakční periodou. Zřejmá špička stojí za anlickým pojmem spike.

neuronu. Příkladem chemických látek, které fungují jako neuropřenašeče, jsou kyselina gamaaminomáselná, která je inhibiční; a dopamin, serotonin či acetylcholin, které jsou excitační a nebo inhibiční, podle typu receptoru.

## 3.2 Umělý neuron

Jeden z původních matematických modelů neuronu vznikl v roce 1943, viz McCulloch a Pitts [9]. Jedná se o zajímavé zjednodušení konceptu biologické nervové buňky. Jednotka má  $n$  vstupů  $x_i$ , odpovídajících dendritům. Každý ze vstupů je násobený konstantou  $w_i$ , reprezentující váhu příslušného dendritu daného neuronu (některé vstupy jsou významnější než jiné). Všechny tyto vstupy sečteme do jednoho stimulu

$$\xi = \sum_{i=1}^n w_i x_i. \quad (1)$$

Tento stimulus se nyní porovná s prahem. Pokud je získaný stimulus dostatečně vysoký, neuron dá na výstupu informaci. Typicky je výstup definován aplikací přenosové funkce na rozdíl stimulu a prahu:

$$y = f(\xi - \theta). \quad (2)$$

Tento umělý neuron zvaný perceptron používal pouze binární výstupní funkci jako výsledek porovnání stimulu s prahem. Tento model funguje jako lineární separátor vstupů - je schopen rozdělit prostor vzorů dělicí nadrovinou na 2 podprostory.

Později se repertoár přenosových funkcí značně rozšířil od lineární ( $a \cdot (\xi - \theta)$ ), přes sigmoidu ( $-\frac{a}{1-e^{-(\xi-\theta)}}$ ) k jiným, ještě exotičtějším funkcím.

Logickým krokem od modelu jednoho neuronu bylo spojení více takových jednotek do umělé neuronové sítě. Tento krok rovněž znamenal nový boom v oblasti umělých neuronů, neboť překonal neschopnost jediného perceptronu modelovat lineárně neseparabilní funkce.

Pro síť jednoduchých matematických neuronů - MLP, tj. *multi-layer perceptron* - existuje standardní učicí algoritmus, zvaný *backpropagation* - zpětná propagace chyby. Pomocí tohoto algoritmu jsme schopni správně sestavenou síť naučit dvojice [vzor, výsledek]. Relativní jednoduchost tohoto algoritmu rovněž přispěla k popularitě umělých neuronových sítí v osmdesátých a devadesátých letech minulého století. I dnes mají takovéto neuronové sítě mnoho aplikací - modelování funkcí, klasifikace, predikce, hledání příznaků a mnoho dalších obecných

konceptů v průmyslu, ekonomice, zdravotnictví a samozřejmě matematice a informatice. Autor se podílel na použití MLP pro kontrolu robota popsanou v [17].

V současné době existuje velké množství jak modelů neuronů (perceptron, jednotka používající RBF, radiální bázičké funkce, spiking neuron) tak hlavně modelů sítí - mnohvrstevný perceptron, Kohonenova mapa, Hopfieldův model, všemožné hybridní modely, a podobně.

### 3.3 Spiking neuron

Spiking neuron je matematický model blízký biologickému neuronu. Od perceptronu se liší hlavně tím, že má pro každý dendrit - vstup - definované časové zpoždění, s jakým k němu daný signál dojde. Tato změna umožňuje tomuto modelu získat řádově mocnější výpočetní vlastnosti (více viz [15]). Model v této práci popisujeme pomocí biologických termínů. Vstupní informace označujeme jako proud, případně náboj, vlastnostmi neuronů a synapsí jsou odpor a nejrůznější časové konstanty odpovídající iontovým kanálům (viz biologická sekce 3.1).

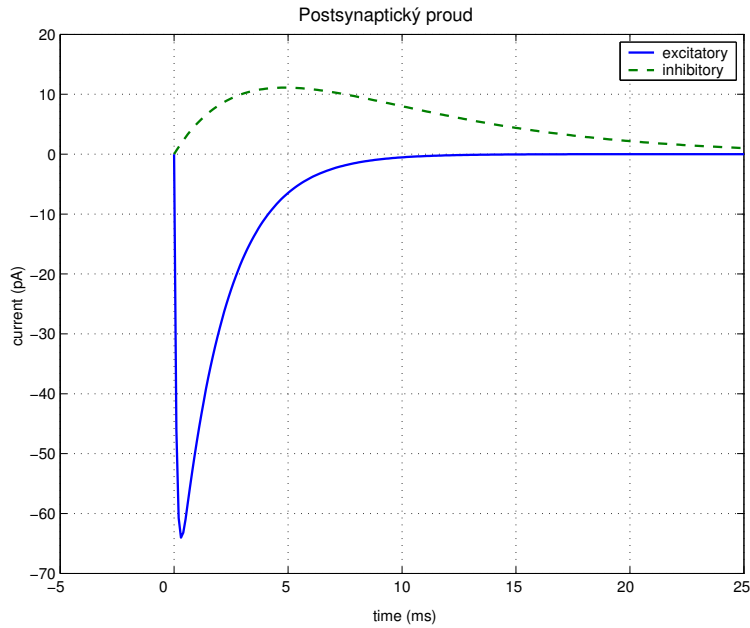
Pro každý neuron  $i$  počítáme vstupní proud  $I_i$  jako vážený součet proudů přes všechny vstupní synapse (každá synapse má presynaptický neuron  $j$ ). Odezvu synapse na akční potenciál  $f$  neuronu  $j$  z času  $t_j^{(f)}$  modelujeme pomocí tzv. **alfa funkce** (3).

Ta počítá s následujícími veličinami:  $q$  je náboj, který je do synapse vnesen z presynaptického neuronu.  $\tau_s$  je časová konstanta poklesu, ta modifikuje, za jak dlouho přestane synapse na AP reagovat.  $\tau_r$  je časová konstanta vzrůstu, modifikující za jak dlouho dosáhne funkce nejvyšší hodnoty. Parametr  $\delta$  je časové zpoždění, s jakým dosáhne informace presynaptického neuronu synapsi. A konečně poslední část výrazu obsahuje aktuální napětí  $U$  a tzv. rovnovážný potenciál  $E_{syn}$ . Ten udává, jak se bude postsynaptický proud chovat v závislosti na napětí. U excitační synapse je  $E_{syn}$  větší než práh, u inhibiční je naopak přibližně roven klidové hodnotě. Poslední rozdíl v alfa funkci tedy znamená, že v případě excitační synapse bude stimulace přispívat ke zvyšování napětí na neuronu. Naopak stimulus inhibiční synapse, jejíž postsynaptický neuron je nad klidovou hodnotou napětí přispěje k poklesu napětí.

$$\alpha(t) = \frac{q}{\tau_s - \tau_r} [e^{-\frac{t-\delta}{\tau_s}} - e^{-\frac{t-\delta}{\tau_r}}] [U_i(t) - E_{syn}], \text{ iff } t - \delta > 0 \quad (3)$$

Tvar alfa funkce na excitační a inhibiční synapsi neuronů použitých v této práci při konstantním napětí (mezi klidovou a prahovou hodnotou) si lze prohléd-





Obrázek 3: **Postsynaptický proud.** Lze vidět postsynaptický proud při konstantním napětí na excitační (modrá plná čára) a inhibiční (zelená přerušovaná čára) synapsi. Příchod AP na excitační synapsi doprovází téměř okamžitá depolarizace. U inhibiční synapse je reakce pomalejší, dochází k hyperpolarizaci (o menší absolutní hodnotu), nicméně stimulace trvá delší dobu.

nout na obrázku 3.

Oproti perceptronu je zde ještě další důležitý rozdíl, a to, že model má teoreticky možnost zohlednit všechny akční potenciály (AP) presynaptických neuronů v minulosti, kdežto perceptron bere jen poslední (aktuální) stav. Vstupní proud neuronu  $i$  pro čas  $t$  lze tedy přehledně shrnout do následující rovnice ( $j$  jsou indexy presynaptických neuronů,  $F_j$  množina aktivací neuronu  $j$ ):

$$I_i(t) = \sum_j w_{ij} \sum_{f \in F_j} \alpha(t - t_j^{(f)}). \quad (4)$$

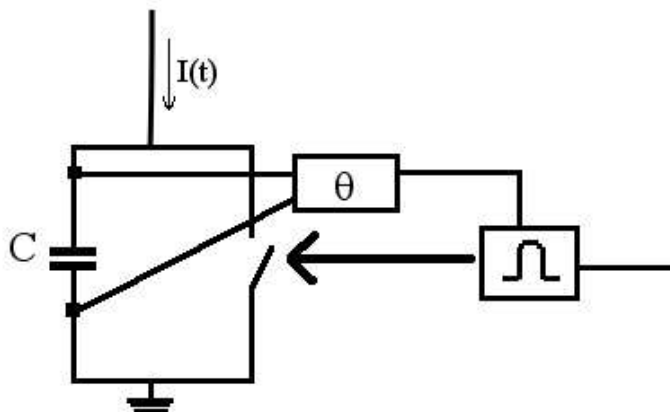
V zadání máme předepsáno použití modelů z práce [3]. V této práci je popsáno několik spiking neuronových modelů. Podíváme se na ně tedy podrobněji.

### 3.3.1 Integrate and fire neuron

Toto je jeden z nejjednodušších modelů spiking neuronu. Jak název napovídá, jednotka provádí pouze dvě základní operace. Přijímá a integruje vstupní signál od presynaptických neuronů, počítá kdy přesáhne práh a v ten moment se aktivuje a vysílá akční potenciál. Tento model neuronu lze znázornit elektrickým obvodem z obr. 4. Obvod se sestává pouze z kondenzátoru (o kapacitě  $C$ ). Ten zajišťuje postupnou kumulaci vstupního náboje. Jeho napětí je porovnáváno s prahovou hodnotou  $V_{th}$ . Jakmile je prahové hodnoty dosaženo, *Integrate and fire* (IAF) jednotka vygeneruje akční potenciál, který předá informaci na ni napojeným neuronům. Po proběhnutí AP projde refrakční fází a postupně zresetuje svůj membránový potenciál na klidovou hodnotu. Pro testování a obzvláště matematické modelování je tento model svou jednoduchostí velmi vhodný a často vyhledávaný. Je ovšem zřejmé, že realismus takového modelu není dostačující a proto se používá spíše model s únikem proudu (leaky), neboli též zapomínající (forgetful), který si popíšeme v následující podkapitole. Tento nejjednodušší model navíc nelze pro řešení článků [1, 2] použít, neboť neurony z těchto článků zřejmě vykazují vlastnosti zapomínajícího modelu.

### 3.3.2 Leaky integrate and fire neuron

Jedná se o velmi podobný neuron jako v předchozím případě. Má ale jednu podstatnou odlišnost, která jej více přibližuje realitě. Jde o ekvivalent postupného unikání nadměrné koncentrace iontů zpět do okolí buňky - proto *leaky* neuron. Pokud tedy delší dobu nedostane žádné podněty, vrací se jeho napětí do klidové hodnoty. Fyzikálně toho lze dosáhnout velmi jednoduše. Do obvodu na obrázku 4 přidáme ke kondenzátoru ještě odpor  $R$ . Takový obvod pak nazýváme  $RC$  obvodem. Veličině  $\tau = RC$  se říká pasivní časová konstanta. Změnu napětí  $dV_m(t)$  takovéto zapomínající jednotky pak vyjadřuje rovnice 5.  $V_m$  v ní označuje aktuální hodnotu membránového napětí,  $\tau$  výše zmíněnou časovou konstantu.



Obrázek 4: **IAF neuron**. Vstupní proud nabíjí kondenzátor  $C$ . Pokud měřené napětí na kondenzátoru přesáhne práh  $\theta$  je generován AP a napětí resetováno na původní hodnotu (znázorněno šipkou).

$I(t)$  je zřejmě hodnota proudu v čase  $t$  a  $C$  pak kapacita LIAF jednotky.

$$\frac{dV_m(t)}{dt} = \frac{-V_m}{\tau} + \frac{I(t)}{C} \quad (5)$$

Pokud napětí  $V_m$  překročí práh  $\theta$ , dojde k aktivaci neuronu a zresetování napětí na hodnotu pod klidovou hladinou.

Tento model používáme v této práci, neboť jeho vlastnosti zcela odpovídají těm požadovaným v zadání naší úlohy.

### 3.3.3 Pyramidová buňka

Modely spiking neuronů ovšem nekončí u takto jednoduchých typů. Obdobně lze modelovat i zjednodušený model pyramidové buňky v lidském mozku. U tohoto typu modelu je samozřejmě mnohem obtížnější zjistit matematickým aparátem jeho vlastnosti, protože obsahuje řádově více parametrů. V případě, že nám stačí simulovat chování bez podrobné znalosti parametrů je ale samozřejmě užitečný. Takto složitý model nicméně pro naši úlohu nepotřebujeme.

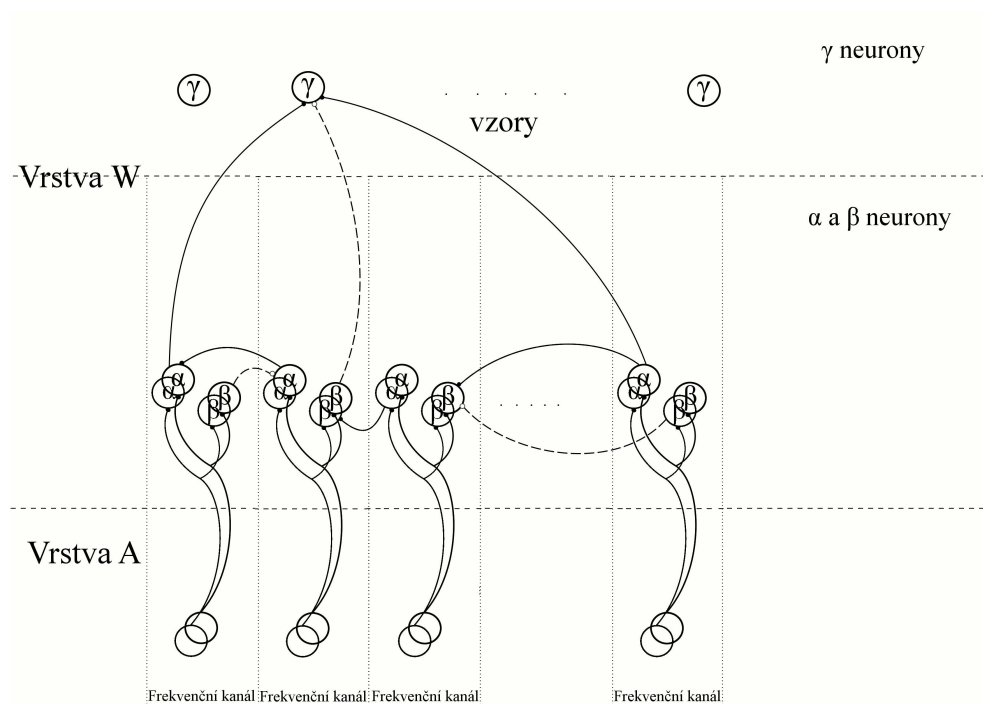
## 4 Původní neuronová síť

Tato práce je postavena na zadání podle článků [1] a [2], J. Hopfielda a C. Brodyho. Ve svých člancích používají síť popsanou v této kapitole pro rozpoznávání jednoslabičných zvuků pomocí synchronizace v této síti. Všechny neurony použité v síti jsou určitým způsobem parametrizované instance LIAF modelu popsaného v kapitole 3.3.2. Jednotlivé detaily parametrů použitých neuronů naleznete v následujících podkapitolách. Ty popisují jednotlivé vrstvy sítě, tak, jak je lze vidět na obrázku 5. Každá vrstva obsahuje specifické neurony a má specifickou funkci.

Nejnižší vrstvou A vsupují do sítě informace o zvukovém signálu. Její neurony jsou jednoduché, jediné spojení s vrstvou A jsou axony ústící do vyšší vrstvy nad ní.

Prostřední vrstva má za úkol synchronizaci neuronů. Obsahuje dvojice excitačních a inhibičních neuronů napojené na vstup. Mezi jednotlivými dvojicemi (odpovídajícími vstupním vzorům) jsou rovněž paralelní spoje, pomocí nichž dochází k synchronizaci.

Ta zaručí aktivaci neuronů poslední vrstvy, který odpovídá skupině propojených neuronů a touto skupinou naučenému zvuku. Funkce nejvyšší vrstvy je tedy čistě výstupní, kde každý neuron odpovídá jednomu známému zvuku.



Obrázek 5: **Schématický vzhled celé sítě.** Za základní rozdělení sítě považujeme vstupní vrstvu A a výpočetní vrstvu W. Vrstva A a spodní část vrstvy W jsou rozděleny na části, odpovídající jednotlivým příznakům vstupního vektoru. Každá z těchto částí má potom neurony, odpovídající na podnět různou délkou aktivování. Vrchní část vrstvy W má neurony odpovídající naučeným vzorům.

## 4.1 Vrstva A

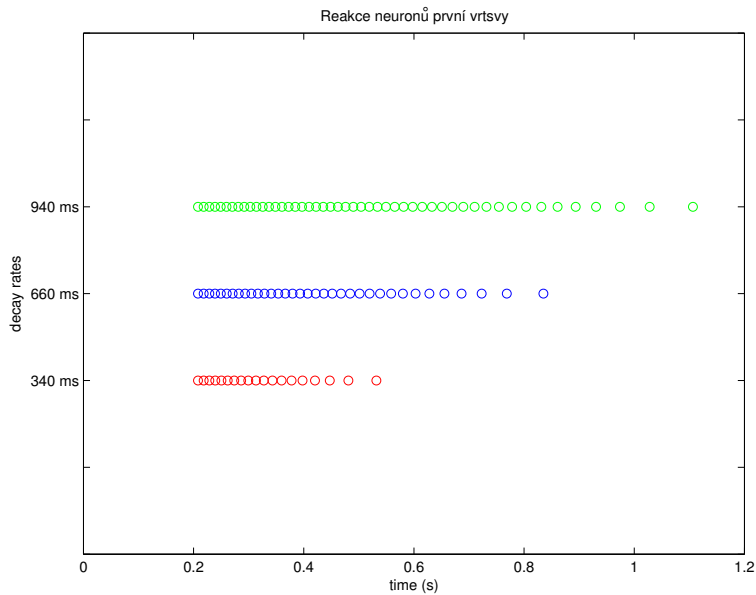
Jako první si popíšeme nejnižší vrstvu sítě, kterou autoři nazvali *layer A*. Vrstva A zastává v naší síti vstupní funkci. Pomocí jejích neuronů přichází do celé sítě informace o vstupním signálu.

V této *vstupní* vrstvě se nachází neurony reagující na 3 typy událostí - náběhy (onsets), ukončení (offsets) a vrcholky (peaks) ve zvukovém signálu. Pro každou z těchto tří událostí existuje několik neuronů, reagujících na čas této události získaný z různých frekvenčních kanálů vstupu (viz sekce o zpracování zvuku 5). To znamená, že existuje více neuronů reagujících na náběhy, ale každý na náběh získaný z jiného frekvenčního kanálu. Obdobně je tomu s vrcholky a ukončeními.

Jako druhý rozměr vstupní vrstvy použijeme délku odezvy neuronu na příznak. Tento rozměr potřebujeme, abychom mohli dosáhnout synchronizace přes různé vstupní kanály. Pro každý z výše popsaných příznaků tedy nebudeme mít jeden neuron, ale větší množství neuronů s různými délkami odezev.

V původní práci [1, 2] je použito 40 příznaků, které jsou získány z 20ti frekvenčních kanálů. Jedná se tedy o neúplný výběr z množiny náběhů, ukončení a vrcholů získaných ze vstupního signálu (těch je logicky  $3 \times 20$ , tj. 60). Tento výběr lze ovšem dle výsledků práce označit za dostačující pro naše účely. Pro každý z těchto 40 příznakových kanálů máme 20 neuronů s různou délkou odezev. Ve vrstvě A se tedy (bez redukce) nachází 800 neuronů, popisujících svými řetězci akčních potenciálů vstupní zvuk.

Na následujícím obrázku 6 lze jasně vidět rozdíl spike trainů pro různé délky odezev. Jedná se o kanál jednoho konkrétního příznaku (neurony se aktivují ve stejnou dobu).



Obrázek 6: **Délky odezev neuronů vrstvy A.** Pro každý příznakový kanál existuje ve vstupní vrstvě množství neuronů s odlišnými odezvami. Zde jsou znázorněny tři z nich, s délkami 340, 660, a 940.

## 4.2 Vrstva W

Vrstvu W lze rozdělit na dvě podvrstvy (znovu viz. 5). Jednu, obsahující neurony typu  $\alpha$  a  $\beta$  (viz níže), která by se dala označit jako výpočetní. A druhou, obsahující neurony  $\gamma$ , která plní výstupní funkci. Verze prezentovaná v článkách [1, 2] tedy obsahuje tolik neuronů, kolik máme naučených vzorů. Ten nejvíce aktivovaný z nich potom odpovídá zvuku, za který vstup považujeme.

V podvrstvě  $\alpha$ - $\beta$  dochází k jevu, díky němuž jsme schopni vstup přiřadit jednomu z naučených vzorů. Jedná se o *synchronizaci* aktivačních frekvencí neuronů přes příznakové kanály. Jev je podmíněn paralelním propojením neuronů mezi sebou uvnitř podvrstvy. To, které neurony propojíme, je dáno časem příznaků konkrétního vzoru - viz sekce 6.

Tabulka 1: Parametry  $\alpha$  neuronu

Parametr	Hodnota	Význam
$C$	$2,5 \times 10^{-10}$ F	kapacita
$V_{th}$	-55 mV	práh
$V_{rest}$	-65 mV	klidová hodnota
$V_{ref}$	-75 mV	napětí po AP
$\tau$	20 ms	membránová časová konstanta
$t_{ref}$	2 ms	délka období refrakce

#### 4.2.1 $\alpha$ neurony

$\alpha$  neurony můžeme nazvat excitační synchronizační jednotky sítě. Dostávají signál ze vstupních neuronů vrstvy A. Každý  $\alpha$  neuron je napojen na právě jeden vstup. To implikuje, že stejně jako neurony ve vrstvě A reagují na 3 typy událostí - náběhy, vrcholy a ukončení - ve zvukovém signálu. Také existují v několika verzích, odlišených dobou odezvy na podnět.

Tyto neurony, však již jsou schopny specifické odpovědi na určitý typ signálu. Tato schopnost je dána vzájemnými propojeními  $\alpha$  a  $\beta$  neuronů. Díky němu se propojené neurony aktivují více, pokud je na vstupu vektor podobný tomu, na který byly naučeny. Pro tyto neurony používáme model LIAF neuronu s diferenciální rovnicí (5) s parametry zapsanými v tabulce 1.

Proč nazýváme jednotky  $\alpha$  excitační? Důvodem je způsob, jakým jednotka  $\alpha$  stimuluje své postsynaptické neurony. Chování post- $\alpha$  synapse na podnět lze tedy shrnout následujícími parametry alfa funkce (3) zapsanými do tabulky 2. Reakce je velmi rychlá (téměř okamžitá) a vybuzení postsynaptického neuronu trvá jen chvíli. Reverzní potenciál nad hodnotou prahu neuronu zajišťuje, že přicházející podnět přispívá ke zvětšování napětí až k prahové hranici.



Tabulka 2: Parametry  $\alpha$  synapse

Parametr	Hodnota	Význam
$\tau_r$	0.2 ms	konstanta růstu
$\tau_s$	2 ms	konstanta poklesu
$E_{syn}$	-25 mV	rovnovážný potenciál
$q$	$1.5 \times 10^{-15}$ C	náboj

#### 4.2.2 $\beta$ neurony

$\beta$  neurony lze obdobně označit za inhibiční synchronizační jednotky. Tyto jednotky jsou velmi podobné neuronům  $\alpha$ . Dostávají rovněž signál, odpovídající příslušným příznakům vstupního vektoru, z vrstvy A. Stejně je též to, že tyto neurony jsou duplikovány přes rozměr délky odezev. Je jich pochopitelně stejný počet, jako  $\alpha$  neuronů. Parametry neuronů jsou totožné s tabulkou 1.

Liší se tím, že jejich synapse posílají inhibiční signál do napojených neuronů. Tyto synapse lze popsat parametry v tabulce 3. Odlišnosti jsou viditelné na první pohled. Mnohem větší konstanta růstu podmiňuje delší působení na postsynaptické neurony. Rovněž stimulace po dosažení maxima funkce bude trvat déle než v případě excitačních jednotek. Reversní potenciál je nastaven na klidovou hodnotu napětí. To s sebou přináší následující chování: pokud je jednotka v klidu, inhibiční signál ze synapse ji neovlivní. Jestli je však již nad klidovou hodnotou, AP presynaptické inhibiční jednotky přispěje ke snížení hladiny aktivity. V případě, že jsme v období refrakce, inhibiční jednotka pomáhá vrátit se na klidovou hodnotu napětí.

#### 4.2.3 $\gamma$ neurony

$\gamma$  neurony lze označit za výstupní integrační jednotky. Dostávají signál z  $\alpha$  a  $\beta$  neuronů. Každá je spojena se skupinou  $\alpha$  a  $\beta$ , propojených navzájem paralelními spoji. Pokud dojde v této skupině neuronů k synchronizaci, je  $\gamma$  neuron nastaven

Tabulka 3: Parametry  $\beta$  synapse

Parametr	Hodnota	Význam
$\tau_r$	6 ms	konstanta růstu
$\tau_s$	4 ms	konstanta poklesu
$E_{syn}$	-65 mV	rovnovážný potenciál
$q$	$1.5 \times 10^{-15}$ C	náboj

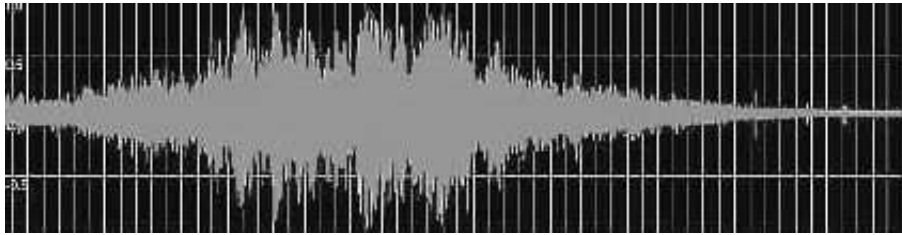
tak, aby jeho stimulace při této synchronizované aktivitě přesáhla práh.

## 5 Prostor vstupů - přirozená řeč

### 5.1 Lidská řeč

Řeč může být popsána jako akt vytváření zvuku pomocí hlasivek a hlasového aparátu, kterým chceme vyjádřit/předat informaci. Za řeč může být také považována znaková řeč a řeč, produkováná pouze v duchu. My máme nicméně na mysli fyzický proces vytváření zvuku pomocí lidských hlasivek. Parametry lidského hlasu se pochopitelně během evoluce vyvinuly do podoby, která koresponduje s parametry lidského sluchu. Omezení sluchového aparátu tedy značným způsobem ovlivnila parametry vydávaného hlasu. Hlas je vytvářen prouděním vzduchu přes hlasivky a jeho rezonancí v ústní a nosní dutině. Dokážeme ovlivňovat hlasitost zvuku silou proudu vzduchu, tón pomocí tvaru hlasové štěrbiny a barvu pomocí tvaru a velikosti rezonančních dutin. Specifické vlastnosti lze samozřejmě dobarvovat pomocí rtů, zubů a v neposlední řadě jazykem.

Rozsah frekvence řeči je (včetně vysokých tónů) asi 40 - 2000 Hz. Rozsah hlasu při řeči je asi jedna oktáva, při zpěvu dvě oktávy. U trénovaných zpěváků to mohou být až tři oktávy. Muži a ženy mají odlišnou stavbu hlasového ústrojí. To má za následek odlišné vlastnosti mužského a ženského hlasu. Rozdíly existují i mezi jedinci stejného pohlaví. Např. často dělíme lidské hlasy na bas, tenor, alt a soprán, podle jejich posazení kolem určité frekvence. Přirozená řeč je velmi ovlivněna naší lingvistickou zkušeností. Vnímání řeči u dospělých jedinců může být velmi odlišné. Specializace mozku na specifický mateřský jazyk přitom začíná být znatelná již v prvních měsících života.



Obrázek 7: **Oscilogram** Ukázka toho, jak vypadá záznam zvuku v oscilogramu.

## 5.2 Zvuk v digitální podobě

Zvuk je vlastně řada změn tlaku v médiu mezi zdrojem a příjemcem. Známou reprezentací zvuku je oscilogram, reprezentující zmenšování a zvětšování tlaku v průběhu času. Protože pracujeme s počítači, je potřeba zvuk zpracovat do digitální podoby. To se děje pomocí digitalizace. Používá se k ní zařízení zvané analogově/ digitální (A/ D) převodník, které převede analogový vstup do digitální (diskrétní) podoby.

Vstupní signál je spojitý v čase, proto je potřeba definovat vzorkovací frekvenci, která bude použita při převedení na digitální signál. Zde je potřeba mít na paměti, že vzorkování omezuje nejvyšší možnou frekvenci, kterou budeme schopni zakódovat (vzorkování musí být dvakrát větší než nejvyšší frekvence, kterou chceme zachytit). Další operací je tzv. kvantizace. Zde jde o použití malého počtu diskrétních symbolů/hodnot pro reprezentaci spojitě množiny hodnot zvukového signálu. Používá se 8 až 24 bitové kvantizace. Kde většinou je použita 8mi (256 hodnot) - nejčastěji v telefoních aplikacích. 16ti (65536 hodnot) bitová kvantizace se používá např. u CD audio záznamu. 24 bitová se používá při nahrávání zvuků v hudebních studiích. Výše napsané popisuje převládající metodu záznamu zvuku - tzv. pulzně kódovou modulaci (PCM), která pravidelně odečítá hodnotu z A/D převodníku a zaznamenává ji v binární podobě.

### 5.3 Zpracování přirozené řeči

Kvůli omezením hlasového ústrojí, které se může pohybovat pouze omezenou rychlostí, se v řeči vyskytují úseky mající stejný charakter. Řečový signál tedy běžně zpracováváme po krátkých úsecích konstantní délky (přibližně 10 - 30 milisekund). Na každý takový úsek se ještě aplikuje filtr - zpravidla tzv. Hammingovo okno. Každá část je pak samostatně zpracována.

Vnímání lidského ucha není rozmístěno lineárně podél frekvenční osy. Naopak většina řeči je situována do pravé části intervalu 0-4 KHz a to tak, že nejvíce je sluch zaměřen na nižší frekvence kolem 200 Hz a postupně se stává méně specializovaným. Tomu dobře odpovídá logaritmické rozdělení frekvence a proto se dnes pro záznam hlasu téměř bezvýhradně používá. Nejčastěji je to melovské rozložení. Každý krátký úsek signálu je potřeba popsat příznakovým vektorem. Nejprve je přenesen do frekvenční domény pomocí Fourierovy transformace. V současné době je zcela nejpoužívanější reprezentace pomocí tzv. Melovských frekvenčních keprstrálních koeficientů - MFCC. Jinou dnes používanou metodou je výpočet koeficientů perceptivní lineární prediktivní analýzy - PLP.

## 5.4 Kortikální reprezentace zvuku

Rozšiřování vědomostí v psychoakustické a neurofysiologické oblasti již delší dobu ovlivňuje podobu algoritmů v oblasti analýzy a rozpoznávání zvukového signálu. Nedávné hlubší porozumění mozkovým oblastem ve vyšších etapách sluchové dráhy umožnilo nový náhled na metodu reprezentace.

Běžné metody postavené na Fourierově transformaci nemají schopnost jednoduše reprezentovat signál ve všech požadovaných rozměrech. Opomínají kvality jako intenzita, tón (zabarvení) a výška hlasu.

Práce především skupiny kolem S. Shamma [5, 6] nabízejí odlišnou metodu reprezentace zvuku - nazývají ji kortikální, kvůli inspiraci biologickou skutečností. Jejich model je schopen obsahovat informace v rozměrech tónu a výšky a dokonce manipulovat signálem pouze podél jedné z těchto "os". Model má dvě fáze. První - časná sluchová fáze - provádí transformaci zvuku na *sluchový spektragram*. Jedná se o převod pomocí metod blízkých biologickému zpracování z jednorozměrné časové domény do dvojrozměrného prostoru frekvence  $\times$  čas, rovnoměrně podél logaritmičticky kvantované osy frekvence. Druhá - kortikální fáze (napodobující vyšší sluchová centra) - provádí analýzu vstupu, používá banku filtrů různých časových, spektrálních a frekvenčních parametrů. Model vypadá velmi zajímavě, zejména pro oblast výzkumu chování mozku se zdá lepším kandidátem než klasické metody postavené na FFT (rychlé Fourierově transformaci). Autoři uvádějí dobré výsledky měření kvality po zpětné konverzi a rovněž úspěšná použití této metody.

## 6 Řešení

### 6.1 Postup hledání parametrů řešení

V této sekci popíšeme, jakým způsobem jsme hledali řešení úlohy z článků [1, 2] za použití LIAF modelu z [3].

Vůbec první krokem bylo vytvořit implementaci LIAF neuronu (reprezentovaná třídou `CLIAFNeuron`), která by odpovídala zadání. Byla použita diferenciální rovnice (5) popsaná prací [3]. Celkovou změnu napětí udává rovnice (4). Parametry byly zvoleny z prací [1, 2] a jsou zapsány v tabulce 1. Bylo potřeba vyzkoušet reakci na konstantní vstupní proud a výsledný spike train porovnat s vlastnostmi modelu v práci [1]. Dále bylo potřeba vytvořit model synapse (třída `CSynapse`), který by odpovídal Hopfieldovým pracem. Uchýlil jsem se k použití tradiční alfa funkce (viz. (3)). Rovnice byly čerpány z práce [4]. Bylo potřeba vytvořit odlišné postsynaptické chování pro synapse z  $\alpha$  a  $\beta$  neuronů a rovněž jinak nastavit synapsi přicházející do neuronů  $\gamma$ . Parametry se nacházejí v tabulkách 2 a 3. Výsledný tvar alfa funkce si lze prohlédnout na obrázku 3. Dalším krokem bylo vytvořit model sítě odpovídající zadání. Tedy vrstvu A (proměnná `layerA`) regulující příslušným způsobem na vstup, na ní napojena dvojice  $\alpha$  a  $\beta$  neuronů (ve vektorech `alfaNeurons` a `betaNeurons`), to všechno ve dvou rozměrech - přes příznaky a přes délku odezvy. Po proměření reakcí vytvořených neuronů na podnět bylo potřeba odstranit případné odchylky chování od toho popsaného v zadání.

Následně byla síť naučena jeden vzor, tzn. bylo třeba vhodně paralelně propojit  $\alpha$  a  $\beta$  neurony a otestovat jejich chování. Toto zahrnovalo rozumnou volbu časů pro délky odezev jednotlivých  $\alpha$  a  $\beta$  dvojic ve stejném vstupním kanále. Na základě těchto odezev pak bylo třeba určit volbu vhodného času, kdy dojde k synchronizaci, a ruční výpočet neuronů, které budou spojeny paralelním propojením. Těmto neuronům se pak vytvořil výstupní  $\gamma$  neuron. Následovalo doladění

vah paralelního spojení, aby vykazovalo příslušnou reakci a  $\gamma$  neuron se při daném vzoru aktivoval. Proces probíhal porovnáváním hodnot aktivace paralelně propojených neuronů pro různé váhy. Jakmile bylo dosaženo vhodné hodnoty, byl laděn práh  $\gamma$  neuronu, tak aby páčil při oné synchronizace. Dalším krokem bylo vytvořit algoritmus propojení  $\alpha$ ,  $\beta$  neuronů odpovídajících vstupnímu zvuku. Toto je posáno v sekci 6.2 a zajištěno funkcí `wirePattern`. Jakmile jsme měli tento algoritmus, mohli jsme přejít k učení a testování většího množství vstupních zvuků. Toho bylo využito při zkoumání vlastností paralelních propojení (sekce 6.3) a prahů výstupních neuronů pro případnou algoritmizaci. Bylo to také důležité pro doplňování vzoru (6.4). Taktéž bylo možno pracovat na zjištění kapacity sítě.



## 6.2 Algoritmus propojení neuronů

V původní práci [1] a [2] propojují autoři neurony "prostřední" vrstvy ručně. Součástí zadání práce bylo nalézt algoritmus pro propojení  $\alpha$  a  $\beta$  neuronů tak, aby docházelo k synchronizaci.

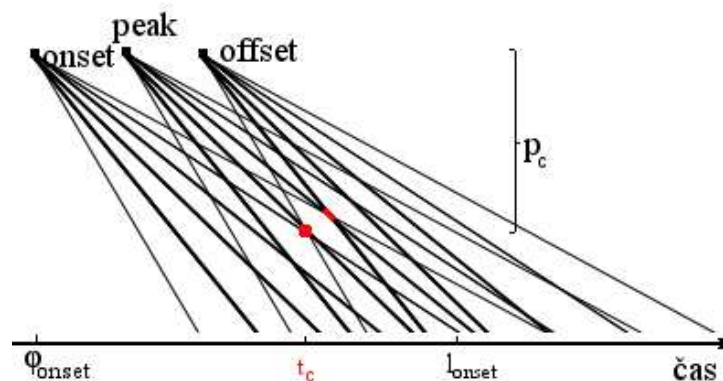
Podstatou úlohy je nalezení vhodných délek odezev v každém frekvenčním kanálu tak, aby existoval čas  $t_c$  (čas konvergence), ve kterém mají všechny příslušné neurony stejnou frekvenci  $f_c$  svých aktivací.

Pokud pro libovolný frekvenční kanál o frekvenci  $f_{req}$  zvolíme délku odezvy  $l_{f_{req}}$  a časový okamžik konvergence  $t_c$ , máme jednoznačně určeno celé řešení (tj. všechny délky odezev) - plyne to z požadavku na shodnost frekvencí v  $t_c$ . Vzhledem k tomu, že klesání frekvence akčních potenciálů u jednotlivých neuronů je lineární, budeme  $t_c$  udávat pomocí počtu procent (od začátku odezvy), kdy dojde ke konvergenci -  $p_c$ . Čas, kdy se na příznakovém kanále  $i$  objeví příznak, označíme  $\varphi_i$ .

Jedná se tedy o optimalizační úlohu. Snažíme se nalézt minimum této účelové funkce:

$$\sum_{i \in \text{features}} \min_{l_i \in L} [(\varphi_{f_{req}} - \varphi_i + l_{f_{req}} p_c) \frac{l_{f_{req}}}{p_c} - l_i]^2 \quad (6)$$

Kde  $l_i$  je nejvhodnější délka odezvy pro kanál  $i$  a  $L$  množina všech odezev systému. Z délek promluv a typického časového rozpětí mezi jednotlivými příznaky vstupního vektoru dostáváme omezení na  $p_c \in (0.4, 0.7)$ . Pro velmi blízká  $p_c$  jsou funkční hodnoty účelové funkce zcela bez tendence - nemonotóní. Po jejím studiu tak zjišťujeme, že bude nejjednodušší empirické řešení rovnice 6.



Obrázek 8: **Zobrazení synchronizace.** Obrázek ukazuje synchronizaci v čase  $t_c$  (v  $p_c\%$  původní hladiny aktivace). Kdy námi sledovaný kanál *onset* začíná vláček aktivací ve  $\varphi_{onset}$  s celkovou délkou aktivací  $l_{onset}$ .

### 6.3 Vlastnosti paralelního propojení

Jedním ze základních kamenů řešení naší úlohy je paralelní propojení neuronů ve vrstvě W přes různé vstupní kanály. Toto propojení umožňuje synchronizaci takto spojených neuronů. Je tedy velmi důležité zmapovat chování tohoto paralelního propojení pro porozumění dynamice a chování modelu. V teoretickém případě je možné propojit všech osmdesát neuronů - 40 vstupních kanálů, jeden aktivační a jeden inhibiční neuron pro každý kanál. Z těchto osmdesáti spojů je samozřejmě teoreticky možné řešení prožezávat směrem k co nejmenšímu možnému dostatečnému množství spojů.

Vyvstává tedy několik otázek. Které parametry budeme měnit, abychom pořád měli funkční síť? Jak velké množství paralelních spojů je ideální? Chceme spíše co nejmenší množství spojů a tudíž, předpokládejme, největší množství možných naučených vzorů? Nebo budeme mít spíše na paměti biologickou plausibilitu? Nebo nás zajímá co nejefektivnější nastavení parametrů?

Rozeberme si to postupně. Budeme vycházet z parametrů daných Hopfieldovými články. Jsou popsány nahoře v sekci o použitém modelu sítě. Jediným

parametrem, kterým budeme měnit chování sítě, bude poměr váhy spojů mezi paralelními neurony a běžnými spoji směrem kupředu (mezi vrstvami).

Je zřejmé, že pokud se váha paralelních spojů bude blížit nule, není možné, aby k jakékoliv synchronizaci vůbec docházelo. Z druhé strany, v případě velmi vysoké váhy těchto paralelních spojů je možné, že se systém dostane do stavu saturovaného synchronizovaného vysílání akčních potenciálů všech spojených neuronů. Příspěvek proudu z paralelních spojů bude prostě tak vysoký, že vybudí ostatní připojené neurony k činnosti. Navíc se takto propojené neurony budou nutit k aktivaci neustále, každé další kolo aktivace, totiž díky vysoké váze opět překoná práh.

Z toho plyne, že je potřeba mít váhu těchto paralelních spojů správně nastavenou, aby bylo možné synchronizace vůbec dosáhnout. Na druhou stranu, nesmí dojít k saturované aktivaci. Tedy musí se jednat pouze o několik sesynchronizovaných vln, kde nadále již samotná energie paralelních spojů nebude nezávisle na přicházejících aktivacích z předchozí vrstvy schopna zajistit další aktivaci.

Podívejme se na případ, že propojíme všechny neurony, které odpovídají danému vzoru, tj. všech osmdesát neuronů s odpovídajícími zpožděními. Zde testy ukazují, že systém přechází přímo ze stavu neschopného synchronizace do opačného extrému - systému se saturovanou periodickou synchronizací. Z těchto testů vyplývá, že každá paralelně provázaná skupina musí být "dostatečně" malá, jinak bude systém nepoužitelný.

V případě zkoumání paralelních skupin o velikosti dvoutřetinové (cca 60 neuronů) a poloviční (40 neuronů) už je systém schopen kýženého mezistavu mezi nesynchronizujícím systémem a saturovanou aktivační smyčkou. Existuje tedy taková váha pro paralelní spoje, že systém se v čase podobnosti vstupních aktivačních frekvencí propojených neuronů "sesynchronizuje" - propojené neurony budou nadále pokračovat v synchronizované aktivaci v přibližně stejnou dobu, kdy v normálním stavu by každý pokračoval svým vlastním tempem.

Ovšem i zde je velikost významným faktorem. Ovlivňuje totiž přípustné hod-

noty parametru pro relativní váhu paralelních spojů. Tendence je taková, že větší systém má menší interval přípustných parametrů vyhovujících zadanému chování. Bohužel, tyto intervaly jsou dosti omezené. Tento fakt negativně ovlivňuje možnost algoritmizace celého řešení a vysvětluje fakt, proč měli autoři celý systém postavený ručně.

Pokud se posuneme na paralelní množinu o čtvrtinové velikosti (tj. kolem dvaceti neuronů - je ovšem třeba mít na paměti, že synchronizace, díky omezenému množství diskrétních časů poklesu, neprobíhá vždy na plném počtu takto propojených neuronů, neboť pro některé z nich prostě ideální časová konstanta poklesu neexistuje) dostáváme se na druhou stranu extrému. V takto malé paralelní množině nedojde k synchronizaci neuronů ani pro velmi velkou váhu horizontálních spojů.

Tento popis vlastností paralelního připojení nám bude velmi užitečný v praktických záležitostech sítě - doplňování vzorů, automatickém učení vah a pochopitelně jsou na něm (i když implicitně) postaveny vlastnosti kapacity sítě.

## 6.4 Doplnování vzorů

Dalším z vytyčených cílů bylo dosáhnout možnosti doplňování cílových zvuků o další vzory.

Toho se dá dosáhnout více možnými cestami. Jednou z možností je spojit každý jednotlivý vzor do vlastní propojené množiny neuronů. Každá z těchto množin by pak stimulovala cílový  $\gamma$  neuron. Druhou variantou je připojit synchronizované neurony z nového vzoru do původní množiny. A přidat několik spojů do  $\gamma$  neuronu. Další způsob je vytvoření více  $\gamma$  neuronů pro jeden vzor. Pro vzor k doučení modifikujeme tu paralelně propojenou množinu, která je nejbliž doučovanému vzoru.

Jak ovšem plyne ze sekce o vlastnostech paralelního připojení, záležitost není tak jednoduchá, jak se zdá. Interval váhy paralelních spojů je totiž většinou velmi úzký a závislý na počtu neuronů v horizontální množině. Druhou věcí je fakt, že  $\gamma$  neuron je také naladěn na určitý počet presynaptických neuronů a v případě, že bude na něj napojeno příliš mnoho nezávislých množin je možné, že se bude aktivovat i bez významné synchronizace. Na druhou stranu, vytvářet nový  $\gamma$  neuron pro úplně každý nový vzor se také nezdá nejvhodnější. Minimálně je to plýtvání z hlediska výpočetního. Jako rozumné se tedy jeví následující hybridní řešení. Pokud máme ve výše popsaném algoritmem získané množině zpoždění neuronů velké množství takových, které se již kryjí s existující množinou odpovídající danému vzoru, připojíme zbylé neurony (musí jich být ovšem dostatečně málo - budeme značit  $\Delta$ , aby nebyly poškozeny příslušné rovnováhy konstant) do této horizontální skupiny. V případě, že žádná taková dostatečně podobná skupina neexistuje vytvoříme nová propojení a k nim nový  $\gamma$  neuron.

Je samozřejmě otázkou experimentů, kde nastavit hranici  $\Delta$  pro přidávání nového neuronu. Jako snad rozumná hodnota se zdá být něco kolem pětiny velikosti původního paralelního propojení  $P$ , tj.  $\Delta \approx \frac{|neuron \in P|}{5}$ . Po zvážení všech eventualit jsme se rozhodli pro vytváření nového  $\gamma$  neuronu pro každou množinu

vzorů, která není až na  $\Delta \in (0, 5)$  neuronů obdobná již existující množině. Důležitý parametr vytvářených množin vychází z vhodné synchronizace paralelních množin menších než úplný graf přes všechny kanály. Díky tomu vytváříme paralelní množiny o menší velikosti - okolo 25 dvojic  $\alpha$  a  $\beta$  neuronů. To nám umožní vybrat ty z neuronů (které dosahují synchronizace) nalezených algoritmem z oddílu 6.2, které jsou dosud zapojeny do nejmenšího počtu paralelních množin. Za cenu toho, že si u každého neuronu  $i$  vrstvy  $W$  držíme počet paralelních množin  $\pi_i$ , ve kterých je zapojen tedy získáváme velké množství výhodných vlastností. Je to větší kapacita sítě, než při náhodném vybírání neuronů a rovněž než při propojování všech osmdesáti neuronů úplným grafem spojů. Rovněž to s sebou přináší menší dopad na váhu již existujících paralelních spojů.

## 6.5 Nastavování paralelních vah

Dalším z potenciálních cílů práce bylo vybudovat automatický systém pro nastavení paralelních vah pro jednotlivé propojené skupiny neuronů odpovídajících naučeným vzorům.

Zde vycházíme ze znalostí popsaných v podkapitole 6.3 Vlastnosti paralelního propojení. Z informací tam uvedených vyplývá existence intervalu paralelních vah pro skupinu každého vzoru, ve které dochází ke kýžené synchronizaci. Naším cílem bude tedy dostat paralelní váhy do příslušných intervalů. Nejprve získáme množství aktivace neuronů v síti bez paralelních spojů. Tato informace bude důležitá pro nalezení vhodného množství synchronizace v paralelní množině. Budeme potřebovat o jisté procento větší aktivaci, než u nezapojené sítě, nicméně ne příliš velkou aktivaci (ta by znamenala zpětnou vazbu aktivací paralelní skupiny a samozřejmě nepoužitelnost pro naše účely).

Metoda získání vah dosahujících synchronizace pro naši paralelní množinu tedy bude v podstatě binární vyhledávání na intervalu *žádná synchronizace* (tj. množství vzruchů ne o více než delta procent větší, než v nezapojeném případě) až saturace, neboli *aktivační smršť* (tedy stav, kdy se systém dostane do neustálého koloběhu akčních potenciálů, které samy dokáží vybudit další vlnu aktivace). Postupně tedy posunujeme vahou do stavu, ve kterém je množina schopna dosáhnout vhodné hladiny synchronizace (má o jisté procento více akčních potenciálů než v nezapojeném případě).

## 6.6 Problém prahů neuronů výstupní vrstvy

V zadání diplomové práce byl zmíněn další problém originálních článků. Jednalo se o nutnost mít velmi specificky nastavené prahy na neuronech výstupní vrstvy.

Principiálně se samozřejmě jedná o velmi obtížný úkol. Neurony poslední vrstvy totiž opravdu musí být schopny reagovat na velmi specifickou událost - současné aktivování presynaptických neuronů - proto musí být dostatečně vyladěny. V opačném případě totiž buď nepřesáhne stimulace práh nebo budeme získávat falešné aktivace při příliš nízkém prahu. To jsou požadavky dané modelem. Proto můžeme chtít pouze částečná zlepšení.

Nám se podařilo dosáhnout dílčích úspěchů. Hlavní rozdíl je v tom, že náš model používá pro většinu vzorů novou paralelní množinu s vlastním  $\gamma$  neuronem. To klade mnohem menší nároky na práh tohoto výstupního neuronu a hlavně na změnu neuronu v případě změny (přidání) počtu síti známých vzorů. V původním modelu bylo totiž třeba upravovat hodnotu prahu u výstupního modelu po přidání každého dalšího vzoru, což činilo síť velmi obtížně modifikovatelnou.

Další výhodou máme díky v předchozí sekci popsanému algoritmu, kterým získáme synchronizující paralelní množinu. Díky tomu není potřeba tolik "ladit" chování pomocí prahu  $\gamma$  neuronu, neboť víme, že pod ním navázané neurony **vhodně** synchronizují.



## 6.7 Kapacita sítě

Dalším na seznamu úkolů práce bylo zjištění kapacity námi postavené sítě. To jest určitým způsobem ukázat limit sítě, po kterém už není dále schopna přijímat další vzory. Jako rozumné měřítko se pochopitelně hodí průměrný počet vzorů (označme  $\pi$ ) na neuron prostřední vrstvy (množinu neuronů označme  $N$ ).  $\Pi = \frac{\sum_{i \in N} \pi_i}{|N|}$  ( $\pi_i$  je číslo, které používáme pro výběr paralelizovaných neuronů v algoritmu 6.4).

Pokud se znovu podíváme na výsledky kapitoly vlastnosti paralelního spojení, zjistíme následující hodnoty. Aby paralelní spojení synchronizovalo je potřeba aktivace přibližně patnácti neuronů. Abychom vyvážili délku intervalu pro synchronizující váhy, co nejmenší příspěvek k naplnění kapacity a zároveň dostatečný počet neuronů vzhledem k malému množství diskrétních délek poklesu aktivací, budeme brát velikost vytvářené paralelní skupiny 25 neuronů.

Pokud předpokládáme dostatečnou náhodnost vstupních vzorů, dojde k přílišné synchronizaci v případě, že dvě paralelní množiny se překrývají v patnácti neuronech. Každou novou 25tici bychom museli umístit do sítě tak, aby se nekryla s žádnou již existující v 15 jednotkách. Je zřejmé, že takto lze umístit jen konečné množství množin. Problém je, že vstupy dostatečně náhodné nejsou. Naopak se ve stejných kanálech vyskytují podobné neurony (protože typicky události začínají v podobnou dobu, stejně jako končí ve stejnou dobu). To samozřejmě dále omezuje kapacitu sítě, my se tomuto sice snažíme zabránit pomocí použití nejméně využité podmnožiny, ale časem začnou díky podobnosti paralelní množiny více a více kolidovat.

Obecně lze říci, že čím je síť plnější (naučená více vzorů), tím obtížnější je nastavit parametry tak, aby docházelo k precizním synchronizacím a aktivacím pouze příslušných  $\gamma$  neuronů. To dále stěžuje naši schopnost zkoumání hranic kapacity sítě. Z praktického testování kapacity je zřejmé, že pokud se použití "žhavých" neuronů (používaných v nejvíce vzorech) dostává na číslo  $\pi_i = 3$ ,

stává se vyladění parametrů pro synchronizaci příliš obtížným. Toto číslo lze zvětšit přidáním dalších vyhasínacích frekvencí a jejich zjemněním.

## 7 Technická implementace

V této kapitole bude popsáno technické zpracování práce. Zdrojový kód je napsán v programovacím jazyce C++. Intenzivně je využívána standardní knihovna STL. Jako kompilátor je použit *gcc*. Vývojová platforma byla *WIN32*, nicméně program neobsahuje žádný na platformě závislý kód a měl by být spustitelný všude, kde je *gcc* a STL (a také byl testován i na Linuxu).

Základní cyklus programu začíná inicializací sítě podle vstupních parametrů a následně předložením jednoho vzoru a proběhnutím jednoho životního cyklu sítě. Výstupem jsou pak seznamy aktivací jednotlivých neuronů, seřazených dle zapojení do naučených paralelních množin (aby bylo možné porovnávat hladinu aktivace) a zakončených aktivacemi výstupních neuronů (což udává, za který vzor síť vstup považuje). Popíšeme si podrobněji použité třídy a jejich funkce, stejně jako funkce hlavního programu. Pro úplné informace doporučujeme čtenáři prohlédnutí zdrojového kódu, který by měl být dostatečně srozumitelný a okomentovaný.

### 7.1 Třída CSynapse

Tato třída reprezentuje jednu synapsi či dendrit neuronu. Má následující atributy:

- `mPresynapticNeuron` - je ukazatel na neuron před synapsí, z jehož axonu přijímáme informace
- `mWeight` - je váha synapse, její hodnota udává, jakým poměrem presynaptický neuron ovlivňuje postsynaptický
- `mReversePotential` - udává jakým způsobem ovlivní aktuální napětí výstup synapse. Určuje hlavní rozdíl mezi inhibiční a excitační jednotkou.
- `mDecayTime` - časová konstanta poklesu

- mRiseTime - časová konstanta vzrůstu
- mInjectedCharge - náboj, vnesen do synapse

Dále třída obsahuje následující funkce:

- alpha (iSpikeAge) - odpovídá funkci (3)
- CSynapse (iSynapseType) - konstruktor vyplní atributy v závislosti na typu synapse: *EXCITATORY\_SYNAPSE* nebo *INHIBITORY\_SYNAPSE*

## 7.2 Třídy CNeuron a CLIAFNeuron

Třída CNeuron je virtuální, jedná se pouze o návrh obecné třídy reprezentující (informatický) neuron. Jako taková má pouze virtuální konstruktor, vektor časů svých aktivací *mFireTimes* a virtuální funkci *step()*, pro vykonání jednoho kroku simulace.

Třída CLIAFNeuron je potomkem CNeuron implementující leaky integrate and fire spiking neuron (viz sekce s popisem tohoto typu spiking neuronu).

Má následující atributy:

- *mFireTimes* - vektor s časy všech akčních potenciálů neuronu
- *mCurrent* - aktuální proud neuronu
- *mRestVoltage* - klidová hodnota napětí neuronu
- *mVoltage* - aktuální hodnota napětí
- *mCapacitance* - kapacita neuronu
- *mThreshold* - práh neuronu
- *mResetVoltage* - hodnota, na kterou se resetuje napětí po akčním potenciálu
- *mTau* - pasivní časová konstanta
- *mRefractPeriod* - délka refrakční periody
- *mActualRefractPeriodDuration* - říká, jak dlouho už probíhá aktuální refrakční perioda
- *mDendrites* - vektor všech vstupů neuronu

- mPatternParticipation - toto číslo označuje počet paralelních zapojení, ve kterých je neuron obsažen

Dále třída obsahuje následující funkce:

- CLIAFNeuron (iVoltage, iCapacitance, iThreshold, iResetVoltage, iTau, iRefractPeriod) - konstruktor nastavující jednotlivé parametry neuronu
- CLIAFNeuron () - destruktork neuronu
- setInputCurrent (iCurrent ) - manuálně nastaví vstup neuronu v tomto kroce
- processDendrites - provede získání stimulu z dendritů pro tento časový krok
- countVoltageDelta - spočítá změnu napětí neuronu na základě nového stimulu
- step - provede jeden časový krok
- addDendrite (iDendrite) - přidá vstup neuronu
- getVoltage () - vrátí aktuální napětí
- getCurrent () - vrátí aktuální proud
- incPatternParticipation () - zvýší počet zapojení neuronu
- getPatternParticipation () - vrátí aktuální počet zapojení
- inRefractoryPeriod (mActualRefractPeriodDuration) - řekne, zda se neuron nachází v refrakční periodě

### 7.3 Hlavní program

Program simuluje chování sítě po krocích délky `timeStep` definované jako 0.1 ms. Celková délka simulace je dána proměnnou `steps`. V každém kroku se spočítá změna napětí všech neuronů pomocí rovnice (5), jak je již popsáno v sekci 6.1.

Popíšeme si nejdůležitější proměnné a funkce hlavního programu.

- `layerA` - zřejmě obsahuje neurony vrstvy A
- `alfaNeurons` - vektor  $\alpha$  neuronů
- `betaNeurons` - vektor  $\beta$  neuronů
- `gammaNeurons` - vektor  $\gamma$  neuronů, pro každý výstupní zvuk
- `decayTimes` - seznam vyhasínacích časů v každém kanálu příznaku
- `connectedPatterns` - vektor paralelních množin neuronů pro každý výstupní zvuk
- `double steps` - délka simulace
- `wirePattern` - zapojí nový vzor do sítě (jak je popsáno v 6.4)
- `chooseDelays` - vybere pro každý kanál takový neuron, aby pálily v určitý okamžik synchronizovaně
- `countDecays` - spočítá vyhasínací časy pro zadané parametry, popsáno v 6.2
- `countDecayError` - spočítá chybu množiny časů, podle toho jak moc se vzdaluje od ideální aktivující ve stejný čas
- `main` - zde probíhá zpracování parametrů a vlastní simulace

## 8 Porovnání se současnou technologií rozpoznávání řeči

Diplomovou práci s tímto zadáním jsem si vybral také proto, že pracuji v oddělení firmy IBM zabývající se rozpoznáváním řeči. Proto považuji za zajímavé srovnání modelu studovaného v této práci se současnou technologií pro rozpoznávání řeči, na které pracuji.

Je samozřejmě zcela absurdní doufat v konkurenceschopnost s profesionálním řešením. O to se ostatně naše síť ani nepokouší. Naším hlavním cílem je pokus o proniknutí do možných neuronálních mechanismů zpracování řeči. Neklademe si tedy cíle ani v chybovosti, ani v rychlosti řešení. Lze tedy jen obtížně tyto přístupy vůbec porovnávat.

Dnešním zcela převažujícím řešením problému rozpoznávání řeči je statistický přístup a metoda skrytých markovovských modelů. Statistická metoda sice požaduje pro své fungování dostatek trénovacích dat, které musí navíc dostatečně pokrývat celou doménu, nicméně ty lze relativně snadno získat jejich namluvením. Výhodou pak je dostatečná obecnost výsledků, které se vztahují i na nová data.

Cílem statistické metody je pro vektor  $W$  popisující vstupní promluvu vytvořit vektor příznaků  $O$  (tak, jak je popsáno v kapitole 5.3) a k tomu pak nalézt **nejlepší** posloupnost slov  $W^*$ . Pomocí Bayesova vztahu lze toto kritérium upravit do podoby  $\operatorname{argmax}_W p(O|W)p(W)$ , kde  $p(W)$  je apriorní pravděpodobnost posloupnosti  $W$  a  $p(O|W)$  je pravděpodobnost pozorování  $O$  za podmínky, že byla vyslovena posloupnost  $W$ .

Problém lze díky tomu rozdělit na dvě oddělené části (ty mohou být modelovány a trénovány nezávisle). Akustický model a jazykový model. Zde nalézáme výše zmíněné skryté markovovské modely. Pomocí této metody hledáme model nejvhodnější posloupnosti slov, přes výpočet pravděpodobnosti  $p(O|M)$ . Metoda se používá díky velmi efektivním algoritmům pro výpočet kýžené pravděpodobnosti



pomocí dynamického programování. Na rozdíl od statistickým metod jsou metody založené na neuronových sítích (a tedy i naše metoda) výhodné nesrovnatelně menším potřebným množstvím dat, pro vybudování modelu.

## 9 Závěr

### 9.1 Diskuse řešení

Na základě článků [1, 2] jsme zkonstruovali řešení soutěže vypsané v článku [1]. Toto řešení jsme se dále pokoušeli zlepšovat. Podařilo se nám uspět u následujících vytyčených cílů.

Vytvořili jsme algoritmus pro vkládání vzorů (zvuků) do sítě (funkce `wirePattern`). Ten prohledá prostor funkčních hodnot optimalizační rovnice 6 (funkce `chooseDelays`). Následně zvolí nejlepší nalezené řešení (`chooseDelays` pomocí výpočtu chyby `countDecayError`) a vytvoří paralelní propojení  $\alpha$  a  $\beta$  neuronů, jakož i spojení s  $\gamma$  neuronem odpovídajícím tomuto vzoru.

Pro proces doučování existujících vzorů jsme zvolili metody zmnožení  $\gamma$  neuronu pro příslušný vzor. Pro každý vzor tedy nemáme jeden, ale větší množství distinktních neuronů. Pro vzor k doučení vytvoříme nová paralelní propojení společně s novým  $\gamma$  neuronem (rovněž funkce `wirePattern`).

Bohužel jsme příliš často naráželi na malou obecnost použitého řešení. Všechny parametry musí být velmi přesně vyladěny, aby systém vykazoval požadované chování. Zkoumané funkce byli často bez jakéhokoliv globálního trendu, natož monotonie. Tyto vlastnosti jsou potom dost značnou překážkou k úplné algoritmicke řešení. Proces vylepšování původního zadání tedy často zápasil s omezeními modelu. I přes to se podařilo alespoň částečným způsobem dotknout většiny bodů v zadání, a vytvořit množství vylepšení a algoritmů.

## 9.2 Možná zlepšení

Hlavním směrem pro možná rozšíření je délka zvuku, který je síť schopna rozpoznat. V současné době se z každého frekvenčního kanálu bere pouze *první* příznak každého typu. To zcela vylučuje možnost rozpoznávat složitější strukturované zvuky delšího trvání. Pro jejich zpracování je tedy potřeba vkládat do sítě informaci o všech příznacích z každého kanálu. Je tedy potřeba použít jinou reprezentaci vstupu a pro takovou situaci připravit vstupní vrstvu.

Ideální vzhledem k zaměření práce, by bylo použití kortikální metody reprezentace zvuku. Viz kapitola 5.4 a literatura [5, 6]. Ta je totiž dle výsledků článku [6] dostatečně mocná a zároveň blízká biologické skutečnosti.

Co se týče kapacity, jako rozumné se ukázalo více zjemnit stupnici vyhasínacích časů. Jiné možné řešení by bylo prostudovat jednotlivé vstupní kanály a nastavit vyhasínací konstanty specificky pro každý kanál. Tím se zabrání tak rychlému přetížení horkých neuronů a dosáhne vyšší potenciální kapacity.

Další pokrok by si zasloužil algoritmus pro výpočet paralelních vah. Lze totiž (pokud má člověk určitou zkušenost s ručním nastavováním) celkem rozumně vytvářet heuristiky podle počtu naučených vzorů, jak velká váha asi bude potřeba. Jak bylo naznačeno v předchozí podsekci, model nijak zvlášť pružný není. Proto se jako nejvhodnější pokračování této myšlenky jeví nesvazovat se designem z původních článků a zkusit navrhnout jinou pružnější metodu řešení problému.

### 9.3 Závěr

V práci jsme dokázali funkčnost modelu synchronizace spiking neuronů za účelem rozpoznávání zvuku. Podařilo se nám úspěšně sestavit řešení práce článků [1, 2], za použití modelu z [3]. Vytvořili jsme algoritmus, který nalezne vhodné kandidáty pro paralelní spojení odpovídající vstupnímu vzoru. Podařilo se nám navrhnout algoritmus přidávající vzory do postavené sítě a vytvořit pro ně nový výstupní neuron. Tento algoritmus rovněž používá nejméně zapojené neurony sítě, takže dobře využívá kapacitu sítě. Prostudovali jsme vlastnosti paralelního spojení a vytvořili metodu, kterou jsme schopni dosáhnout synchronizace  $\alpha$  a  $\beta$  neuronů.

Díky použití separátního  $\gamma$  neuronu pro každý odlišný vstup se nám podařilo snížit citlivost řešení na hodnotě prahu výstupní vrstvy. Rovněž metoda automatického nalezení paralelních vah, pro které dochází k synchronizaci snižuje důležitost tohoto prahu v modelu. Odhadli jsme kapacitu modelu pomocí přílišné aktivity v málo paralelizované síti. Podrobili jsme průzkumu prostor vstupů sítě. Popsali jsme jak současné tradiční řešení oblasti digitalizace řeči, tak moderní přístup založený na podobnosti s lidským mozkem.

Pro budoucí výzkum nám zůstalo použití některé vhodnější vstupní metody (ideálně kortikální popsané v 5.4) a s tím spojená možnost rozpoznání delších zvuků.

Činnost spojená s diplomovou prací byla velmi zajímavá a poučná a jsem velmi rád, že jsme si vybral toto téma. Systém možná zcela nereflexuje svůj biologický předobraz, nicméně vybudování systému, který je rozpoznávání zvuků schopný a přitom biologicky plausibilní znamená zcela určitě dosažení cíle.

K práci je přiloženo CD obsahující zdrojový kód, Makefile, vstupní data a návod k použití CD (readme.html).

## Seznam obrázků

1	Biologický neuron . . . . .	6
2	Akční potenciál . . . . .	7
3	Postsynaptický proud . . . . .	12
4	Obvod reprezentující IAF neuron . . . . .	14
5	Schématický vzhled celé sítě . . . . .	16
6	Délky odezev neuronů vrstvy A . . . . .	18
7	Oscilogram . . . . .	23
8	Zobrazení synchronizace . . . . .	29

## Seznam použité literatury a dalších zdrojů:

- [1] Hopfield J.J., Brody C.D.: "What is a moment? 'Cortical' sensory integration over a brief interval", Proc. Natl. Acad. Sci. USA 97, 13919-24, Dec 2000
- [2] Hopfield J.J., Brody C.D.: "What is a moment? Transient synchrony as a collective mechanism for spatiotemporal integration", Proc. Natl. Acad. Sci. USA 98, 1282-87, 2001
- [3] Maršálek P., Koch C., Maunsell J.: "On the relationship between synaptic input and spike output jitter in individual neurons", Proc. Natl. Acad. Sci. USA 94, 735-740, 1997
- [4] Gerstner W., Kistler W.M.: "Spiking neuron models. Single Neurons, Populations, Plasticity.", Cambridge University Press, New York, 2002
- [5] Zotkin D., Shamma S., Ru P., Duraiswami R., Davis L.: "Pitch and timbre manipulations using cortical representations of sound", conference Acoustics, Speech, and Signal Processing, 2003
- [6] Chi T., Ru P., Shamma S.: "Multiresolution spectrotemporal analysis of complex sounds", J. Acoust. Soc. Am. 118, 887-906, 2005
- [7] Gil-da-Costa R., Martin A., Lopes M., Munoz M., Britz J., Braun A.: "Species-specific calls activate homologs of Broca's and Wernicke's areas in the macaque", Nature Neuroscience 9, 1064-1070, 2006
- [8] Rahn Benjamin: "Speech Recognition Using Neuron Coincidence Detection", solution submitted to [1], 2000, bylo dosažitelné přes

<http://www.inference.phy.cam.ac.uk/mackay/HBMouse.html>, navštíveno 2005

[9] McCulloch W., Pitts W.: "A logical calculus of the ideas immanent in nervous activity", *Bulletin of Mathematical Biophysics* 5, 115-133, 1943

[10] Šíma J., Neruda R.: "Teoretické otázky neuronových sítí", MATFYZPRESS, Praha 1996

[11] Kuhl P., Williams K., Lacerda F., Stevens K., et.al.: "Linguistic Experience Alters Phonetic Perception in Infants by 6 Months of Age", *Science* 255, 606-608, 1991

[12] Beran T., Bergl V., Hampl R., Krbec P., Šedivý J., Tydlitát B., Vopička J.: "Embedded ViaVoice", v: *Text, Speech and Dialogue, Proc. 7th Intl. Conf. TSD 2004*, Sojka P., Kopeček I., Pala K. (Eds.), *Lecture Notes in Artificial Intelligence* vol. 3206, 269-274, Springer Verlag, Berlin, 2004

[13] Vopička J.: "Vyhledávání klíčových elementů v souvislé promluvě", ČVUT Praha, doktorská práce, 2002

[14] Šmídl L.: "Metody rychlé detekce klíčových slov", Fakulta Aplikovaných Věd Plzeň, doktorská práce, 2005

[15] Maas W.: "Lower bounds for the Computational Power of Network of Spiking Neurons", *Neural Computation* 8, 1-40, 1996

[16] Tuckwell, H. C. "Introduction to Theoretical Neurobiology", Cambridge Univ. Press, Cambridge, Vols. 1 and 2., 1988

[17] Winkler Z., Mrázová I., Iša J, Krchák J., "Neural Motion Controller for Robots Daisy and Ester", konference Dextrous Autonomous Robots and Humanoids, 2005

[18] [www.wikipedia.org](http://www.wikipedia.org), sekce o neuronech, rozpoznávání zvuku,  
<http://en.wikipedia.org/wiki/Neuron>,  
[http://en.wikipedia.org/wiki/Digital\\_signal\\_processing](http://en.wikipedia.org/wiki/Digital_signal_processing),  
[http://en.wikipedia.org/wiki/Speech\\_Recognition](http://en.wikipedia.org/wiki/Speech_Recognition) a další,  
navštíveno 2006.