



TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

DIPLOMOVÁ PRÁCE

Parametrická syntéza české řeči

Parametric synthesis of Czech speech

Liberec 2004

UNIVERZITNÍ KNIHOVNA
TECHNICKÉ UNIVERZITY V LIBERCI



3146072467

Stanislav Šilhán

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Katedra elektroniky a zpracování signálů

Akademický rok: 2003/2004

ZADÁNÍ DIPLOMOVÉ PRÁCE

pro: **Stanislava ŠILHÁNA**

studijní program: 2612 M – Elektrotechnika a informatika

obor: 3902T005 – Automatické řízení a inženýrská informatika

Vedoucí katedry Vám ve smyslu zákona o vysokých školách č.111/1998 Sb. určuje tuto diplomovou práci:

Název tématu:

Parametrická syntéza české řeči

Zásady pro vypracování:

1. Seznamte se s problematikou syntézy řeči, především se syntézou pomocí parametrických metod. Zaměřte se na lineárně prediktivního kódování (LPC), metodu PSOLA v frekvenční oblasti a kepstrální syntézu.
2. Prostudujte možnosti jednotlivých metod a zvažte možnosti modelování prosodie.
3. Navrhněte a realizujte syntezátor pro češtinu pracující na principu zvolené parametrické metody.
4. Pro navržený syntezátor vytvořte pravidla fonetické transkripce a prosodie.
5. Vlastní software syntezátoru uzpůsobte pro praktické použití v hlasových informačních dialogových systémech.

Rozsah grafických prací: dle potřeby dokumentace

Rozsah průvodní zprávy: cca 40 až 50 stran

Seznam odborné literatury:

- [1] Psutka J.: Komunikace s počítačem mluvenou řečí. Academia, Praha, 1995
- [2] Nouza J. (editor): Počítačové zpracování řeči. Sborník článků, Liberec, 2001
- [3] Přibil, J.: Pitch-synchronní syntezátor řeči s kepstrálními koeficienty. 1. mezinárodní konference o telekomunikačních technologiách Telekomunikácie 95. Sborník referátů díl – vědecká sekce, Bratislava 31.5. – 1.6. 1995, str. 124–129.
- [4] Huang X., Acero A., Hon H.W.: Spoken Language Processing. Prentice Hall 2001.
- [5] články ze sborníků konferencí Eurospeech a ICSLP

Vedoucí diplomové práce: Ing. Miroslav Holada

Konzultant: Ing. Petr David

Zadání diplomové práce: **23. 10. 2003**

Termín odevzdání diplomové práce: **21. 5. 2004**



Novák

.....
Vedoucí katedry

Psutka

.....
Děkan

V Liberci dne 21.10.2003

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména §60 - školní dílo.

Beru na vědomí, že Technická univerzita v Liberci TUL nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

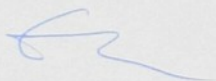
Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu užití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum

19. 5. 2004

Podpis



Poděkování

Rád bych poděkoval celému kolektivu Laboratoře počítačového zpracování řeči při TU Liberec a také panu Ing. Petrovi Horákovi PhD. z Ústavu radiotechniky a elektroniky při AV ČR za jejich pomoc, ochotu a čas věnovaný konzultacím mé diplomové práce, i když to často nebylo jednoduché.

Resumé

Diplomová práce se zabývá parametrickými metodami syntézy lidské řeči a jejich realizací. Hlavním cílem práce je vytvoření jednoduchého TTS systému. V teoretické části jsou jednotlivé metody rozebrány a popsána jejich realizace. Dále se zde zabývám problematikou tvorby TTS systémů. V praktické části jsou popsány následující dosažené výsledky:

Realizace LPC a keprální metody syntézy řečového signálu v programovém prostředí Matlab a porovnání výsledků obou syntezeátorů. Dále vytvoření unikátního inventáře stavebních jednotek pro syntézu řeči. V konečné fázi pak naprogramování jednoduchého TTS systému, pracujícího s LPC syntezeátorem a s možností modelování prozodie.

Summary

The diploma work deals with methods of parametric speech synthesis methods of human speech and with its realization. The main aim of diploma work is to realise simply Text-To-Speech system. In teoretical part are describe principles of basic speech synthesis metods and its realization. Futher deals with problem with Text-To-Speech systems. In practical part are presented these obtained results:

Realization of LPC and cepstral method of speech synthesis in Matlab environment and comparison of both methods. Creation of unique units inventory for speech synthesis. And realization of siple Text-To-Speech system, including LPC synthetizer and posibility of prosody modeling.

Obsah

PROHLÁŠENÍ.....	4
PODĚKOVÁNÍ.....	5
RESUMÉ	6
SUMMARY	6
OBSAH.....	7
1. ÚVOD.....	9
1.1. HISTORIE A SOUČASNÝ STAV PROBLEMATIKY	10
TEORETICKÁ ČÁST.....	12
2. LIDSKÁ ŘEČ.....	13
2.1. LIDSKÁ ŘEČ JAKO KOMUNIKAČNÍ PROSTŘEDEK	13
2.2. TVORBA AKUSTICKÉHO ŘEČOVÉHO SIGNÁLU V LIDSKÉM TĚLE.....	14
3. METODY SYNTÉZY ŘEČI.....	17
3.1. LINEÁRNÍ MODEL ŘEČI	17
3.2. ROZKLAD NA SEGMENTY A PŘEDZPRACOVÁNÍ SIGNÁLU.....	19
3.3. FORMANTOVÝ SYNTEZÁTOR	21
3.3.1. Rezonátory a antirezonátory	21
3.4. LPC MODEL HLASOVÉHO TRAKTU.....	23
3.4.1. Výpočet LPC koeficientů.....	24
3.5. KEPSTRÁLNÍ MODEL HLASOVÉHO TRAKTU	26
3.5.1. Výpočet keprálních parametrů	26
3.5.2. Aproximace exponenciálního rozvoje keprálního modelu.....	28
3.6. SINUSOVÝ MODEL ŘEČI	29
3.6.1. Výběr lokálních maxim amplitudového spektra.....	30
3.6.2. Aproximace obálky amplitudového spektra	30
3.7. SYNTÉZA ZALOŽENÁ NA SPOJOVÁNÍ ŘEČOVÝCH ÚSEKŮ - PSOLA.....	31
3.7.1. PSOLA (Pitch-Synchronous Overlap-Add).....	31

4. TTS SYSTÉMY (TEXT TO SPEECH)	33
4.1. STRUKTURA TTS SYSTÉMŮ	34
4.1.1. Předzpracování	35
4.1.2. Fonetická transkripce	35
4.1.3. Modelování prozodie	36
4.2. INVENTÁŘ STAVEBNÍCH JEDNOTEK	39
4.2.1. Fonémový inventář	39
4.2.2. Dífónový inventář	40
4.2.3. Trífónový inventář	40
4.2.4. Poloslabikové a slabikové inventáře	41
PRAKTICKÁ ČÁST	42
5. REALIZACE SYNTEZÁTORU	43
5.1. LPC VOKODÉR	43
5.2. POROVNÁNÍ RŮZNÝCH DRUHŮ SYNTÉZY V PRAXI	48
5.2.1. Kepstrální syntéza	48
5.2.2. Formantová syntéza	50
6. VÝBĚR STAVEBNÍCH JEDNOTEK	52
6.1. VÝVOJ POUŽITÉHO INVENTÁŘE	52
6.1.1. Praktické vytvoření inventáře	53
6.2. VÝPOČET LPC KOEFICIENTŮ PRO JEDNOTLIVÉ JEDNOTKY	56
6.3. ALGORITMIZACE VÝBĚRU A SKLÁDÁNÍ JEDNOTEK	57
6.4. PROGRAM PRO VÝPOČET PARAMETRŮ JEDNOTEK INVENTÁŘE	58
7. REALIZACE VLASTNÍHO TTS SYSTÉMU	59
8. ZÁVĚR	64
SEZNAM POUŽITÉ LITERATURY	65
SEZNAM PŘÍLOH	67

1. Úvod

V současné době probíhá na naší fakultě v Laboratoři počítačového zpracování řeči vývoj několika velmi zajímavých projektů. Většina těchto projektů se zabývá rozpoznáváním řečového signálu a problémy s tím vznikajícími. Několik stávajících nebo vyvíjených aplikací využívá i syntézu řeči, která je jedním ze stavebních prvků při komunikaci s počítačem mluvenou řečí. Jedním z projektů realizovaných v laboratoři je i dialogový systém InfoCity. Jedná se o informační dialogový systém, se kterým uživatel komunikuje pomocí telefonní linky. Systém rozpoznává vyřčené požadavky volajícího a sděluje mu odpovídající informace. Sdělení je syntetizováno v TTS modulu, který byl vyvinut na pracovišti Ústavu radiotechniky a elektroniky při AV ČR, jelikož v Laboratoři počítačového zpracování řeči se v době vývoje systému nikdo touto problematikou nezabýval. A právě tato absence vedla k myšlence rozšířit výzkum i na vývoj TTS systémů. Proto bylo vypsáno několik témat diplomových prací, zabývajících se problematikou syntézy řeči. Jednou z nich je i má práce. Mým cílem bylo seznámit se tématem parametrické syntézy lidské řeči a po zmapování a zhodnocení současného stavu vybrat jednu z metod syntézy a tuto metodu realizovat. Při realizaci tohoto syntežátoru jsem měl zahrnout možnost modelování prozodie a realizovat fonetickou transkripci. Výsledkem by tedy měl být jednoduchý TTS systém, umožňující základní modelování prozodie. Tato aplikace by měla být realizována tak, aby ji bylo možno použít v dialogových informačních systémech.

1.1. Historie a současný stav problematiky

Pokusy o vytvoření umělé řeči jsou zaznamenány již ve druhé polovině 18. století, kdy Kratzenstein reprodukoval zvuk samohlásek spojením jazýčkové píšťaly s trubicemi různých tvarů a velikostí (1780) a von Kempelen sestavil mechanický syntezátor řeči (1791). Mezi velmi úspěšné mechanické syntezátory můžeme zařadit také Reiszův syntezátor z roce 1937. První elektronické syntezátory, které generují rovněž pouze samohlásky, se objevují až v první polovině dvacátého století. V roce 1939 Dudley v Bell Laboratories představil elektronický syntezátor založený na elektronických rezonátorech. V tom období v Bellově telefonní laboratoři pracovali na efektivním kódování řeči pro přenos telekomunikačními kanály. V principu potřebovali kódovat zdroj signálu za účelem jeho zpětné rekonstrukce. Takto vznikl koncept vokodérů. Výraznější výsledky v této oblasti jsou datovány až v období po II. světové válce (první parametrický syntezátor vytvořil v roce 1948 Walter Lawrence a nazval ho „Pat“) a zejména po r. 1960, kdy byla uveřejněna Fantova práce o modelování hlasového traktu. Jako první se začaly vyvíjet formantové syntezátory. Současně s nimi se pracovalo i na syntezátorech s tabelovanými zvuky. Tyto syntezátory pracovaly na principu spojování namluvených a komprimovaných vzorků řeči. Obě metody jsou v současné době již překonány díky nepříliš velké kvalitě produkované řeči. Největší rozvoj v oblasti syntézy řeči nastal s rozvojem osobních počítačů v 80. a zejména pak v 90. letech. V této době se syntezátory vyvíjely na stejných principech, které se používají dodnes. Jedná se o LPC a keprstrální syntézu, o metodu PSOLA a o sinusovou metodu, pracující ve frekvenční oblasti. V 90. letech se v praxi nejvíce rozšířila metoda PSOLA, protože poskytuje bez velké výpočetní náročnosti relativně vysokou kvalitu syntetické řeči [3]. Neumožňuje však změnu hlasu a změnu prozodie. Pouze na úkor velkého snížení kvality syntetické řeči. V současné době, kdy syntetickou řeč můžeme slyšet každý den v nejrůznějších telekomunikačních či informačních systémech, se kladou stále větší požadavky na její kvalitu a přirozenost. Proto se přistoupilo zpět k metodě LPC a začali se rozvíjet keprstrální a sinusové syntezátory. První český prakticky použitelný systém pro převod psaného textu na mluvenou řeč, pocházející z Ústavu radiotechniky a elektroniky AV ČR, byl vyvinut ve spolupráci s Fonetickým ústavem Filozofické fakulty Karlovy univerzity (FF UK) v letech 1990 až 1993. Tento TTS systém používá metodu LPC. Inventář řečových jednotek

se skládal celkem ze 441 difónů a těl samohlásek vyskytujících se v češtině a ve slovenštině. Ten byl základem pro systém Epos, který je v současné době asi nejznámějším a nejkvalitnějším TTS systémem v naší zemi. Další pracoviště vyvíjející syntézu českého jazyka jsou na Masarykově univerzitě v Brně, kde se na Fakultě informatiky vyvíjí syntezátor Demosthenes založený na použití poloslabik a syntézy v časové oblasti a na Katedře kybernetiky Západočeské univerzity v Plzni, kde se pracuje na syntéze řeči založené na automatické segmentaci jednotek pomocí skrytých Markovových modelů (Hidden Markov Models – HMM). Jako jednotka je použit foném závislý na svém pravém i levém okolí; složitostí se tedy jedná v podstatě o trifonovou syntézu, kde počet řečových jednotek je cca 5000. Na Drážďanské univerzitě je vyvíjena syntéza češtiny ve spolupráci s FÚ Filozofické fakulty. Z komerční sféry lze zmínit např. firmu FROG Systems a její TTS systém CSVoice.

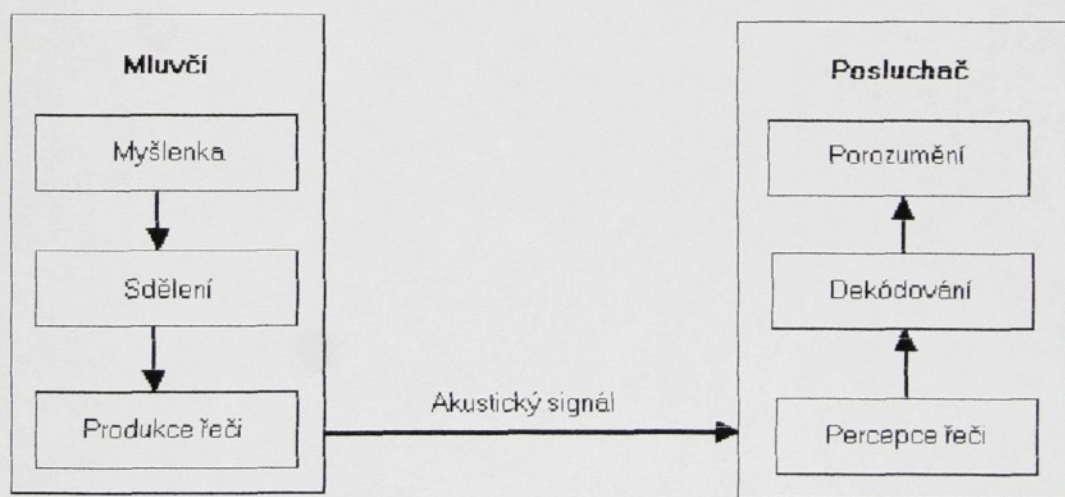
Teoretická část

2. Lidská řeč

"Kolik řečí umíš, tolikrát jsi člověkem."

2.1. Lidská řeč jako komunikační prostředek

Lidská řeč je nejvýznamnějším komunikačním prostředkem, kterým se odlišujeme od ostatních živočichů. Schopnost vyjadřovat se článkovou řečí je vlastní pouze člověku. Řečí jsme schopni sdělit prakticky libovolnou informaci. Je to člověku nejpřirozenější komunikační kanál, od kterého se odvíjí další způsoby komunikace například psaní. Řeč má při komunikaci mnoho výhod. Její tvorba je realizována reflexně v řečovém centru mozku, proto je velmi rychlá. Díky intonaci jsme schopni doplnit informaci o další rozměr a zvýšit tak srozumitelnost projevu, případně vyjádřit postoj mluvčího k tématu (skepse, ironie), vyjádřit jeho emoce (radost, strach). Člověk může obsah promluvy interaktivně měnit dle okolních podnětů (reakce posluchačů). Z řeči lze vyčíst i vlastnosti mluvčího, jako například národnost, naznačuje jeho vzdělání, prostředí ve kterém žije (sociální úroveň), momentální psychický (nervozita) a zdravotní stav (nachlazení).



Obr 2.1 - Proces vytváření a porozumění promluvy

Ucelenou část řeči, která je vyjádřením nějaké myšlenky či sdělení, nazýváme promluva. Proces vytváření a porozumění promluvě lze rozdělit na několik hierarchicky uspořádaných úrovní – viz obr 2.1. Při komunikaci řeči hraje velkou roli vliv prostředí a

situace. Velmi důležitý je kontext. Díky kontextu lze redukovat promluvu o informace, které z kontextu vyplívají. Dochází ovšem i k nedorozumění, kdy mluvčí vypustí z promluvy informaci, kterou posluchač z kontextu nevyvodí. Tento problém v umělé řeči odpadá, protože se zatím s kontextovými informacemi nepracuje.

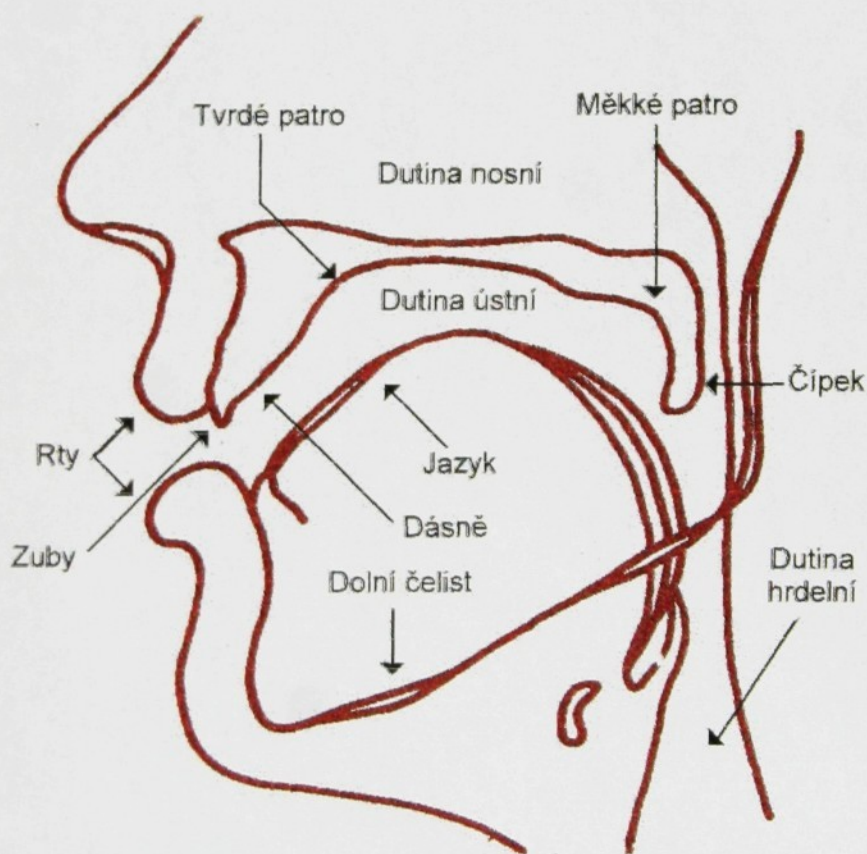
2.2. Tvorba akustického řečového signálu v lidském těle

Na tvorbě akustického řečového signálu se v lidském těle podílí několik orgánů. Ty lze podle jejich funkce při tvorbě řeči rozdělit do tří skupin: ústrojí dýchací, ústrojí hlasové a ústrojí modifikační.

Ústrojí dýchací slouží primárně k základní životní funkci – fyziologickému dýchání, to se při řeči nápadně mění. Při normálním klidovém dýchání je poměr vdechů a výdechů zhruba 2:3 a člověk vdechne 0,5 litru vzduchu, kdežto při řeči se zkracuje vdech a prodlužuje výdech v poměru 1:7 až 1:12 a objem vdechnutého vzduchu se zvětšuje na 1,5 litru. Množství nadechnutého vzduchu a rytmus dýchání se při řeči provádí z části vědomě, zatímco fyziologické dýchání je reflexivní. Mluvní projev se realizuje při výdechu, který lze díky zásobě vzduchu v plicích podle potřeby prodloužit, a který zajišťuje trvalý tlak v hrtanu, což je důležité pro vznik hlasu. Některé speciální hlásky se vyslovují při vdechu, ale to se týká jen určitých cizích jazyků (například africké etnikum).

Ústrojí hlasové je uloženo v hrtanu. Je to typicky lidský orgán a slouží k vytváření základního hlasu, ze kterého po dalších úpravách vzniká řeč. Základem hlasového ústrojí jsou dva svalové valy pokryté sliznicí – hlasivky. Jsou napjaty mezi chrupavkou štítnou a chrupavkami hlasivkovými, které jsou částečně pohyblivé. Jejich pohybem se hlasivky přibližují a oddalují a mění se i jejich napětí. Vytváří se hlasivková štěrbina, která je tvořena blanitou částí hlasivek a okrajem chrupavek pokrytých sliznicí. Otevírání a uzavírání hlasivkové štěrbiny mění proudění vzduchu a vytváří tak základní hlasový signál. Při znělých hláskách dochází k přerušování proudu vzduchu a vzniká tak kvaziperiodický signál, jehož perioda se nazývá základní hlasová perioda. Při neznělých hláskách se v hlasivkové úžině vytváří turbulentní proudění, které je ovlivňováno dle dané hlásky. Vlastnosti lidského hlasu jsou především dány délkou hlasivek. Čím jsou hlasivky kratší, tím rychleji kmitají (otevírají se a uzavírají se) a tím je hlas vyšší.

Ústrojí modifikující je uloženo nad hrtanem a skládá se ze tří dutin: dutiny hrdelní, dutiny nosní a dutiny ústní. Dutina hrdelní se nachází nad hlasivkami a při řeči funguje jako rezonanční dutina s proměnným objemem. Změny objemu se využívá zvláště při tvorbě vokálů (samohlásek). Dutina nosní je také rezonančním prostorem, který se ovšem využívá jen u části hlásek. Tyto hlásky se nazývají nazály. Při tvorbě ostatních hlásek je průchod do dutiny nosní uzavřen pomocí měkkého patra. Dutina ústní se také uplatňuje jako rezonanční prostor a proměnným objemem a to pro všechny hlásky. V dutině ústní jsou uloženy mluvní orgány. Mezi ně patří rty, jazyk a případně měkké patro (ty jsou označovány jako aktivní) a dále zuby, dásně a tvrdé patro (ty jsou označovány jako pasivní).



Obr 2.2 - Průřez artikulačními orgány

Při tvorbě vokálů se používá nejvíce jazyk, který svým pohybem mění objem v dutině ústní. Dále pak spolupracují rty, které mění svůj tvar (to je zřetelné například při vyslovení "o","u"). Při tvorbě konsonantů dochází k vytváření překážky v cestě

výdechovému proudu - striktury. Striktury dělíme na tyto typy: okluze (závěr), konstriktce (úžina), semiokluze (poloužina), vibrace (kmity). Při okluzi dochází k úplnému přerušení výdechového proudu a vznikají tím například "d", "t", "n". Při konstrikci prochází proud zúženým místem a používá se například při tvorbě "s", "z", "f". Semiokluze je slabá okluze a v průběhu jedné hlásky přechází v konstrikci. To se děje pro české "c" a "č". Při vibracích se úžina zvětšuje a zmenšuje (kmitá), příkladem je "r", "ř".

Se znalostí lidského hlasového ústrojí lze poměrně dobře modelovat jeho chování. Model, vycházející z fyzické struktury hlasového traktu a využívající neměnnost hlasového traktu pro krátké časové úseky, se nazývá lineární model hlasového traktu. Tento model využívá několik metod syntézy lidské řeči. I když není úplně dokonalý, přesto dosahuje při syntéze řeči dobrých výsledků. Navíc umožňuje snadnou změnu parametrů řeči (základní hlasový tón, intenzita, délka trvání atd.) a díky tomu modelovat prozodii řeči, měnit hlas mluvčího a provádět další úpravy pro zvýšení přirozenosti řeči. Ta je v současné době středem zájmů odborníků na syntézu řeči. Lidský mozek totiž řídí hlasový trakt a upravuje vlastnosti řeči podle vlastní potřeby. A právě tento problém není stále dokonale vyřešen. Počítač zatím není sám schopen určit, jak bude výsledná řeč vypadat (její prozodie). Proto je zatím syntetický hlas poměrně nepřirozený a posluchač snadno pozná, že se nejedná o běžného mluvčího.

3. Metody syntézy řeči

V dnešní době je již vyvinuto několik kvalitních metod pro syntézu řeči, a proto bych v této kapitole rád vysvětlil jejich principy. V současnosti jsou známy a používají se tyto metody syntézy řeči:

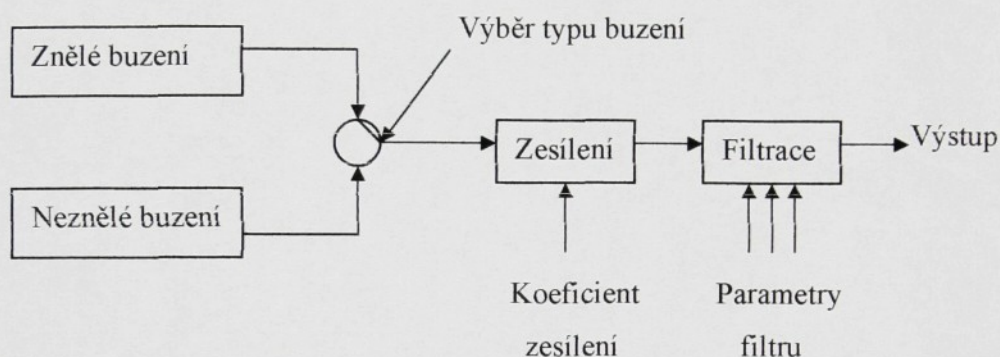
- Syntéza založená na spojování řečových úseků
- Pitch-synchronní spojování řečových úseků - PSOLA
- Formantová syntéza
- LPC syntéza
- Kepstrální syntéza
- Sinusová metoda

Z metod pro syntézu lze vybrat jak kvalitní metody, tak metody s menší paměťovou náročností, aplikovatelné například do signálových procesorů, takže nejsme omezeni jen na výkonné počítače. Metody, založené na spojování řečových úseků, disponují velmi vysokou kvalitou řeči. Parametrické metody nemají sice tak kvalitní výstupní signál, umožňují však velmi snadno měnit parametry tohoto signálu (základní hlasová frekvence, délka, atd.).

3.1. Lineární model řeči

Tento model řeči používá LPC, kepstrální a formantová metoda, a proto je třeba seznámit se s ním podrobněji. Lineární model vychází, jak název napovídá, z linearizace řečového signálu. Lidský hlasový trakt není schopen měnit své parametry okamžitě. Jeho změny jsou poměrně pomalé a lze říci, že v krátkém časovém úseku je hlasový trakt stacionární. Tento úsek trvá 10 – 30 ms (délka linearity je závislá na základní hlasové periodě). Jestliže je tedy signál rozdělen na úseky trvající 10 – 30 ms (často se používá délka 20 ms), pak jsou tyto úseky lineární. Tento předpoklad není ovšem dokonalý, jelikož například při tvorbě některých souhlásek (při tvorbě explosiv) dochází k náhlým změnám. V praxi se ovšem tato nelinearita zanedbává. Lineární model tedy nesyntetizuje celou promluvu najednou, ale syntetizuje pouze její segmenty. Blokové schéma lineárního modelu řeči je na obrázku 3.1 a skládá se ze dvou hlavních částí.

První je buzení, to modeluje proud vzduchu, modifikovaný hlasivkami. Druhou částí je číslicový filtr, který modeluje hlasový trakt. Zesílení ovlivňuje amplitudu výstupního signálu.



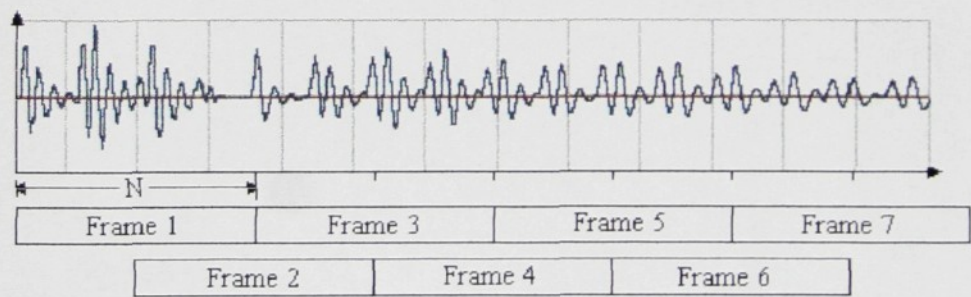
Obr 3.1 – *Blokové schéma lineárního modelu řeči*

Jako budící signál pro znělé hlásky je generována posloupnost jednotkových pulsů s periodou odpovídající základní hlasové periodě. Tento signál ovšem plně nevystihuje přesně budící signál v lidském hlasovém traktu. Reálný budící signál obsahuje i šumovou složku s omezeným výkonovým spektrem. Proto některé syntezátory používají takto rozšířený budící signál. Pro buzení neznělých hlásek je používán signál bílého šumu. Některé syntezátory tento šum ještě spektrálně omezují dolní propustí, aby neobsahoval vysokofrekvenční složky. Volba úpravy budících signálů (spektrální omezení signálů) se provádí na základě zvolené metody syntézy a také experimentálně. Budící signály jsou přepínány mezi signálem znělým a neznělým, podle typu segmentu, který právě syntetizujeme. Budící signál je dále amplitudově upraven pomocí zesílení. Posledním blokem je blok modelu hlasového traktu. Ten je realizován časově proměnným číslicovým filtrem řádu N . Konkrétní realizace filtru závisí na zvolené metodě syntézy. Tento filtr popisuje jednotlivé stupně lidského hlasového ústrojí. Ty jsou naladěny na jednotlivé formantové frekvence, které se vyskytují v daném řečovém segmentu. Výsledná přenosová funkce filtru je charakterizována těmito rezonančními frekvencemi. Parametry vstupující do filtru tomu odpovídají - jsou konstantní pro jeden segment. To vyplývá z předpokladu, že přenos hlasového traktu pro jeden segment je lineární. Pro

následující segment se parametry filtru mění (není-li průběh předchozího segmentu stejný). Výsledná kvalita syntézy, ale také výpočetní náročnost je dána řádem filtru.

3.2. Rozklad na segmenty a předzpracování signálu

Jak bylo uvedeno v předchozí kapitole, je třeba řečový signál rozložit na jednotlivé segmenty (framy), aby bylo dosaženo linearity signálu. Segmentace může být provedena synchronně nebo asynchronně. Asynchronní segmentace není závislá na vstupním řečovém signálu, jelikož je délka segmentu pevně dána. Segment má délku 10 – 30 ms (pro dětský hlas má délku i kratší) a obsahuje tedy 2 – 3 řečové periody. Při analýze řečového signálu často používáme rychlou fourierovu transformaci, která předpokládá, že vstupní signál je periodický. Asynchronní segmentace však rozděluje signál bez ohledu na jeho periodu, tudíž signál obsažený v jednom segmentu není periodický. Proto je třeba každý segment vážit okénkovou funkcí (nejčastěji používáme pro řečový signál Hammingovo okénko). Tím je zaručeno, že signál v jednom segmentu může být uvažován jako periodický. Jestliže však signál vynásobíme okénkovou funkcí, ztratíme tím část informací ze začátku a konce segmentu. Proto je vhodné jednotlivé segmenty překrývat. To je zobrazeno na obrázku 3.2, kde N je zvolená délka segmentu.



Obr 3.2 – Překrývání segmentů při asynchronní segmentaci

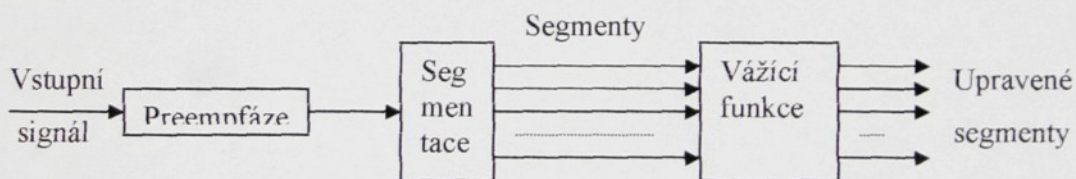
Při synchronní segmentaci (nazývané též pitch-synchronní) obsahuje jeden segment jednu periodu řečového signálu. Je třeba detekovat základní hlasovou periodu a podle ní nastavovat délku segmentu. Délka segmentu je tedy proměnná. Segment není potřeba vážit okénkovou funkcí, protože obsahuje právě jednu periodu a je tudíž periodický.

Protože segment nevážíme, neztrácíme žádnou informaci a není proto nutné používat překrývání segmentů.

Před analýzou signálu, která je potřeba pro výpočet parametrů vstupujících do modelu hlasového traktu, je vhodné předzpracovat řečový signál tak, aby byly potlačeny nežádoucí informace a žádoucí vynikly. To se většinou provádí pomocí preempfáze, kdy se v signálu zvýrazní vyšší frekvence, které by jinak nebyly tolik zřetelné jelikož je jejich výkonové spektrum menší než u složek s nižší frekvencí. Preempfáze je realizována jako číslicový filtr prvního řádu. Jeho diferenciální rovnice je ve tvaru

$$y(n) = x(n) - k \cdot x(n-1) \quad (1)$$

kde k je konstanta preempfáze, která se obvykle pohybuje v rozmezí 0,89 – 0,95, $x(n)$ a $x(n-1)$ jsou hodnoty vstupního řečového signálu. Hodnota upraveného výstupního signálu je udána $y(n)$. Celý postup při předzpracování a segmentaci pro analýzu signálu pro určení koeficientů modelu hlasového traktu je znázorněn na obrázku 3.3.



Obr 3.3 – Postup při úpravě a asynchronní segmentaci signálu

Vstupem je řečový signál, který má být analyzován a výstupem je M segmentů (jejich počet závisí na délce vstupního signálu), které lze považovat za periodické a mohou být převedeny do frekvenční oblasti pomocí fourierovy transformace. Nyní lze tedy použít tyto segmenty pro další analýzu dle zvolené metody modelování hlasového traktu. Tyto metody jsou popsány v následujících kapitolách.

3.3. Formantový syntezátor

Formantový syntezátor vychází z lidského hlasového ústrojí, kde je průchod kmitajícího vzduchu doplněn o jiné frekvence nebo zesílen pomocí rezonátorů a tlumen pomocí antirezonátorů. Nesnaží se modelovat do detailů lidské hlasové ústrojí, ale vytvářejí výsledný signál pomocí několika jednoduchých pravidel. Tomu odpovídá i kvalita výsledného signálu. Zatímco při použití LPC se snažíme modelovat přenosovou charakteristiku hlasového traktu najednou, u formantových syntezátorů je možné nastavovat jednotlivé parametry samostatně. Formantový syntezátor se tedy skládá ze zdroje buzení (znělého či neznělého) a z posloupnosti rezonátorů a antirezonátorů.

3.3.1. Rezonátory a antirezonátory

Základním prvkem formantového syntezátoru jsou rezonátory, sloužící k modelování spektrálních vrcholů (formantů). Jejich chování je dáno dvěma parametry, rezonanční frekvencí F_i a šířkou pásma B_i . Rezonátor je možno simulovat zpětnovazebním filtrem druhého řádu. Pro výstupní signál y platí diferenční rovnice

$$y(n) = A \cdot x(n) + B \cdot y(n-1) + C \cdot y(n-2) \quad (2)$$

kde $y(n-1)$ a $y(n-2)$ jsou dva předchozí výstupní hodnoty $x(n)$ je vstupní hodnota a A , B a C jsou konstanty. Pro ně platí

$$\begin{aligned} C &= -e^{-2\pi B_i T} \\ B &= 2 \cdot e^{-\pi B_i T} \cdot \cos(2\pi F_i T) \\ A &= 1 - B - C \end{aligned} \quad (3)$$

Pro modelování tzv. antiformantů, vznikajících vlivem vedlejších rezonátorů v hlasovém traktu, se používá antirezonátor. Ten je používán pro úpravu spektra budícího zdroje, přesněji řečeno k modelování propojení zdroje buzení a hlasového traktu, a v sériovém zapojení též k simulaci nazálů (hlásek, při jejichž tvorbě se používá dutina nosní). Jeho popis lze odvodit z předchozích vztahů. Přenosová funkce antirezonátoru je zrcadlovým

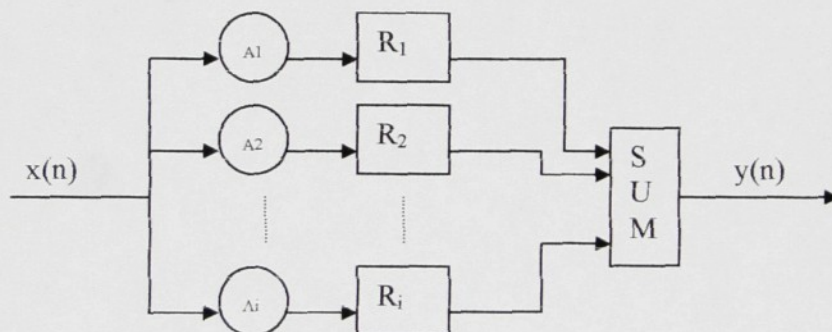
obrazem přenosové funkce rezonátoru. Diferenční rovnice antirezonátoru má tvar (4) a je z ní zřejmé, že výstupní hodnota je vypočítána pouze z hodnot vstupních

$$y(n) = A' \cdot x(n) + B' \cdot x(n-1) + C' \cdot x(n-2) \quad (4)$$

a parametry A' , B' a C' určíme podle vztahů

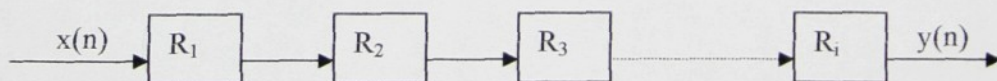
$$A' = \frac{1}{A}, \quad B' = -\frac{B}{A} \quad \text{a} \quad C' = -\frac{C}{A} \quad (5)$$

Při zapojení jednotlivých rezonátorů můžeme použít dva druhy zapojení - sériové a paralelní. V paralelním uspořádání každému rezonátoru R_i předchází obvod pro řízení amplitudy A_i . Ten nastavuje relativní velikost rezonanční křivky příslušného formantu.



Obr 3.4 – Blokové schéma paralelního zapojení rezonátorů

V sériovém řazení rezonátorů naopak individuální nastavování amplitud chybí, jednotlivé rezonátory jsou řazeny přímo za sebou.



Obr 3.5 – Blokové schéma sériového zapojení rezonátorů

Nevýhodou takovéto konfigurace je nemožnost dobrého generování hlásek buzených šumem. Výhodou je mnohem přesnější modelování vlastností hlasového traktu pro případ znělých hlásek.

Tyto rozdílné vlastnosti obou základních zapojení vedou přirozeně k myšlence zahrnout do syntezátoru obě dvě cesty, jak sériovou, tak i paralelní. Zřejmě nejznámějším formantovým syntezátorem je sériově/paralelní systém navrhnutý D. Klattem v roce 1979. Formantový syntezátor používá lineární model syntézy řeči. Pro vyšší přirozenost řeči se používá při znělém buzení průchodu posloupnost pulsů jednoduchým IIR filtrem druhého řádu. Tento filtr odstraní nespojité skoky buzení a řeč není tak "strojová". Buzení neznělých hlásek budící signál prochází dolní propustí s frekvenční charakteristikou klesající o 6 dB na oktávu. Při reprodukci se na výstup modelu hlasového traktu připojuje blok vyzařovací funkce. Ten simuluje chování signálu při vyzařování do prostoru. Jedná se o horní propust', jejíž charakteristika stoupá o 6 dB na oktávu. Celý model se tedy skládá ze dvou typů budících funkcí, které jsou dle potřeby přepínány. Dále pak z řady rezonátorů a antirezonátorů, které jsou mezi sebou zapojeny paralelně, sériově nebo kombinací sériového a paralelního zapojení a vyzařovací funkce.

3.4. LPC model hlasového traktu

Tento model vychází z analýzy signálu, kdy odhadujeme přenosovou funkci AR modelu, který signál modeluje. Tato analýza se nazývá lineárně prediktivní kódování (Linear Predictive Coding). Modelování řeči pomocí LPC koeficientů dosahuje lepších výsledků než metoda formantová, jelikož modelujeme hlasový trakt jako jednu přenosovou funkci, kdežto formantová syntéza modeluje hlasový trakt jako několik navzájem se neovlivňujících filtrů, což neodpovídá realitě. Vychází též z lineárního modelu syntézy, ale přenosová funkce hlasového traktu je realizována jako jeden číslicový filtr typu IIR řádu M (M se většinou pohybuje mezi 8 – 12). Tato syntéza se vyznačuje jednoduchostí a nízkou paměťovou a výpočtovou náročností.

3.4.1. Výpočet LPC koeficientů

Při výpočtu LPC koeficientů, vstupujících do číslicového filtru, se vychází z předpokladu, že n -tý vzorek modelovaného signálu lze určit pomocí hodnot předchozích a hodnoty vstupu. Odhad koeficientů je prováděn metodou nejmenších čtverců. V praxi ovšem není možno určit LPC koeficienty tak, aby predikovaný výstup přesně odpovídal modelovanému vstupu, a proto je definována chyba lineární predikce. Tato chyba je dána jako rozdíl vstupního signálu a signálu predikovaného. Jestliže tedy predikovaný signál označíme jako

$$s_p(n) = -\sum_{i=1}^M a_i \cdot s(n-i) \quad (6)$$

kde a_i jsou jednotlivé LPC koeficienty, $s(n)$ aktuální hodnota vstupního signálu a M řád filtru, můžeme chybu predikce zapsat jako:

$$e(n) = s(n) - s_p(n) = s(n) - \left(-\sum_{i=1}^M a_i \cdot s(n-i) \right) = s(n) + \sum_{i=1}^M a_i \cdot s(n-i) \quad (7)$$

Pro určení LPC koeficientů je tedy třeba minimalizovat chybu predikce, resp. její energii. Energie chyby je minimální, jestliže je její parciální derivace podle jednotlivých LPC koeficientů nulová. Řešení vede na soustavu rovnic, kterou můžeme napsat v maticovém zápisu takto:

$$\begin{bmatrix} R_n(0) & R_n(1) & \dots & R_n(M-1) \\ R_n(1) & R_n(0) & \dots & R_n(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_n(M-1) & R_n(M-2) & \dots & R_n(0) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix} = \begin{bmatrix} R_n(0) \\ R_n(1) \\ \vdots \\ R_n(M-1) \end{bmatrix} \quad (8)$$

$$\text{kde } R(i-j) = \sum_{n=0}^{N-1+M} s(n-i) \cdot s(n-j) \quad (9)$$

Tato soustava rovnic se řeší pro segment řečového signálu o délce N (segment obsahuje N vzorku). Další předpoklad je, že segment byl vážen okénkovou funkcí (nejčastěji Hammingovo okénko), tudíž začíná i končí nulou. Podrobné odvození řešené soustavy popisuje např. [1]. K řešení této soustavy rovnic lze použít Levinson - Durbinův algoritmus, jelikož je matice zobrazení symetrická a má všechny prvky na diagonálách stejné. LPC koeficienty $a_1, a_2, \dots, a_N$ prediktivní funkce získáme rekurzivním řešením následujících rovnic:

$$\begin{aligned}
 E_n^{(0)} &= R_n(0) \\
 k_i &= -\frac{R_n(i) + \sum_{j=1}^{i-1} a_j^{(i-1)} R_n(i-j)}{E_n^{(i-1)}} \\
 a_i^{(i)} &= k_i \\
 a_j^{(i)} &= a_j^{(i-1)} + k_i a_{i-j}^{(i-1)} \quad \text{kde } 1 \leq j \leq (i-1) \\
 E_n^{(i)} &= (1 - k_i^2) E_n^{(i-1)}
 \end{aligned} \tag{10}$$

Zde je k_i koeficient odrazu, a často se též používá k syntéze řeči. Navíc chyba lineární predikce popisuje budící signál, takže může být použita k jeho úpravě a tím získáme i kvalitnější výstup syntezátoru.

Vypočtené koeficienty jsou dosazeny do přenosové funkce číslicového filtru v tomto tvaru:

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}} \tag{11}$$

Tento filtr je pak použit jako model hlasového traktu do lineárního modelu řeči. Budící signál pak může být násoben energií chyby lineární predikce. Hodnoty filtru i hodnota zesílení musí být vypočteny a měněny pro každý segment. V praxi jsou nejprve vypočteny LPC koeficienty a energie chyby, ty jsou pak ukloženy do kódové knihy. Takto je analyzován celý signál. Při syntéze jsou parametry pro každý segment načítány přímo z kódové knihy.

3.5. Kepstrální model hlasového traktu

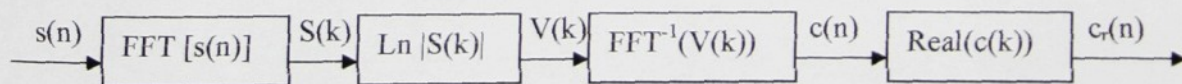
Tento model vychází z poznatků o tvorbě řečového signálu v lidském těle. Hlasový trakt je modelován pomocí přechodové funkce, která obsahuje póly i nuly (modeluje formanty i antiformanty). Další výhodou je, že oproti LPC modelu není zatížena chybou lineární predikce. V praxi však nelze přesně vyčíslit přenosovou funkci modelu, a proto je třeba nějak ji aproximovat z určitého počtu kepstrálních koeficientů. Výsledná chyba je tedy dána počtem použitých kepstrálních koeficientů a chybou použité aproximační metody. Díky tomu lze realizovat model s libovolnou přesností. Kepstrální metoda stejně jako metoda LPC vychází z lineárního modelu tvorby řeči. Přenosová funkce IIR filtru má tvar:

$$H(z) = \beta \cdot \frac{B(z)}{A(z)} \quad (12)$$

Zde β určuje velikost zesílení budícího signálu, $B(z)$ určuje nuly a $A(z)$ určuje póly přenosové funkce.

3.5.1. Výpočet kepstrálních parametrů

Výpočet kepstrálních parametrů vychází z časově - frekvenční analýzy signálu. Vokální trakt je proměnný v čase, a pokud chceme postihnout jeho změny ve spektrální oblasti, musíme analyzovat pouze segment takové délky, kde můžeme předpokládat stacionaritu traktu. Pro segmentaci je možno zvolit buď pitch synchronní nebo pitch asynchronní metodu. Rovněž je vhodné provést vážení signálu okénkovou funkcí (nejčastěji Hammingovým okénkem).



Obr 3.5 – Blokové schéma výpočtu reálného kepra

Pro lepší rekonstrukci signálu se používá překrývání jednotlivých segmentů. Řečový signál je možno zapsat jako konvoluci budícího signálu a impulzní charakteristiky modelu hlasového traktu. Tento zápis je vyjádřen touto rovnicí:

$$s(n) = x(n) * h(n) \quad (13)$$

Zde $s(n)$ je řečový signál, $x(n)$ je budící signál a $h(n)$ impulzní odezva. Následující rovnice popisují výpočet parametrů reálného kepstra signálu. Význam těchto operací je zřejmý jestliže pro popis řečového signálu použijeme vyjádření (13).

$$FFT\{s(n)\} = FFT\{x(n) * h(n)\} = FFT\{x(n)\} \cdot FFT\{h(n)\} = X(k) \cdot H(k) = S(k) \quad (14)$$

$$\ln|S(k)| = \ln|X(k) \cdot H(k)| = \ln|X(k)| + \ln|H(k)| \quad (15)$$

$$c_x(n) = \text{Re}(IFFT(\ln|X(k)|)) \quad (16)$$

$$c_h(n) = \text{Re}(IFFT(\ln|H(k)|)) \quad (17)$$

$$c_r(n) = \text{Re}(IFFT(\ln|S(k)|)) = c_x(n) + c_h(n) \quad (18)$$

Vlastností reálného kepstra je, že vstupní signál rozkládá na součet budícího signálu a impulzní odezvy hlasového modelu. Jelikož byl proveden tento rozklad ve frekvenční oblasti a je známo, že odezva hlasového traktu je řádově pomalejší než je frekvence buzení, lze vyvodit, že počáteční hodnoty reálného kepstra reprezentují hlasový trakt a vzdálenější hodnoty jeho buzení. Z reálného kepstra lze také vyčíst základní periodu řeči, která se promítne jako maximum vzdálenějších hodnot kepstra. Hranice mezi počátečními a vzdálenějšími hodnotami kepstra není pevně definována (ani ji pevně definovat nelze), ale orientačně ji lze stanovit mezi 25 až 50 koeficientem.

3.5.2. Aproximace exponenciálního rozvoje kepstrálního modelu

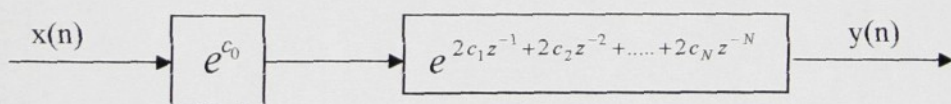
Jestliže se použije pouze N koeficientů, je získán omezený kepstrální rozvoj, který lze zapsat jako:

$$C(z) = c_0 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_N z^{-N} \quad (19)$$

Z tohoto rozvoje však nelze přímo získat původní řečový signál. Syntézu je třeba realizovat pomocí exponenciální funkce, která je inverzní k logaritmu, jenž byl použit při výpočtu reálného kepstra. Avšak díky této exponenciální funkci je syntéza prakticky nerealizovatelná a je třeba tuto funkci nějak aproximovat. Filtr realizující syntézu je definován:

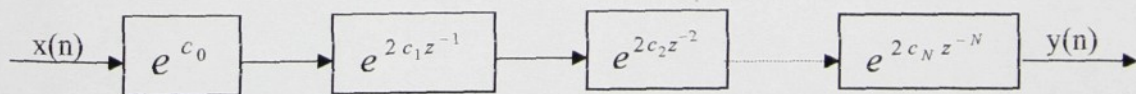
$$H(z) = e^{c_0 + 2c_1 z^{-1} + 2c_2 z^{-2} + \dots + 2c_N z^{-N}} = e^{c_0} \cdot e^{2c_1 z^{-1} + 2c_2 z^{-2} + \dots + 2c_N z^{-N}} \quad (20)$$

V literatuře se konstantní člen e^{c_0} často označuje jako zesílení β a je rovno střední hodnotě logaritmu modulu spektra řeči. Model hlasového traktu lze realizovat buď souhrnným (Obr. 3.6) a nebo kaskádním modelem (Obr. 3.7). Souhrnný model sdružuje kepstrální koeficienty do přenosové funkce jednoho číslicového FIR filtru.



Obr 3.6 – Blokové schéma realizace modelu hlasového traktu souhrnným modelem

Kaskádní model realizuje hlasový trakt jako posloupnost za sebou jdoucích filtrů, kde přenosovou funkci každého filtru definuje jeden kepstrální koeficient.



Obr 3.7 – Blokové schéma realizace modelu hlasového traktu kaskádním modelem

Exponenciální funkci souhrnného modelu nebo jednotlivé funkce kaskádního modelu lze aproximovat rozkladem na řetězové zlomky konečné délky. Tento rozklad se provádí podle následující rovnice:

$$e^x = 1 - \frac{x}{1} + \frac{x}{2} - \frac{x}{3} + \Lambda + \frac{x}{2} - \frac{x}{2s-1} \quad (21)$$

Zde je s konečné číslo, které odpovídá délce rozvoje, a za x se dosadí buďto celá přenosová funkce (u souhrnného modelu) nebo jen dílčí přenos pro daný keprstrální koeficient (u kaskádního modelu). Přesnost aproximace závisí na počtu členů rozvoje řetězového zlomku. Další možnost aproximace je pomocí Padého aproximace. Ta aproximuje exponenciální funkci jako racionální lomenou funkci. Rovnice (22) popisuje aproximaci přenosu $\exp(2C(z))$, který odpovídá přenosu číslicového filtru u souhrnného modelu.

$$e^{2C(z)} = \frac{1 + \alpha_{s1}C(z) + \alpha_{s2}C^2(z) + \dots + \alpha_{ss}C^s(z)}{1 - \alpha_{s1}C(z) + \alpha_{s2}C^2(z) + \dots - \alpha_{ss}C^s(z)} \quad (22)$$

Tato aproximace je s -tého řádu. Hodnoty členů $\alpha_{s1}, \alpha_{s1} \dots \alpha_{ss}$ lze nalézt např. v [4]. Aproximaci lze přepsat i pro dílčí filtry, použité v kaskádním zapojení. Přesnost aproximace závisí na volbě řádu aproximace.

Při tvorbě keprstrálního syntezátoru je tedy třeba nejdříve vypočítat keprstrální koeficienty jednotlivých segmentů, a poté aproximovat exponenciální přenosovou funkci modelu. Po aproximaci dostaneme přenos, odpovídající přenosové funkci modelující hlasový trakt. Tento přenos budíme znělým, resp. neznělým buzením, násobeným vhodným zesílením (to odpovídá energii modelovaného signálu), a výsledkem je syntetizovaný řečový signál.

3.6. Sinusový model řeči

Tento model získává své parametry analýzou signálu ve frekvenční oblasti. Řečový signál je nejprve transformován do frekvenční oblasti pomocí rychlé Fourierovy

transformace. Ve frekvenční oblasti je pak tento signál analyzován a jsou z něj získány parametry pro syntézu (dané hodnoty amplitud, frekvencí a fází). Syntetizovaný signál je pak generován podle rovnice

$$y(n) = \sum_{i=1}^N A_i \cos(i\omega_0 n + \varphi_i) \quad (23)$$

kde A_i a φ_i je amplituda a fáze i -té složky signálu, ω_0 je základní kruhová frekvence a N je počet složek, ze kterých signál skládáme. Tento počet ovlivňuje kvalitu vzniklé řeči. Parametry modelu lze získat výběrem lokálních maxim amplitudového spektra nebo aproximací obálky amplitudového spektra.

3.6.1. Výběr lokálních maxim amplitudového spektra

Tato metoda vychází z faktu, že sinusové průběhy, které mají největší spektrální čáry, přenášejí největší výkon, a proto také nesou nejvíce informací pro rekonstrukci signálu. Algoritmus vyhledávání těchto čar je poměrně jednoduchý. Je procházeno čárové spektrum, a jestliže je zjištěna změna polaroty přírůstku jednotlivých spektrálních čar, vybere se čára, která je větší než čáry napravo a nalevo od ní. Protože tyto úseky mezi jednotlivými maximy nejsou přesně ekvidistantní, neukládáme jejich skutečnou hodnotu, ale vzdálenost mezi nimi. Průměrná spektrální vzdálenost těchto maxim pak odpovídá základní hlasové frekvenci mluvčího. Lze ji určit jako podíl vzdálenosti poslední vybrané čáry a počtu všech vybraných čar.

3.6.2. Aproximace obálky amplitudového spektra

Tuto metodu lze realizovat například pomocí algoritmu SEEVOC (Spectral Enevelope Estimation Vocoder). Ten vychází z definované frekvence ω_0 . Nejdříve je nalezeno maximum v intervalu od poloviny základní frekvence do jejích tří polovin. V nalezeném maximu je pak odečtena hodnota amplitudy A_1 a frekvence ω_1 . Poté je třeba se ve spektru posunout o ω_1 a opět hledat maximum. Nyní v intervalu od $\omega_1 + \omega_0/2$ do $\omega_1 + 3\omega_0/2$. Po nalezení maxima jsou odečteny hodnoty amplitudy A_2 a frekvence ω_2 . Takto se

postupuje až do konce spektra. Pokud někdy v intervalu maximum není nalezeno, pak se za amplitudu dosadí větší z krajních hodnot intervalu a za frekvenci hodnota frekvence střední čáry intervalu. Z takto získané obálky spektra odpovídající hlasovému traktu jsou získány hodnoty fází. Průběh fázové charakteristiky se ekvidistatně navzorkuje.

Parametry získané pomocí jedné z těchto dvou metod dosadíme do rovnice (23) a získáme z ní průběh syntetizovaného řečového signálu, který je součtem sinusoid jednotlivých harmonických frekvencí signálu. Výsledný model syntézy řeči tedy není tvořen průchodem buzení číslicovým filtrem, ale pouze skládáním sinusových vlny, definovaných pomocí parametrů získaných frekvenční analýzou signálu.

3.7. Syntéza založená na spojování řečových úseků - PSOLA

Jedny z prvních syntezátorů vytvářely řeč spojováním nahraných úseků řeči. Tato metoda je velmi jednoduchá. Je třeba zvolit vhodné jednotky, ze kterých je výsledná promluva skládána. Ty je třeba pomocí zvukové karty nahrát a uložit do paměti počítače (případně jiného hardwaru). Výsledná řeč pak vzniká jednoduchým spojováním těchto jednotek v pořadí podle vstupního textu. Výsledný signál může být velmi kvalitní promluvou, pokud jsou vhodně zvoleny stavební jednotky. Jestliže je potřeba syntetizovat pouze malé množství předem známých promluv (např. hlasové samoobsluhy mobilních operátorů), pak je nahrána každá promluva jako celek a výsledek je oproti ostatním syntezátorům "bezkonkurenční". Ovšem v případě, že je třeba aby syntezátor dokázal syntetizovat jakoukoli promluvu, pak je výsledek velmi nízkých kvalit, bez jakékoliv možnosti změny parametrů promluvy (základní hlasový tón, její délka, atd.). Postupným vývojem byla tato metoda modifikována a vedla k vzniku metody nazývané PSOLA.

3.7.1. PSOLA (Pitch-Synchronous Overlap-Add)

PSOLA je metoda syntézy řečového signálu v časové oblasti. Vychází z namluveného lidského řečového signálu. Ten je analýzou rozdělený na malé úseky, jenž odpovídají jednotlivým periodám základního hlasového tónu. Výsledný syntetizovaný signál vzniká spojováním jednotlivých period za sebou tak, aby dosáhl požadovanou délku a výšku [15]. Výšku hlasu (základní hlasový tón) můžeme měnit změnou vzdálenosti jednotlivých

period nebo změnou délky periody (např. interpolací). Změnu délky dosáhneme změnou počtu period (opakováním nebo vynecháním). Tyto úpravy je možné provádět pouze na periodických úsecích. Na neperiodických úsecích měníme pouze jejich délku. Díky tomu, že jsou spojovány úseky o velikosti právě jedné periody, je eliminována možnost napojení úseků, které končí resp. začínají jinou hodnotou nebo jejich derivace v koncovém resp. počátečním bodě je jiná. Při napojování jednotlivých úseků je třeba dbát na to, aby nebyl napojován úsek, který končí jinak, než následující úsek začíná. Jestliže tento předpoklad není splněn, je výsledná promluva nespojitá a dochází k nežádoucím zvukovým efektům ve výstupním signálu (tzv. "lupání", tvorba vyšších frekvencí, atd.). Proto bylo třeba modifikovat metodu jednouchého spojování řečových úseků, kde tento jev vznikal (při tvorbě libovolné promluvy). Pro lepší napojování úseků se dále používá vážení úseku okénkovou funkcí. Aby nedošlo k velké ztrátě informace, váží se úsek signálu z dvojnásobnou délkou než je délka periody a střed okénka se umístí nad maximum úseku. Metoda PSOLA umožňuje modelovat prozodii díky možnosti měnit základní hlasovou periodu, intenzitu a délku úseku. Tyto změny jsou však provázeny ztrátou kvality syntetizovaného signálu (zejména změna základního hlasového tónu). Metoda PSOLA je tedy schopna syntetizovat velmi kvalitní řečový signál, který ztrácí svou kvalitu při modelování prozodie. Proto se v dnešní době, kdy se výzkum zabývá zlepšováním přirozenosti řeči, od metody PSOLA ustupuje a přechází se zpět k metodám LPC, kepstrální nebo sinusové.

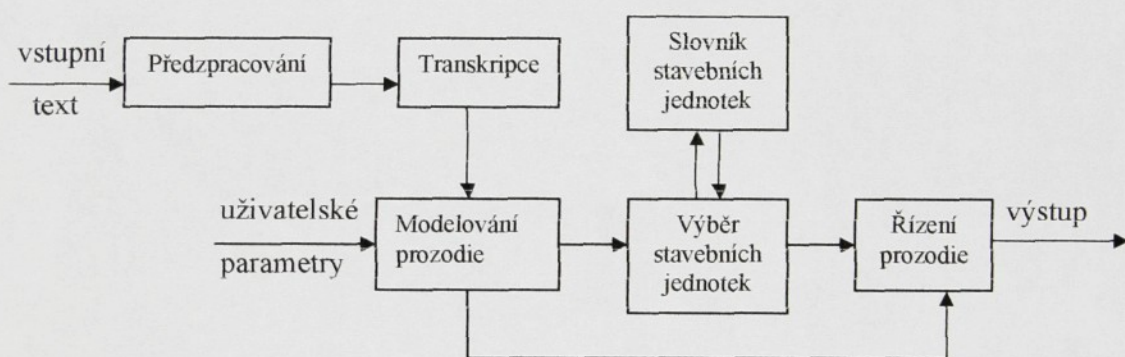
4. TTS Systémy (Text To Speech)

Systémy, které jsou schopné převádět psaný text na mluvené slovo jsou nazývány TTS systémy. Jedná se tedy o systémy, které jsou nejen schopny syntetizovat řečový signál, ale dále dovedou převést text z psané podoby do podoby, ve které je vyslovován (fonetická transkripce). Poté takto předzpracovaný text rozdělí na základní stavební jednotky a spolu s prozodickými parametry vstupují do bloku vlastní syntézy. Je tedy zřejmé, že tyto systémy řeší několik poměrně složitých problémů, z nichž jedním je vlastní syntéza řeči. V dnešní době jsou možnosti využití těchto systémů velmi rozsáhlé. Můžeme se s nimi setkat v bankovníctví, telekomunikacích a slouží i při komunikaci zařízení s nevidomými. Na naší univerzitě byl vyvinut systém InfoCity, který též syntézu využívá. TTS systémy mají velikou budoucnost, jelikož řeč je pro člověka nejpřirozenějším komunikačním prostředkem.

TTS systémy lze dělit podle několika kritérií. Rozlišují se systémy, kde je text zpracováván a převeden na řečový signál mimo hlavní aplikaci a provádí se většinou předem (off-line systémy) a systémy, kdy je text převáděn přímo v rámci hlavního programu (on-line systémy). U on-line systémů je kladen nárok na rychlost převodu. Tato rychlost by měla být menší než půl sekundy, aby byly informace sdělené uživateli aktuální. Dále se dělí TTS systémy podle toho, jaké množství různých promluv jsou schopny generovat. Systémy s neomezeným slovníkem jsou schopny generovat jakoukoliv promluvu. Existují ovšem i systémy, které mají omezený slovník. Tyto systémy jsou vyvíjeny pro konkrétní aplikace a jejich syntetizovaná promluva je kvalitnější než u systémů z neomezeným slovníkem. To je způsobeno větší velikostí stavebních jednotek, ze kterých skládáme výslednou promluvu. Existují i systémy, které mají slovník pro syntézu libovolného textu, ale také obsahují větší stavební jednotky, které se často v promluvách vyskytují (např. v bankovníctví čísla – systém má slovník, který obsahuje celá čísla a při syntéze se neskládají ze základních stavebních jednotek, ale jsou syntetizovány jako celek z rozšířeného slovníku). Pak je kvalita promluvy pro často používané výrazy (obsažené v rozšířeném slovníku) lepší než u klasických systému s neomezeným slovníkem.

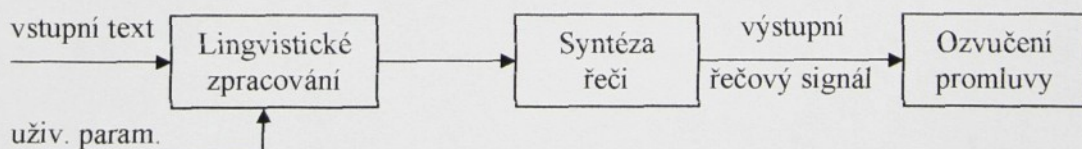
4.1. Struktura TTS systémů

TTS systémy se skládají jednak z lingvistické části, která zpracovává a vyhodnocuje vstupní text. Výsledkem této části je seřazená posloupnost základních řečových jednotek spolu s parametry, se kterými jsou jednotlivé jednotky syntetizovány. Tato posloupnost je vstupem do syntezátoru řeči, který je druhou částí TTS systému. Výstup syntezátoru pak v případě on-line systému bývá přímo zvukový signál generovaný ve zvukové kartě, nebo v případě off-line systému zvukový soubor (wave) uložený na disku PC.



Obr 4.1 – Blokové schéma lingvistické části TTS systému

Struktura lingvistické části TTS systému je znázorněná na obrázku 4.1. Blokové schéma celého syntezátoru je pak na obrázku 4.2. Poslední blok zajišťuje převod výstupního signálu do akustické podoby. Tento blok může být nahrazen dle potřeby blokem, který provede uložení výstupního signálu do zvukového souboru.



Obr 4.2 – Blokové schéma TTS systému

4.1.1. Předzpracování

Blok předzpracování má za úkol převést různé zkratky a jiné textové řetězce tak, aby mohli vstoupit do bloku fonetické transkripce. Ve vstupním textu se mohou vyskytovat takové zkratky, jako je např. "atd.", "resp." a jiné. Dalšími řetězci, které se při předzpracování převádí, jsou číslovky, údaje o datumu, hodinách, matematické operátory atd. Vstupní text "Promoval 16.6.2004" bude převeden na text "Promoval šestnáctého června dva tisíce čtyři" stejně tak text " $0,12 + 0,36 = 0,48$ tj. 48%" bude převeden na "nula celá dvanáct plus nula celá třicet šest rovná se nula celá čtyřicet osm to je čtyřicet osm procent". Převod je prováděn na základě pravidel, které jsou součástí tohoto bloku. Je vhodné, aby uživatel mohl pravidla vytvářet a editovat, jelikož často se mohou v různých souvislostech (tyto souvislosti nemusí být patrné ze syntetizované promluvy) různé symboly nebo zkratky číst různě. Např. řetězec "825913/8456" může vyjadřovat rodné číslo, kde lomítko nečteme nebo čteme jako "lomeno", nebo může vyjadřovat matematický příklad, kde lomítko čteme jako "děleno". Podobný problém může nastat při zápisu datumu jak "11/12/2004". Konkrétní aplikace TTS systému pomůže zjednodušit psaní těchto pravidel a jejich bezchybnost.

4.1.2. Fonetická transkripce

Blok fonetické transkripce zajišťuje přepis předzpracovaného textu do fonetické podoby. Tento přepis je závislý na výběru stavebních jednotek. Jestliže je použit inventář tvořený např. z fonémů, je třeba zkoumat písmeno předcházející a následující, protože tato písmena ovlivňují v jaký foném se písmeno přepíše. V případě trifónového inventáře nemusíme zkoumat předchozí znaky, jelikož vzniká kombinací tří fónémů (tvorba trifónů bude vysvětlena níže). Je třeba ovšem přepsat jednotlivá slova jako kombinaci trifónů. Fonetická transkripce je prováděna na základě fonologických pravidel. Tato pravidla nám popisují za jakých podmínek a jak se přepíše daný text. Například pro fonémový inventář bude pravidlo vypadat podle [1] následovně: Jestliže řetězec A předchází bezprostředně řetězec znaků C a je bezprostředně následován řetězcem znaků D, pak se přepíše na řetězec znaků B. Toto pravidlo lze zjednodušeně napsat ve tvaru:

$$A \rightarrow B/C_D$$

(24)

Konkrétní pravidla pak budou vypadat následovně: $ch \rightarrow X/_$ (pro přepis každého písmena "ch" na foném [X]), $q \rightarrow kv/_$ (pro přepis písmena "q" na fonémy [k] a [v]) nebo $d \rightarrow d'/_ \langle i, i \rangle$ (pro přepis písmen "di" nebo "dí" na fonémy [d'] a [i] resp. [d'] a [i]). Těchto pravidel je třeba pro úplnou fonetickou transkripci sestavit mnohem více. Fonetickou transkripci využívají jak systémy pro syntézu řeči, tak systémy pro rozpoznávání řeči. Proto bylo vyvinuto několik programů, realizující automatickou fonetickou transkripci. Algoritmus pro automatickou fonetickou transkripci je popsán v [1], alternativní způsob fonetické transkripce, fungující na principu neuronové sítě, je popsán v [6]. Více se zde o fonetické transkripci zmiňovat nebudu, jelikož je to problematika poměrně obsáhlá a přesahuje rámec mé práce. Podrobnější informace o fonetické transkripci lze nalézt v [1] a [6].

4.1.3. Modelování prozodie

Prozodie postihuje vlastnosti promluvy, které se vztahují k vyšším jednotkám než jsou segmenty nebo v některých případech i stavební jednotky. Prozodie se aplikuje na slabiky, promluvové takty (slabika nesoucí slovní přízvuk a jistý počet nepřízvučných slabik), celé věty nebo souvětí. V problému modelování prozodie je definován promluvový úsek, který je nadřazený taktu. Promluvový úsek je skupina taktů, které jsou svým intonačním průběhem vázány do určitého celku. Při modelování prozodie na základní úrovni se modeluje melodie průběhu promluvy. Jako jednotka je brána věta. Melodie věty se modeluje pomocí změny průběhu základního hlasového tónu. Tento průběh je označován jako větný melodém. Pro úplné modelování prozodie je třeba vedle větného melodému, modelovat i slovní prozodii (přízvuk, prodlužování a zkracování) a modelovat i mikrointonaci (intonace mezi jednotlivými fonémy). Kromě modelování melodému lze ještě modelovat prozodii změnou intenzity výstupního signálu a změnou doby trvání fonému. Při modelování prozodie jsou na základě analýzy textu přiděleny jednotlivým stavebním jednotkám dle pevně stanovených pravidel hodnoty základního hlasového tónu, intenzity a délky trvání jednotek. Tvorba těchto pravidel je poměrně složitá, jelikož je potřeba analyzovat mnoho nahrávek přirozených promluv a zjišťovat v

jakém případě je kladen na slabiku přízvuk, jak probíhá větný melodém, atd. Aby nevznikl příliš velký slovník pravidel, je třeba pravidla zobecňovat, což není jednoduché.

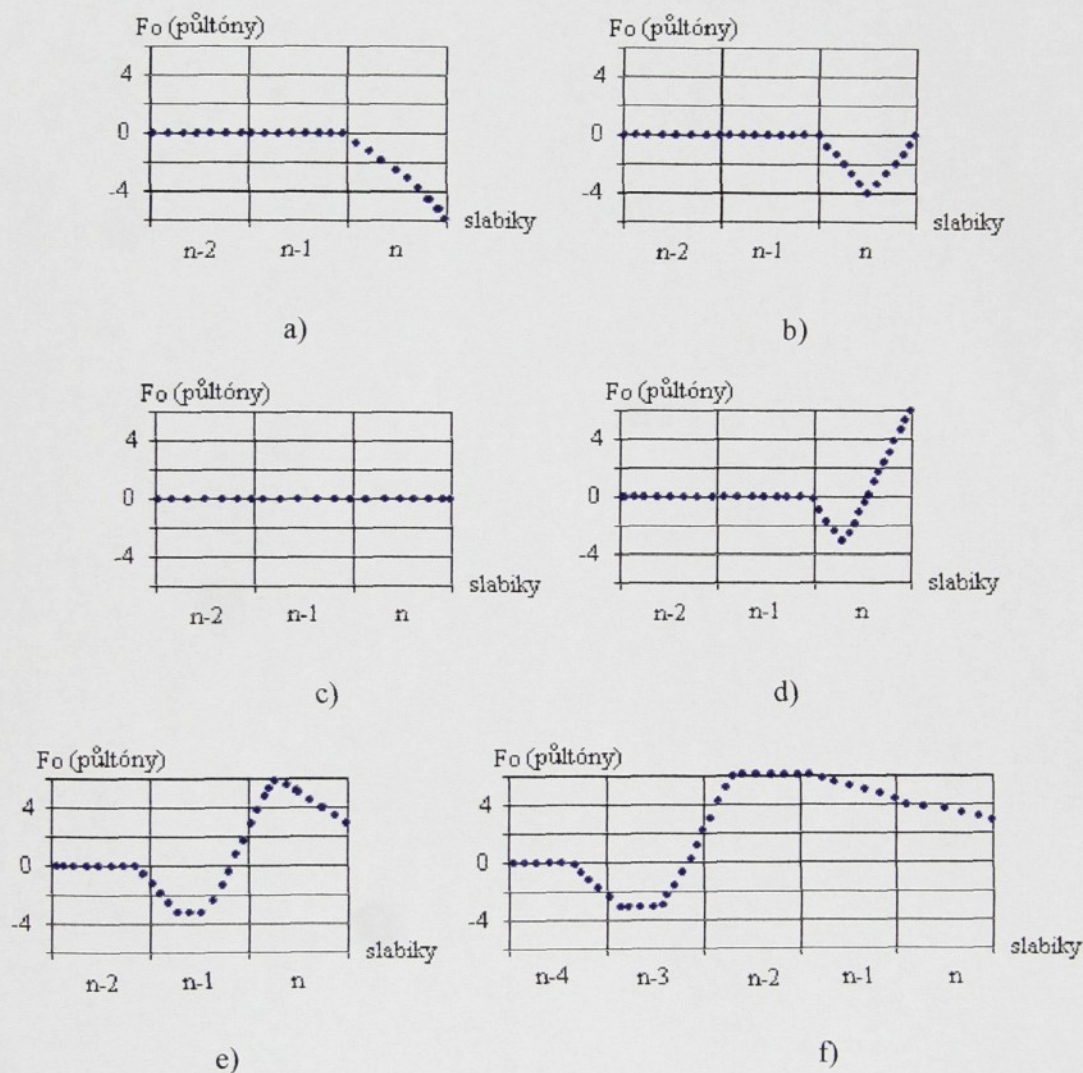
Slovní přízvuk je v češtině pevný a váže se na první slabiku slova. Proto je považován za hranici mezi jednotlivými slovy. Může se ovšem stát, že slovu s přízvukem předchází jednoslabičné slovo bez přízvuku (pan Novák). Přízvuk ve frekvenční oblasti lze modelovat jako pokles nebo zvýšení základního hlasového tónu o jeden až dva čtvrttóny. V oblasti intenzity je vhodné zvýšit intenzitu zhruba o 3dB. Občas se však v přirozené řeči nacházejí přízvučné slabiky s intenzitou stejnou nebo dokonce nižší než je intenzita okolí. V časové oblasti se slovní přízvuk neprojevuje změnou, tudíž se ani nemodeluje.

Při modelování větné prozodie se snažíme jednak odlišit větu, která končí, od věty, která bude následovat, jednak dát posluchači najevo o jaký typ věty se jedná (oznamovací, tázací atd.). Toto jsou hlavní problémy při modelování prozodie věty. Dalším tématem je modelování citově zabarvených vět. K oddělení jednotlivých vět se používá kromě modelování větné melodie i oddělovací pauza. Trvání mezivětné pauzy je proměnné podle interpunkčního znaménka, kterým věta končí. U TTS systémů jsou většinou používány tři různé délky pauz, resp. jedna základní délka pauzy a její násobné řazení za sebe. Základní délka pauzy se většinou používá pro znaménka ",", "(" a ")". Dvojnásobná délka pak pro znaménka ":", ";", a "-". Trojnásobek základní délky je používán pro ".", "?", a "!". Základní délka trvání pauzy se pohybuje kolem 150ms. Pro modelování větné melodie jsou často rozlišovány tyto typy vět:

- 1) Neutrální věta oznamovací, která je ukončena znaménkem "."
- 2) Nekoncová věta v souvětí končící znakem ","
- 3) Nekoncová věta v souvětí končící znakem "-"
- 4) Věta tázací, která je ukončena znaménkem "?"

U věty oznamovací se melodém tvoří lineárním klesáním průběhu základního hlasového tónu přes celý ovlivňovaný úsek (jeden či více taktů). Základní hlasový tón se pro jednotlivé segmenty snižuje o takovou hodnotu, aby celkový pokles na konci činil šest půltónů. Melodém nekoncové věty před znakem "," nejdříve lineárně klesá a následně opět lineárně stoupá na svou původní hodnotu. Celkový pokles je o čtyři půltóny a

minimální hodnota leží ve středu ovlivňovaného úseku. Úbytek při klesání a přírůstek při stoupaní mají tudíž stejnou hodnotu, pouze opačné znaménko. Při modelování nekonečné věty před znakem "-" se větná melodie nemění pouze se vkládá pauza. Melodém otázky zjišťovací je různě modelován pro různé počty slabik v ovlivňovaném úseku. Průběhy všech melodémů jsou zobrazeny na obrázku 4.3. a,b,c,d,e,f.



Obr 4.2 – Průběhy větných melodémů pro větu a) oznamovací, b) neutrální končící “,”, c) neutrální končící “-” a pro větu tázací s koncovým úsekem d) jednoslabičným e) dvojslabičným f) vícetřislabičným.

Z výše uvedených faktů je zřejmé, že uvedená pravidla pro modelování prozodie jsou pouze ta základní. Pro kvalitnější a přirozenější promluvu systému je třeba vytvářet složitější pravidla, zahrnující jak modelování průběhu základního hlasového tónu, tak i modelování intenzity a změny délky trvání jednotlivých úseků. Modelování prozodie je velmi složitý problém sám o sobě, a proto jsou zde nastíněny pouze hlavní problémy a jednoduchá řešení.

4.2. Inventář stavebních jednotek

Volba vhodného inventáře stavebních jednotek je jedním z hlavních kritérií ovlivňujících výslednou kvalitu syntetizované řeči. Řečové inventáře vycházejí ze struktur, z kterých se řeč skládá (bez ohledu na jazyk). Mezi tyto jednotky patří hlavně fonémy, alofony, dióny a slabiky. Při výběru typu řečového inventáře se lze držet pravidla, že čím je inventář rozsáhlejší, tím je syntetizovaná řeč kvalitnější. Ovšem čím je inventář rozsáhlejší, tím jsou hardwarové i softwarové nároky vyšší.

4.2.1. Fonémový inventář

Foném je základní fonetická jednotka, která reprezentuje nejmenší počet foneticky odlišných tříd. Foném odpovídá nejkratšímu úseku promluvy, který když zaměníme za významově jiný, vznikne významově lišící se promluva [5]. V českém jazyce se vyskytuje zhruba 40 fonémů (závisí na konkrétním fonetickém inventáři), které definuje např. SAMPA (viz. příloha č.1). Jak z definice vyplývá, foném odpovídá jednomu písmenu v psaném textu (není to ovšem pravidlo). Protože se některá písmena vyslovují výrazně jinak v různých kombinacích okolních písmen, je většinou více fonémů než písmen. Fonetický inventář tedy obsahuje jednotlivé fonémy a výsledná promluva se vytváří skládáním jednotlivých fonémů za sebe. Fonetická transkripce tedy převede jednotlivá písmena z psaného textu na odpovídající fonémy. Výhoda inventáře je ve velmi malém počtu jednotek (kolem 40). Mezi jednotlivými fonémy však u přirozené řeči probíhá koartikulace (mikrointonace), což je spojitá změna hlasového traktu ze stavu odpovídajícímu končícího fonému do stavu začínajícího fonému. Při skládání promluvy

z jednotlivých fonémů nastává absence tohoto jevu, a proto je výsledná promluva poměrně nekvalitní.

4.2.2. Difónový inventář

Difón popisuje spojení samohláska – souhláska tak, že difónu odpovídá úsek promluvy ze středu dané samohlásky přes koartikulační změnu do středu následující souhlásky nebo naopak. Difónový inventář obsahuje jednak tyto difony, dále pak spojení mezi středy dvou samohlásek, těla souhláskových skupin, hlásky počáteční a koncové. Velikost úplného difónového inventáře se pohybuje kolem 1200 jednotek. Existují i redukované inventáře, které obsahují difony s výraznou koartikulací. Difony s nevýraznou koartikulací skládá pouze spojováním jejich těl. Příkladem redukovaného slovníku je inventář, vyvinutý ve spolupráci Fonetického ústavu FF UK a ÚRE AV ČR a aplikovaný v TTS systému Epos. Tento difónový inventář obsahuje 441 jednotek. Zajišťuje vyšší přirozenost syntetického hlasu díky zahrnutí koartikulačních změn při syntéze. Redukované inventáře nabízejí možnost značného poklesu počtu stavebních jednotek. Výsledná kvalita je menší než u úplného inventáře, ale tento pokles není při vhodné redukci lineární. Redukovaný inventář je možné použít v hardwarově nenáročných aplikacích. Hardwarové nároky jsou nízké a kvalita řeči je poměrně vysoká. V přirozené řeči je však podoba konkrétní hlásky ovlivněna jak hláskou následující, tak hláskou předcházející. To difónový inventář nepopisuje, jelikož se jedná o spojení pouze dvou po sobě jdoucích hlásek.

4.2.3. Trifónový inventář

Tento inventář vychází z faktu zmíněného výše, totiž, že podoba konkrétní hlásky je dána hláskou předchozí a následující. Tento inventář tedy obsahuje oboustranné přechody mezi jednotlivými hláskami. Jestliže je tedy text „**ahoj**“ rozložen na jednotlivé trifóny, pak vypadá takto: *_ah aho hoj oj_*, kde *_ah* je trifón, v němž „**a**“ předchází ticho („**a**“ je tedy počáteční hláska) a následuje „**h**“. Počet jednotek tohoto inventáře se pohybuje kolem 6000 jednotek. Trifónový inventář zaručuje velmi dobrou kvalitu výsledného signálu, jelikož zahrnuje velmi velké množství koartikulačních informací, a

proto je výsledná řeč poměrně přirozená. Trifónový inventář byl vyvinut např. na Západočeské univerzitě v Plzni, kde byl vytvořen automatickou segmentací řečového korpusu na základě metody skrytých Markovových modelů.

4.2.4. Poloslabikové a slabikové inventáře

Slabiky jsou obecně považovány za fonetické útvary, které obsahují samohláskové jádro, plus volitelné počáteční a koncové souhlásky nebo skupiny souhlásek. Délka slabiky se v češtině může pohybovat od jednoho po šest fonémů. Navíc je možno celkem jednoduše akusticky identifikovat a lokalizovat jádro slabiky. Poloslabikový inventář obsahuje jednotky, složené z jedné nebo více souhlásek a následující samohlásky nebo diftongu (samohlásková skupina – např. “ou”). Počet jednotek poloslabikového inventáře je zhruba 7000. Slabikový inventář obsahuje jednotky, které se skládají ze středu samohlásky přes souhlásku či souhláskovou skupinu do středu následující samohlásky. Tento inventář může mít velmi rozdílnou velikost podle volby slabik a řádově obsahuje 20000 jednotek. Existují i inventáře, které obsahují celá slova nebo krátké věty. Ty se uplatňují, zvláště když systém často syntetizuje malou množinu konkrétních promluv. Tyto inventáře umožňují velkou přirozenost syntetizované promluvy. V širším uplatnění systému by však velikost inventáře tvořeného ze slov nebo krátkých vět narostla do hardwarově neúnosných rozměrů. I realizace slabikového inventáře je již poměrně náročnou záležitostí jak hardwarově, tak softwarově.

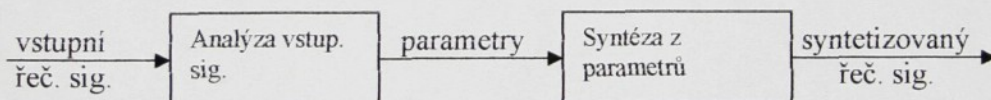
Praktická část

5. Realizace syntezátoru

V předchozí části byly popsány principy nejrozšířenějších metod syntézy lidské řeči. Tyto metody pak byly předmětem mého zkoumání. Blíže jsem se seznámil s metodami formantovou, LPC a kepstrální. Na závěr jsem provedl úplnou realizaci LPC syntezátoru. Dle mého názoru je však důležité seznámit čtenáře se všemi metodami, jelikož tak může získat objektivní pohled na celou problematiku. Dále může teoretická část posloužit jako studijní materiál pro čtenáře, kteří mají zájem zabývat se touto problematikou.

5.1. LPC Vokodér

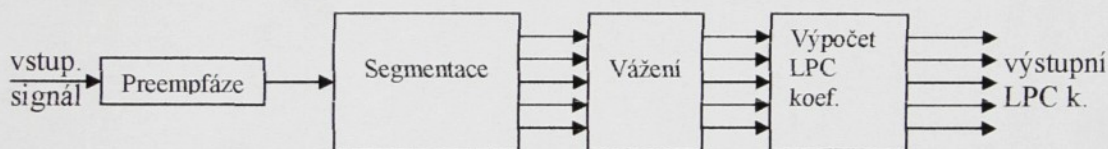
Při výběru metody syntézy byly nejdříve realizovány jednoduché vokodéry pro kepstrální a LPC metodu. Vokodér pro formantovou syntézu byl nalezen na internetu. Vokodér je v našem případě jednoduchý syntezátor, který převede vstupní řečový signál na jeho syntetickou podobu. Vstupní zvukový signál je nejprve analyzován (metoda analýzy závisí na metodě syntézy) a následovně syntetizován pomocí parametrů, které byly analýzou získány.



Obr 5.1 – *Blokové schéma vokodéru*

Blok analýzy se obecně nazývá blokem kódovacím a blok syntézy blokem dekódovacím, proto název vokodér. Vokodéry jsou v praxi používány například v telekomunikacích. Díky analýze vstupního signálu jsou získány parametry, kterých je méně než vzorků ve vstupním signálu (např. u LPC analýzy získáme pro jeden segment (160 vzorků) osm LPC koeficientů). Proto se vokodérů používá jako kompresních nástrojů, kdy vstupní signál je nejdříve převeden na parametry, poté tyto parametry jsou poslány po nízkokapacitním datovém médiu (např. telefoní linka) a následně dekódovány pomocí syntézy.

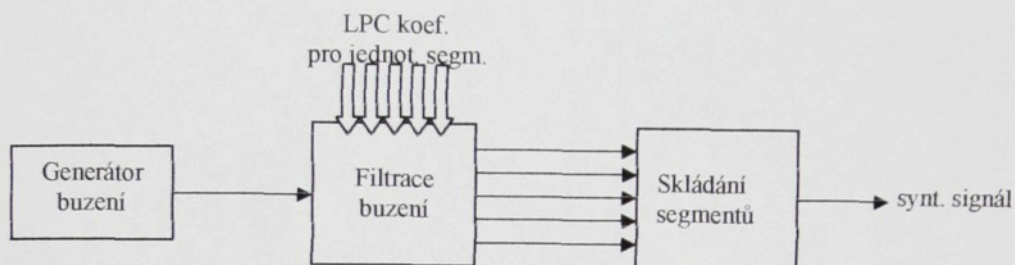
Vokodéry pro jednotlivé metody byly programovány v prostředí Matlab, jelikož nabízí velmi dobré nástroje pro zpracování signálu. Jako první byl vytvořen LPC vokodér. Ten je z hlediska obtížnosti realizace poměrně jednoduchý. Vstupní signál je nejprve upraven pomocí preempfáze pro zvýraznění vyšších frekvencí. Dále je signál rozdělen na jednotlivé segmenty. Počet vzorků segmentu odpovídá délce 20ms (což je 160 vzorků pro vzorkovací frekvenci 8kHz). Segmenty jsou dále váženy Hammingovým okénkem. Aby nedošlo k velké ztrátě dat díky okénkovací funkci, jsou použity segmenty s polovičním překryvem. V této fázi je signál (resp. jeho segmenty) připraven pro výpočet LPC koeficientů. Matlab nabízí funkci **aryule** pro výpočet LPC koeficientů, koeficientů odrazu a chyby lineární predikce, která byla použita. Tato funkce počítá LPC koeficienty pomocí Durbin-Lewisova algoritmu. Koeficientů je osm a jsou ukládány do maticové struktury $8 \times A$, kde A je celkový počet segmentů vstupního signálu. Takto je realizován blok analýzy signálu.



Obr 5.2 – Schéma bloku analýzy signálu (kodéru)

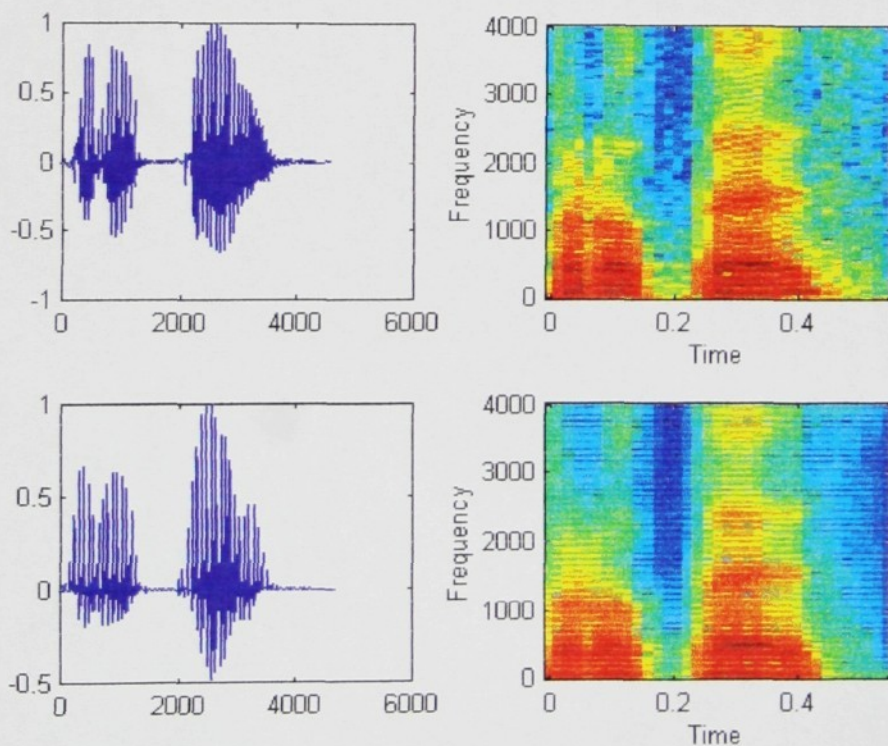
K syntéze výsledného signálu je třeba generovat budící signál. Pro jednoduchost bylo použito pouze buzení pulsy. Při generování neznělých hlásek je sice syntetizovaný signál horší kvality, ale tento vokodér sloužil pouze pro orientační posouzení možnosti této metody a k tomu toto zjednodušené buzení postačovalo. Budící signál byl tedy generován jako posloupnost jednotkových pulsů s amplitudou jedna a s frekvencí 100Hz. Pro začátek nebyla uvažována různá základní hlasová frekvence jednotlivých segmentů, která se v přirozené řeči vyskytuje. Budící signál jsem filtroval FIR filtrem, který lze v Matlabu realizovat např. pomocí funkce **filter** (ta byla i použita). Vstupními hodnotami do této funkce byly LPC koeficienty, budící signál a počáteční podmínky filtru. Výstupem pak byl syntetizovaný signál odpovídající jednomu segmentu, dle vstupních LPC koeficientů. Výsledný signál vznikl poskládáním segmentů s polovičním překrýváním. V praxi se ukázalo, že je vhodné částečně omezit vysokofrekvenční složky signálu, jelikož

jsme při analýze signálu tyto složky zvýraznili. Na výstup byl proto zařazen blok deemfáze, což je FIR filtr prvního řádu, pracující jako dolní propust'. Od aktuální výstupní hodnoty se odečte hodnota minulá. Filtr je však na mezi stability, a tudíž jsem hodnotu minulou násobil koeficientem 0,89.



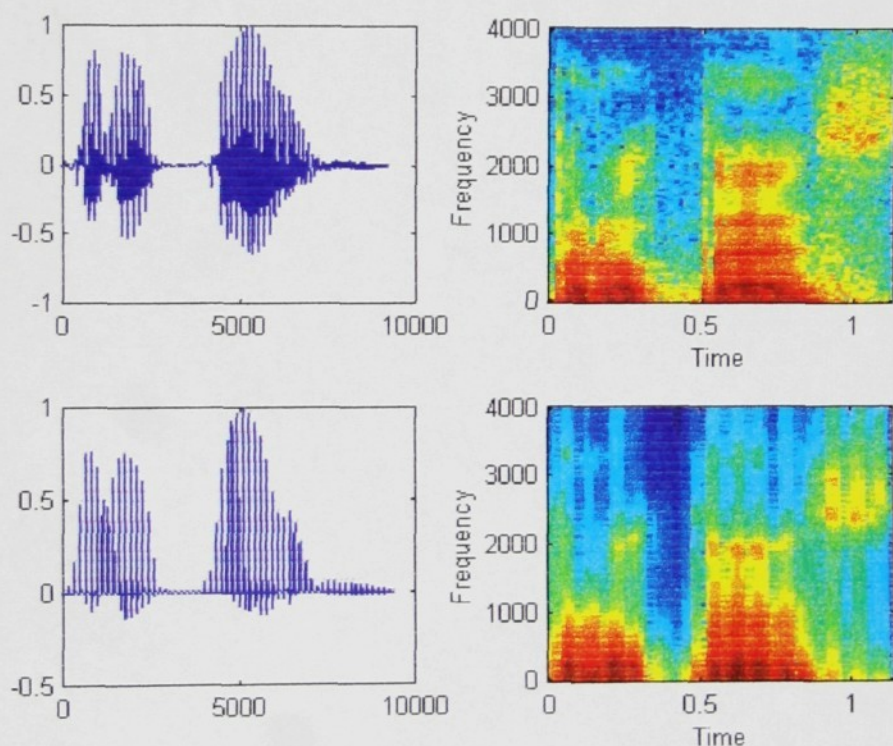
Obr 5.3 – Schéma bloku syntézy signálu (dekodéru)

V praxi bylo nutné prozkoumat mnoho faktorů, které mohou ovlivnit výslednou kvalitu syntézy a bylo třeba je experimentálně nastavit na optimální parametry.



Obr 5.4 – Průběh a spektrogram vstupního a syntetizovaného signálu při vzorkovací frekvenci 8kHz

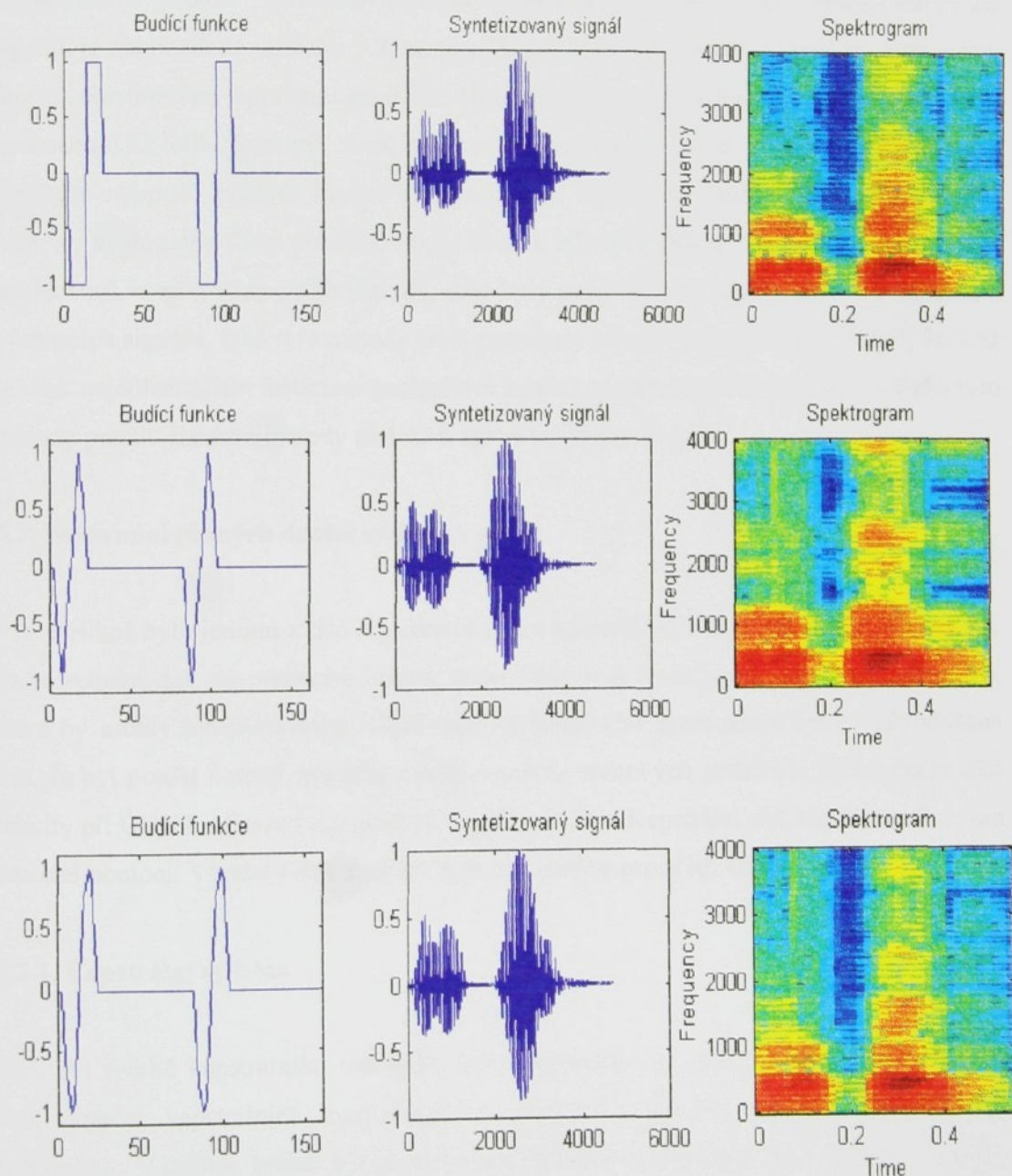
Prvním velmi závažným faktorem ovlivňující výslednou kvalitu byla vzorkovací frekvence. Při vzorkovací frekvenci 16kHz je syntetizovaný signál poměrně zkreslený a “zahuhňaný”. Proto byla použita vzorkovací frekvenci 8kHz, kdy byl výsledek nesrovnatelně lepší. Na obrázku 5.4 a 5.5 jsou zobrazeny pro porovnání výsledky syntézy slova “protest”. V horní části je znázorněn průběh a spektrogram vstupního signálu, dole signál syntetizovaný. Ostatní parametry vokodéru jsou pro oba obrázky stejné (bylo pouze potřeba změnit počet vzorků na segment, aby jeden segment odpovídal 20 ms signálu). Vstupní signál byl nahrán přes zvukovou kartu a filtrací odstraněn šum, se kterým byl signál nahrán. Takto upravený signál byl pak vstupem pro vokodér.



Obr 5.4 – Průběh a spektrogram vstupního a syntetizovaného signálu při vzorkovací frekvenci 16kHz

Z obou obrázků (hlavně u 5.4) je u spektrogramu syntetizovaného signálu zřejmá nespojitost mezi jednotlivými segmenty. To je způsobeno chybou lineární predikce. Dále je patrné, že syntéza při vzorkovací frekvenci 16 kHz rekonstruuje řečový signál hůře než při vzorkovací frekvenci 8 kHz. Tento problém se v současné době nepodařilo odstranit ani zdůvodnit (pouze částečně). Zároveň je ovšem ze spektrogramů vstupních signálů

patrné, že signál navzorkovaný při frekvenci 8 kHz nepopisuje dokonale některé úseky promluvy (např. neznělé úseky – šum má větší frekvenci než je polovina vzorkovací frekvence). Bylo by tedy lepší syntetizovat signál s vzorkovací frekvencí 16 kHz, bohužel to však zatím nevede k lepším výsledkům.



Obr 5.5 – Průběhy a spektrogramy syntetizovaného signálu pro různá buzení

Dalším faktorem, který ovlivňuje výslednou kvalitu signálu je budicí funkce. V literatuře se jako budicí funkce uvádí posloupnost jednotkových pulsů. Tyto pulsy však dodávají syntetizované řeči “kovový” zvuk, jelikož lidský trakt není schopen vytvořit dokonalý jednotkový puls. Proto bylo s budicí funkcí experimentováno, aby se její průběh přiblížil možnostem lidského hlasového traktu. Na obrázku 5.5 jsou zobrazeny průběhy signálů slova “protest“, syntetizovaných se vzorkovací frekvencí 8 kHz. Průběh vstupního signálu je zobrazen na obrázku 5.3, zároveň je na něm zobrazena i odezva na čistě pulsní buzení. Postupnými úpravami budicího signálu bylo vyvinuto buzení, které po syntéze neobsahuje již tolik “kovový” zvuk jako buzení klasické. Z buzení uvedených na obrázku 5.5 bylo nakonec použito buzení obdélníkovým signálem. Zároveň byly též měněny hodnoty výstupního filtru pro různé typy buzení. Hlavní kritériem při výběru buzení byla poslechová kvalita výstupního signálu, dále byly porovnávány spektrogramy vstupního a výstupních signálu. Obě tyto metody jsou metodami subjektivními a neexaktními. Jelikož je však nejdůležitějším kritériem poslechová kvalita syntetizovaného signálu, je třeba tyto metody použít. Exaktní metody nedokáží tuto kvalitu posoudit.

5.2. Porovnání různých druhů syntézy v praxi

Jelikož bylo jedním z cílů diplomové práce seznámit se s různými druhy syntézy jak na teoretické, tak na praktické úrovni, bylo třeba realizovat ještě další typy vokodérů, které by mohly být porovnány. Realizace formantového syntezaátoru byla zjednodušena tím, že byl použit hotový vokodér publikovaný na webových stránkách Elektrotechnické fakulty při ČVUT v Praze (viz. použitá literatura [19]). Kepstrální vokodér byl realizován bez cizí pomoci. Všechny tři vokodéry byly vyvinuty v prostředí Matlab.

5.2.1. Kepstrální syntéza

Při tvorbě kepstrálního vokodéru bylo vycházeno z teorie popsané v kapitole 3.5. Pro výpočet kepstrálních koeficientů byl nejprve upraven vstupní signál pomocí preemfáze. V dalším kroku byl signál rozsegmentován a jednotlivé segmenty váženy Hammingovým okénkem. Pro vlastní výpočet kepstrálních koeficientů pro jednotlivé vektory bylo použito blokové schéma, které je zobrazeno na obr. 3.5. Prostředí Matlab

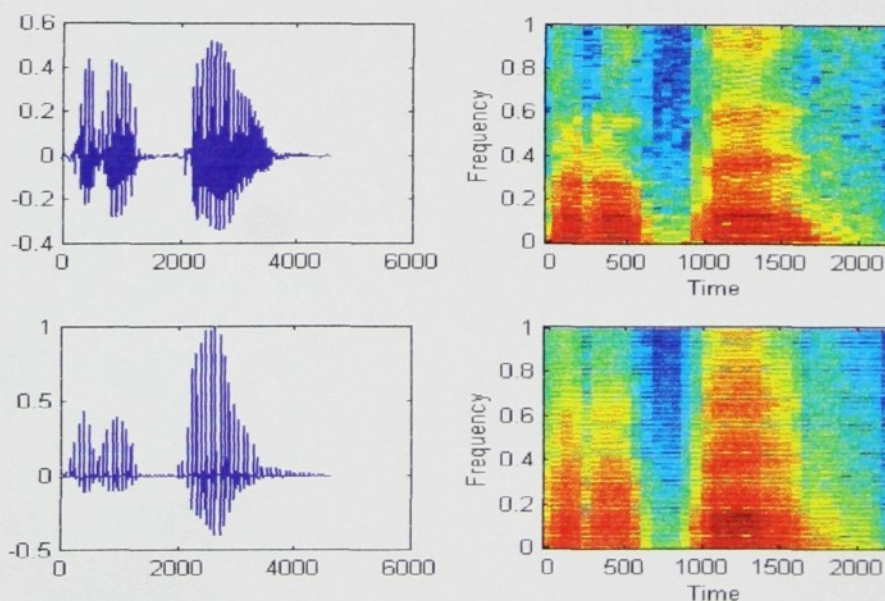
obsahuje funkce pro výpočet jednotlivých bloků. Počet kepstrálních koeficientů byl omezen na 25. Při aproximaci exponenciální funkce filtru byla použita Padého aproximace pro jednotlivé filtry v kaskádním zapojení. Přenosová funkce i -tého filtru po aproximaci byla tedy ve tvaru:

$$H_i = \frac{1 + a_i z^{-1} + b_i z^{-2i}}{1 - a_i z^{-1} + b_i z^{-2i}} \quad (25)$$

kde koeficienty a_i a b_i se vypočítají ze vztahů

$$\begin{aligned} a_i &= p_1 \cdot c_i \\ b_i &= p_2 \cdot c_i^2 \end{aligned} \quad (26)$$

Hodnoty konstant jsou dány z Padého aproximace a v tomto případě nabývají hodnot: $p_1 = 0,5$ a $p_2 = 21/128$. Hodnota c_i odpovídá i -tému kepstrálnímu koeficientu. Nultý filtr, vyjadřující energii signálu, odpovídá prostému násobení členem e^{C_0} . Realizace modelu hlasového traktu je tedy provedena kaskádním zapojením filtrů, kde přenos prvního je e^{C_0} , a ostatních podle rovnice (25).

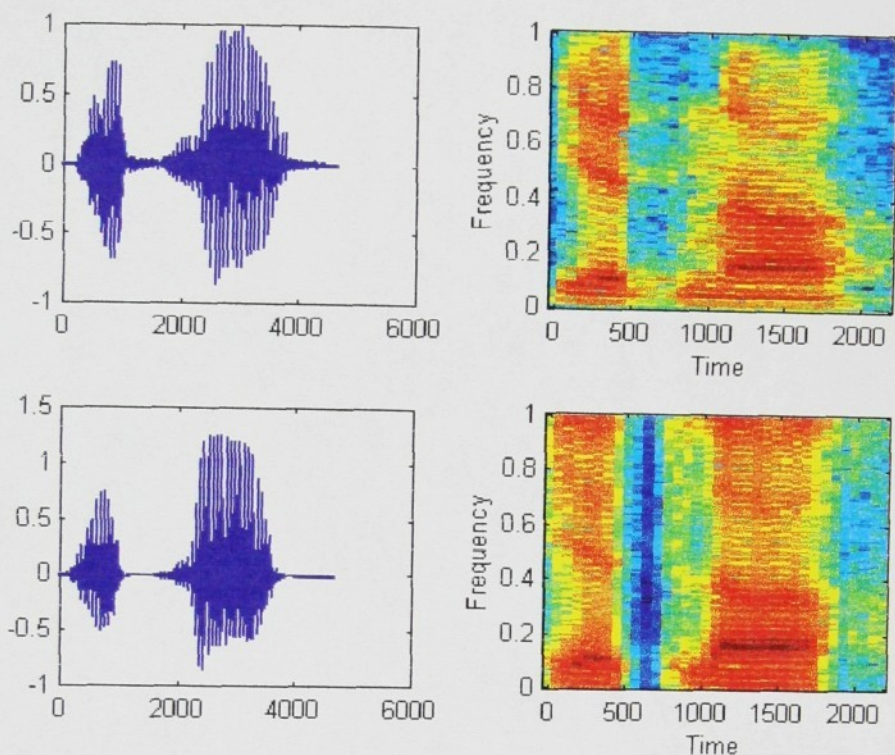


Obr 5.6 – Průběhy a spektrogramy vstupního a syntetizovaného signálu kepstrální syntézou

Takto realizovaný model byl buzen vstupní funkcí, která je posloupností jednotkových pulsů. Výstupní signál je upraven deempfází a dále normalizován (každý člen je vydělen maximální hodnotou signálu). Na obrázku 5.6 je průběh a spektrogram vstupního a syntetizovaného signálu takto naprogramovaného synteázatoru. Horní průběh a spektrogram odpovídá signálu vstupnímu a spodní signálu syntetizovanému. Opět se jedná o slovo "protest". Z obrázku 5.6 je patrné, že syntetizovaný průběh se zřetelně liší od průběhu vstupního signálu. U spektrogramu je podobnost větší, ale také je zřejmé, že syntetizovaný signál je víc "rozmazaný". Jednotlivé formantové frekvence nejsou zdaleka tak čitelné jako u spektrogramu signálu vstupního. Je tedy zřejmé, že pro praktické použití tohoto synteázatoru, by bylo nezbytné jeho vylepšení. Úkolem diplomové práce však byla realizace pouze jedné metody syntézy a jelikož bylo dosaženo lepších výsledků u LPC syntézy, byla zvolena tato. S keprstrálním modelem jsem tedy nedosáhl takových výsledků jako s LPC modelem, i když teoreticky by to mělo být naopak.

5.2.2. Formantová syntéza

Při zkoumání formantové syntézy v praxi byl použit již hotový program, sloužící jako učební pomůcka k předmětu Diskrétní zpracování řeči, vyučované na Fakultě elektrotechnické při ČVUT v Praze. Autorem tohoto programu je Petr Prášek. Jedná se o jednoduchý synteázator s jednou kaskádní větví (rezonátory jsou řazeny sériově). Na následujícím obrázku jsou průběhy a spektrogramy vstupního a výstupního signálu, syntetizovaného formantovou syntézou. Jedná se o slovo "jedna". Z průběhů i spektrogramů je patrné, že syntetizovaný signál neobsahuje úsek odpovídající hlásce "d". Při tvorbě této hlásky (zvláště není-li přímo vázána na samohlásku) jsou používány mluvní orgány (zuby, horní patro a okraj jazyka) k modifikaci výstupního proudu vzduchu a vznikají turbulence. Tyto turbulence však formantová syntéza nepostihuje, tudíž je výstupní signál skoro nulový. Při poslechu však není tento nedostatek tolik zřejmý. Dále jsou na výstupním spektrogramu velmi dobře znatelné jednotlivé formantové frekvence.



Obr 5.6 – Průběhy a spektrogramy vstupního a syntetizovaného signálu formantovou syntézou

Při hledání optimální metody syntézy byly tedy porovnány tři metody parametrické syntézy, založené na lineárním modelu hlasového traktu. Při realizaci syntézy bylo třeba měnit mnoho faktorů a vyzkoušet pro které hodnoty daných faktorů jsou akustické výsledky nejlepší. Z daných metod se jako nejvhodnější jevila LPC syntéza, jelikož byla nejlépe realizována. Formantová metoda byla po zhodnocení výsledků zavržena, jelikož úroveň syntetizovaného signálu nedosáhla takové úrovně jako metoda LPC koeficientů. Tento fakt byl předpokládán na základě prostudované teorie. Při porovnávání zbylých dvou metod bylo oproti očekávání dosaženo lepších výsledků u LPC syntézy, a proto byla pro realizaci zvolena.

6. Výběr stavebních jednotek

Po výběru a realizaci syntezátoru vyvstal další problém. Bylo třeba zvolit vhodný slovník stavebních jednotek, ze kterých výslednou promluvu skládáme. K výběru stavebních jednotek je třeba přistupovat velmi citlivě, jelikož na nich závisí jak kvalita výsledné promluvy, tak paměťová náročnost a složitost algoritmu výběru a skládání jednotek. Podle typu stavebních jednotek je též nutné přizpůsobit fonetickou transkripci. Při analýze jednotlivých možností popsaných v kapitole 4.2., se zrodila myšlenka pokusit se o nový přístup k tvorbě inventáře.

Inventáře popsané v kapitole 4.2. mají tu vlastnost, že pro všechny možnosti promluvy je výsledná kvalita stejná. V běžné promluvě se však některé promluvové úseky vyskytují poměrně často a některé naopak řidčeji. Bylo by tedy možné vyvinout inventář, který by postihoval právě tento jev, a pak by se tedy kvalita výsledných promluv lišila. Pro obvyklé promluvy by tato kvalita byla velmi dobrá, ovšem pro méně běžná slova by se kvalita zhoršila. Další vlastností tohoto inventáře by byl jev, že se délka (lexikální) jednotlivých jednotek liší (oproti fonémovému, difónovému a trifónovému inventáři). Pro český jazyk byly provedeny výzkumy a určeny nejčastěji se vyskytující slova (viz. použitá literatura [17]). Neredukovaný inventář složený ze slov, obsahuje stovky tisíc slov (zvláště pro český jazyk, kde se pro jedno slovo vyskytuje několik jeho tvarů). Redukci tohoto inventáře pomocí četnosti výskytu by bylo dosaženo několika desetitisíců jednotek. Představou však bylo vytvořit inventář s poměrně malým počtem jednotek, bylo tedy vhodné zvolit menší jednotku než celé slovo. Touto jednotkou jsou poloslabiky. Jejich neredukovaný inventář má velikost zhruba 7000 jednotek a jeho redukci by bylo možné dosáhnout zajímavých výsledků.

6.1. Vývoj použitého inventáře

Pro redukci poloslabikového inventáře byla nejdříve uvažována pouze metoda četnosti jednotlivých slabik. Pak ale byla zkoumána i míra koartikulace mezi jednotlivými fonémy ve slabice a dospělo se k závěru, že největší míra je mezi souhláskou a následující samohláskou. Tento poznatek je velmi dobře uplatnitelný při redukci inventáře, jelikož je možné inventář redukovat z tisíců jednotek na stovky jednotek. Jestliže je tedy redukován

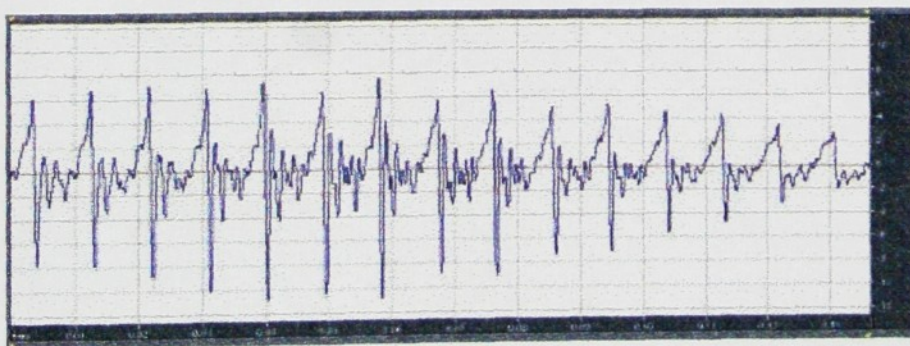
poloslabikový inventář pouze na poloslabiky tvořené jednou souhláskou a následující samohláskou získáme zhruba 350 jednotek. Z tohoto inventáře však není možné složit libovolnou promluvu, proto je potřeba doplnit jej o inventář fonémový. Tento inventář má skoro 400 jednotek a je z něj možno složit libovolnou promluvu s průměrnou kvalitou syntetizované řeči. Proto z něj bylo vycházeno, jen došlo k nutným úpravám, aby neobsahoval žádné přebytečné jednotky. Bylo tedy analyzováno několik textů (tisíce vět), aby byla zjištěna četnost výskytu jednotlivých spojení souhláska – samohláska. Na základě těchto výskytů byly z inventáře vyřazeny jednotky, které se v běžné promluvě nevyskytují, nebo vyskytují pouze výjimečně. Dále byly do inventáře zařazeny jednotky, které obsahují více souhlásek před samohláskou, jejich četnost je vysoká a ani koartikulace mezi souhláskami není zanedbatelná. Takto upravený inventář obsahuje 262 jednotek a je možno z něj složit jakoukoliv promluvu. Výpis všech jednotek inventáře je v příloze č.2.

6.1.1. Praktické vytvoření inventáře

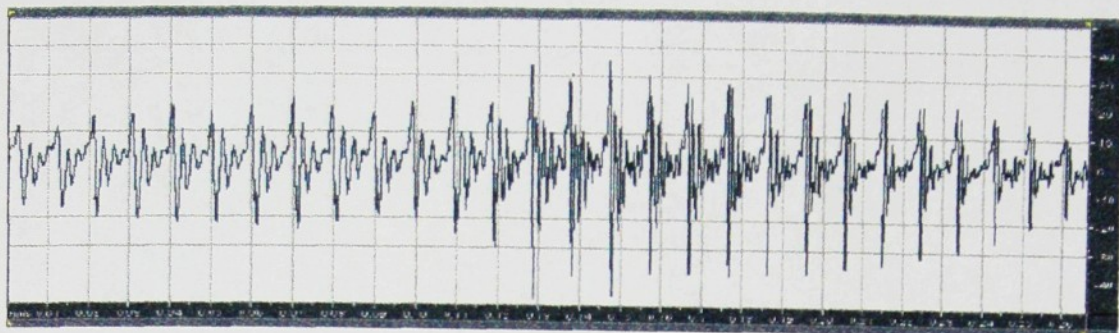
Vlastní tvorba inventáře je též záležitostí, ke které je třeba přistupovat velmi citlivě. Jestliže totiž bude inventář nahrán nekvalitně, pak i výsledný syntetizovaný signál bude se špatnou kvalitou, která bude navíc násobena chybou syntézy. Proto je třeba klást zvýšený důraz právě na kvalitu nahrávané inventární jednotky. Hlavní faktory ovlivňující kvalitu nahraného zvuku jsou: použitý mikrofon, zvuková karta, hluchnost a akustické vlastnosti prostředí, v němž je nahráváno. Mikrofon byl použit kvalitní magneto – dynamický, připojený přes zvukovou aparaturu, která se používá při ozvučení na amatérské úrovni. Pro naše účely disponuje velmi dobrou kvalitou výstupního signálu. Magneto – dynamický mikrofon snímá pouze zvuky generované ve své blízkosti, proto není výstupní signál tolik zatížen ruchem z okolí jako u mikrofonů běžných. Zvuková aparatura, sloužící v našem případě jako předzesilovač pro signál z mikrofону, je dobře odstíněna od rušivých signálů, které se mohou indukovat např. ze síťového napájení. Kabely mezi jednotlivými prvky jsou také stíněné. Jako zvuková karta byl použit SoundBlaster 128 PCI, který v praxi ukázal poměrně dobrou nahrávací kvalitu. Existují i kvalitnější zvukové karty, ale ty jsou samozřejmě ve vyšší cenové kategorii. Místem pro nahrávání byla zvolena velká rozlehlá místnost, umožňující eliminaci případných odrazů

zvukového signálu od blízkých ploch, čímž by mohlo dojít ke zkreslení signálu. Nakonec i přes poměrně dobré technické vybavení byla v signálu patrná šumová složka. Proto byly nejprve nahrány asi dvě vteřiny “ticha“, a teprve potom byl snímán řečový signál. Tím bylo dosaženo toho, že signál obsahoval první dvě vteřiny pouze šumovou složku, a ta mohla být analyzována a odfiltrována. Nástroj pro tuto redukci šumu obsahují některé programy pro editaci zvukových souborů. Touto filtrací byl dosažen poměrně kvalitní, šumem nezátížený řečový signál.

Další problém nastal při nahrávání konkrétní jednotky. Nejprve bylo sestaveno několik vět, které v sobě obsahovaly všechny jednotky inventáře. Jednotlivé věty byly namluveny a uloženy na disk počítače. Z těchto souborů byly pak pomocí programu pro editaci zvukových souborů vybírány jednotlivé úseky obsahující danou jednotku inventáře. Výběr byl prováděn na základě opticko – akustického vyhodnocení. Existují algoritmy pro automatický výběr jednotek (např. Viterbiho algoritmus), ale ty nedosahují takových výsledků jako opticko – akustická metoda. Tato metoda je však časově mnohem náročnější a její výsledky nejsou objektivní. Při sestavování promluvy z takto nahraných jednotek bylo zjištěno, že jednotlivé jednotky jsou natolik poznamenány prozodii jednotlivých slov a promluv ze kterých pocházely, že výsledná promluva, sestavená z těchto jednotek, je málo kvalitní. Jednotlivé jednotky se lišily intenzitou, délkou trvání a základní hlasovou frekvencí, takže některé z nich byly málo zřetelné, jiné zas se špatnou artikulací. Tudiž jednotky byly velmi nesourodé, a to se odrazilo i na syntetizovaném signálu. Proto byla vytvořena druhá verze inventáře, kde byly jednotlivé jednotky namlouvány jako samostatné celky se snahou o zřetelné vyslovení každé jednotky.



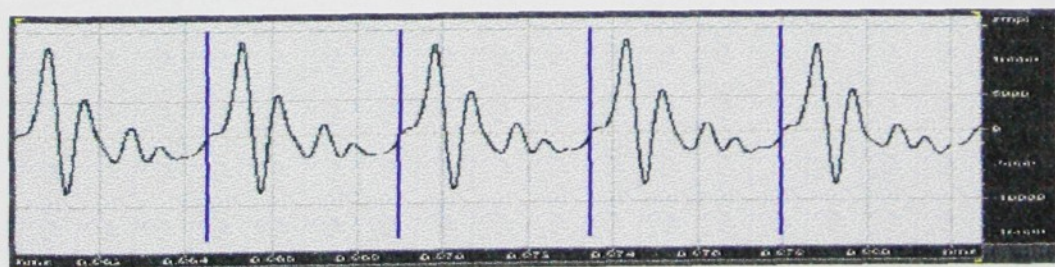
Obr 6.1 – Průběhy jednotky “la“ vytvořené vyjmutím z promluvy



Obr 6.2 – *Průběhy jednotky “la” vytvořené samostatně*

Rozdíly mezi stejnými jednotkami obou verzí inventářů demonstrují obrázky 6.1 a 6.2. Z nich je patrné, že jednotka z prvního inventáře (obr. 6.1) trvá kratší dobu a souhláska “l” je málo zřetelná a trvá pouze krátce. To je způsobeno tím, že část této souhlásky je obsaženo v předchozím fonému resp. v koartikulaci mezi nimi. Jednotka z druhého inventáře (obr. 6.2) je řečena zřetelněji, a proto je i syntetizovaná řeč srozumitelnější. V některých případech však může budit nespojitý dojem.

Jednotlivé jednotky druhého inventáře je třeba ještě upravit. Při nahrávání jednotlivých fonémů bylo třeba eliminovat hlasivkový ráz, který člověk při vyslovování samostatných hlásek vytváří. Dále bylo třeba odstranit začátek a konec jednotky, jelikož v běžné mluvě to není tak, že by jedna jednotka utichla a druhá začala, ale zvuk je souvislý a mezi jednotlivými jednotkami je spojitý přechod. Zároveň však je třeba vyhnout se při napojování jednotlivých segmentů vzniku nechtěných frekvencí. Ty vznikají při spojování úseků, které končí jinou hodnotou než začínají úseky, které napojujeme. Samozřejmě by měla být stejná směrnice v koncovém resp. počátečním bodě spojovaných úseků.

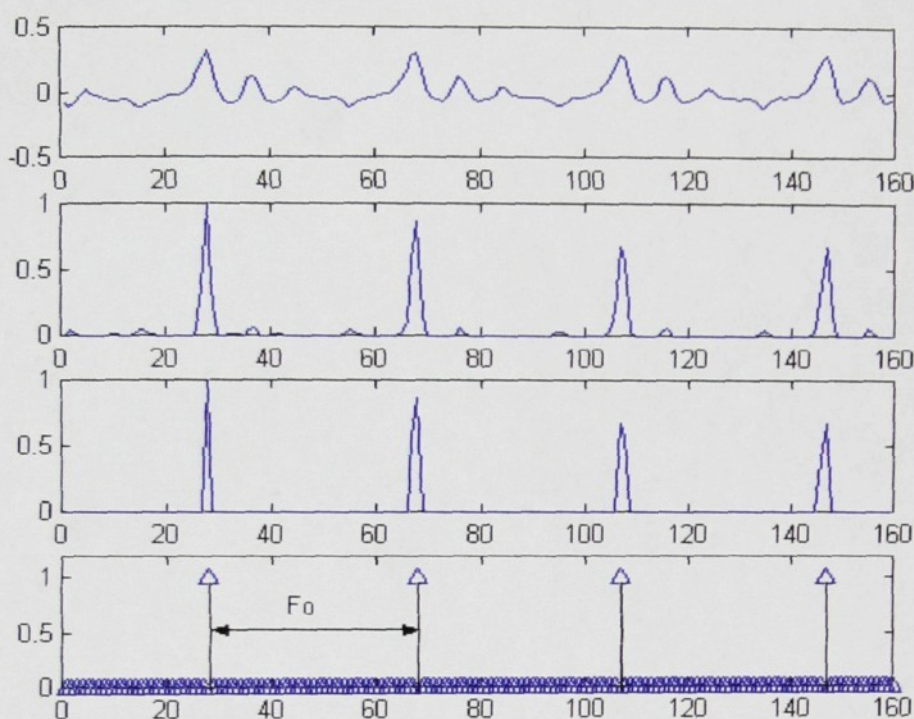


Obr 6.3 – *Průběhy řečového signálu s oddělením jednotlivých hlasových period*

To bylo vyřešeno tím, že při ořezávání počátku a konců jednotek bylo voleno místo ořezu tak, aby u znělých úseků jednotka začínala počátkem a končila koncem periody. Tím bylo dosaženo relativně spojitého průběhu při napojování segmentů. K vyhledávání začátků a konců periody bylo použito zobrazení průběhu signálu v čase. Způsob vizuálního nalezení jednotlivých hlasových period signálu je patrný z obr. 6.3. Je však zřejmé, že touto metodou nelze nalézt přesný konec či počátek periody, ale pochybení u odhadu zkušeného odborníka je poměrně malé.

6.2. Výpočet LPC koeficientů pro jednotlivé jednotky

Jednotlivé upravené jednotky byly uloženy do zvláštních souborů. Ty pak byly vstupem do programu, který jsem pro výpočet a uložení LPC koeficientů do kódové knihy vytvořil v programovacím prostředí Matlab. Tento program automaticky načítal jednotlivé jednotky. Pro každou provedl nejprve preempfázi, poté segmentaci s polovičním skládáním, poté vážení jednotlivých segmentů Hammingovým okénkem, výpočet LPC koeficientů a chyby lineární predikce. Dále byla zjištěna maximální absolutní hodnota signálu v daném segmentu a koeficient znělosti daného segmentu. Ten určuje zda je signál znělý či neznělý a případně jeho základní hlasovou frekvenci. Postup při nacházení základní hlasové periody znělého segmentu je znázorněn na obr. 6.4. Vstupní signál (první graf na obr. 6.4) je nejdříve normován a dále umocněn na třetí, aby velké hodnoty signálu byly zvýrazněny a nízké potlačeny (druhý graf na obr. 6.4). Dále jsou úplně eliminovány hodnoty signálu, které jsou menší než dvě pětiny maxima tím, že jsou položeny rovné nule (třetí graf na obr. 6.4). Nyní je třeba vyhledat v upraveném signálu lokální maxima. Z těch je pak určena základní hlasová perioda (čtvrtý graf na obr. 6.4). Jestliže nebylo možné periodu určit, pak je jako její hodnota dosazena nula a jedná se o neznělý úsek. Každá jednotka byla tedy rozložena na segmenty a pro ně bylo určeno osm LPC koeficientů, hodnota chyby lineární predikce, maximální hodnota průběhu segmentu a hodnota základní hlasové periody segmentu (nula v případě neznělého úseku).



Obr 6.2 – *Postup při nacházení základní hlasové periody*

6.3. Algoritmizace výběru a skládání jednotek

Slovník stavebních jednotek bylo třeba uspořádat tak, aby výběr jednotek probíhal co nejrychleji a nejefektivněji. Dále bylo třeba počítat s faktem, že při sestavování jednotlivých promluv neexistuje pouze jedna kombinace stavebních jednotek. Inventář totiž jednak obsahuje fonémy, z nichž lze sestavit jakoukoliv promluvu a dále i poloslabiky. Cílem je sestavit promluvu z co největších jednotek. Pro jednoduchou orientaci ve slovníku byly jednotky seříděny abecedně. Inventář byl rozdělen na dvě části, a ty pak opět abecedně seřazeny každá zvlášť. První část se skládala z fonémů a druhá z poloslabik. Jednotky se z inventáře vybírají pomocí cyklických průchodů slovníkem. Proto nejdříve procházíme druhou částí, a když není nalezena odpovídající poloslabika, tak přecházíme do části první. Tím je zajištěno, že je vybrána vždy nejdelší jednotka nalezená v inventáři. Tato struktura inventáře umožňuje též přidávání nových

jednotek do slovníku. Předpokládáme totiž, že nové jednotky budou lexikálně delší než jednotky stávající. Současná lexikální délka jednotek je pro fonémy jedna a pro poloslabiky obvykle dvě, ale může být i tři (např. “lou“). Nové jednotky je třeba srovnat podle jejich délky a je vhodné seřadit je i podle abecedy. Díky tomu, že lze snadno přidávat do inventáře nové jednotky, je možné zvýšit kvalitu syntetizované promluvy. Při konkrétní aplikaci syntezátoru lze totiž sestavit množinu často používaných promluv či promluvových úseků, a tu k stávajícímu inventáři přidat. Jednotky inventáře jsou reprezentovány čísly, lexikální hodnoty odpovídající jednotlivým číslům jsou uloženy v textovém souboru. Při výběru stavebních jednotek je tedy třeba převést vstupní text na posloupnost čísel, odpovídajících jednotlivým stavebním jednotkám dle seznamu uloženém v textovém souboru. Číselné označení jednotek usnadňuje určení stavebních prvků. Kdyby jednotky reprezentoval přímo jejich obsah, nastával by problém s rozdílnou délkou názvů a procházení polem by bylo též komplikovanější. Takto lze snadno procházet celý inventář pomocí **for** cyklu a navíc není třeba načítat celý slovník do pole. Zde je přímo hodnota proměnné ve **for** cyklu názvem jednotky. Jednotky v inventáři jsou tedy reprezentovány čísly a bez problému lze přidávat nové.

6.4. Program pro výpočet parametrů jednotek inventáře

V programovém prostředí Matlab byl vyvinul program, který načte jednotlivé zvukové soubory, v nichž jsou uloženy jednotky inventáře, a převede je na parametry potřebné pro syntézu. Vstupní zvukové soubory jsou již pojmenovány pomocí čísel, jak bylo popsáno v předchozí kapitole. Výstupem jsou čtyři soubory se stejným jménem jako vstupní soubor a s různými příponami. V souboru s příponou “.lpc“ jsou uloženy LPC koeficienty jednotlivých segmentů dané jednotky. Budeme-li tedy uvažovat, že daná jednotka má K segmentů, pak bude soubor obsahovat $8K$ čísel. Další soubor má příponu “.err“ a jsou v něm uloženy chyby lineární predikce pro jednotlivé segmenty, tedy K čísel. Soubory s příponami “.max“ a “.znl“ obsahují také každý K čísel a jedná se o maxima resp. znělosti jednotlivých segmentů.

7. Realizace vlastního TTS systému

Po výběru a vývoji LPC syntézy v prostředí Matlab bylo třeba vytvořit program, který by byl použitelný v hlasových informačních systémech. Pro realizaci jsem vybral programovací prostředí C++. V tomto prostředí byla vyvinuta konzolová aplikace, která měla za vstupní parametry text, jenž se má syntetizovat a číslo prosodického pravidla, které má být použito. Výstupem aplikace je syntetizovaný zvuk uložený v souboru wave. Aplikace kromě hlavního spustitelného souboru obsahuje adresář *pravidla*, ve kterém jsou uloženy soubory s jednotlivými prosodickými pravidly. Další adresář má název *tab* a obsahuje soubory s parametry pro syntézu jednotlivých stavebních jednotek. V hlavním adresáři aplikace jsou soubory *param.txt* a *seznam.txt*, kde první z nich obsahuje konfigurační parametry syntézy a druhý seřazený seznam stavebních jednotek.

Program je tvořen sedmi procedurami: *init*, *konverze*, *synt_frame*, *nacti_cihlu*, *prosody*, *uloz_wav* a *main*. Procedura *main* tvoří hlavní program. Pro přehlednost a usnadnění práce s proměnnými jsem definoval dvě důležité struktury. První byla nazvána *Brick* a odpovídá jedné stavební jednotce z používaného inventáře pro syntézu řeči. Tato struktura se skládá z pole proměnných typu *Frame* a z proměnné, která udává počet proměnných v tomto poli. Druhou strukturou je typ *Frame*, který obsahuje pole LPC koeficientů, proměnné pro hodnoty chyby lineární predikce, maxima a znělosti. Nejdůležitější proměnnou programu je pole *Bricků*, ve kterém jsou uloženy všechny parametry syntézy dané promluvy.

Vytvořená aplikace není "čistokrevným" TTS systémem, jelikož neobsahuje blok předzpracování textu a fonetické transkripce. Důvodem k tomu je skutečnost, že toto téma bylo již na naší fakultě realizováno. Vstupem je tedy již foneticky upravený text. Jednotlivé stavební jednotky mají však stejné označení jako je píšeme v textu, a proto je upravený vstupní text velmi podobný textu psanému. Názvy jednotlivých stavebních jednotek lze velmi snadno přepsat dle výstupu dané aplikace realizující fonetickou transkripci. Lze tedy prakticky použít jakýkoliv program, realizující předzpracování textu a fonetickou transkripci.

Při spuštění procedury *main* je nejdříve volána procedura *init*, která načte ze souboru *param.txt* parametry pro konfiguraci syntézy. Tento soubor obsahuje pět čísel. První určuje celkový počet jednotek v inventáři, druhé počet vzorků na jeden segment,

třetí počet LPC koeficientů pro jeden segment, čtvrté trvání pauzy, odpovídající mezeře ve vstupním textu (číslo odpovídá počtu segmentů obsahujících ticho) a poslední základní hlasové periodě syntetizovaného hlasu (základní hlasová perioda je udávána ve vzorcích signálu, kde vzorkovací frekvence je 8kHz). Tyto parametry jsou pak uloženy do globálních proměnných a je s nimi dále pracováno.

Jako první krok po inicializaci programu je převeden vstupní text na posloupnost stavebních jednotek. Tento převod je řešen v proceduře *konverze*, kde je nejdříve načten seznam všech jednotek ze souboru *seznam.txt* a poté je aktuální část textu porovnávána s jednotkami v seznamu. Seznam je při hledání odpovídající jednotky procházen odzadu, jelikož je uspořádán od nejmenších jednotek k největším a my se snažíme vybrat vždy co největší odpovídající jednotku. Z textu vezme od aktuální pozice určitý počet písmen, odpovídající délce aktuální jednotky a porovná se s jednotkou. V případě shody je do výstupní posloupnosti přidáno číslo odpovídající dané jednotce a pozice v textu je posunuta dál o počet písmen odpovídající jednotky.

Posloupnost čísel, která je výstupem z procedury *konverze*, je převedena na pole *Bricků* pomocí procedury *nacti_cihlu*. Vstupem této procedury je číslo odpovídající stavební jednotce a výstupem je proměnná typu *Brick*. Naplnění výstupní proměnné je provedeno hodnotami ze souborů obsahující parametry pro danou jednotku. Tyto soubory jsou uloženy v adresáři *tab* a mají název stejný jako je číslo vstupující do procedury a jejich přípony jsou *“.lpc“*, *“.err“*, *“.max“* a *“.znf“*. Nejdříve je zjištěn a uložen do výstupní struktury počet segmentů dané jednotky, a poté jsou jednotlivé *Framy* z pole obsaženého ve výstupní struktuře naplněny hodnotami parametrů. Jestliže vstupnímu číslu odpovídá znak mezery, pak jsou jednotlivé parametry *Framů* naplněny nulami a jejich počet odpovídá proměnné načtené ze souboru *param.txt* při inicializaci. Volání této procedury je prováděno cyklicky ve **for** cyklu odpovídajícímu počtu jednotek, ze kterých byla sestavena promluva v proceduře *konverze*. Výstupní hodnoty procedury *nacti_cihlu* jsou ukládány do globálního pole typů *Brick*.

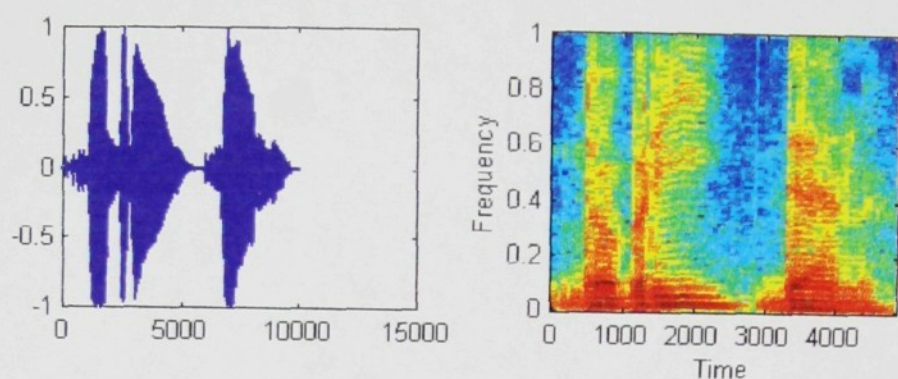
Další volanou procedurou je *prosody*. Ta je používána pouze v případě, že počet vstupních parametrů je větší než jedna. Jejím vstupním parametrem je číslo prozodického pravidla, které má být na výslednou promluvu aplikováno. Tvorba prozodie je omezena pouze na modelování průběhu základní hlasové frekvence. V souboru, kde je uloženo pravidlo, je tedy posloupnost čísel odpovídajících průběhu základní hlasové periody

(hodnota je udávána v počtu vzorků na jednu periodu.) . Jednotlivá čísla odpovídají hodnotám periody pro jednotlivé segmenty. Jelikož se jedná o pravidla popisující prozodii jednotlivých typů vět, jsou aplikována od konce. Hodnoty periody jsou přepisována v globálním poli *Bricků*.

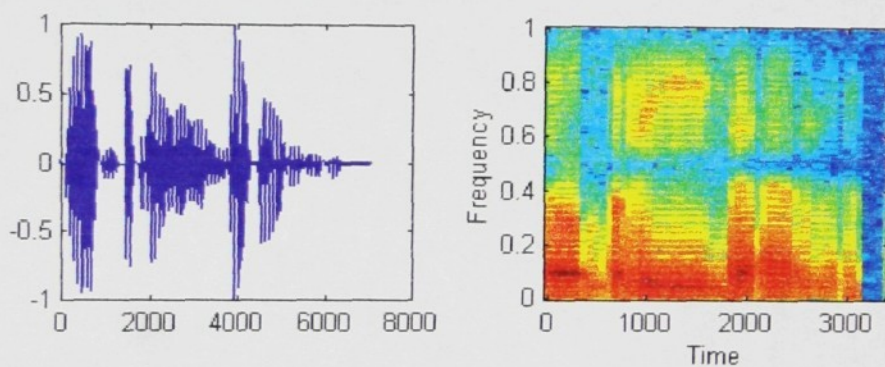
Hlavní částí programu je procedura zajišťující vlastní syntézu a tou je *synt_frame*. Tato procedura má jako své vstupní parametry pole LPC parametrů, chybu lineární predikce, maximum signálu, parametr znělosti a koncové hodnoty předchozího syntetizovaného signálu. Jejím výstupem je pak syntetizovaný signál. Syntetizovaný signál je vypočítán průchodem budícího signálu číslicovým filtrem, který je vyjádřen diferenční rovnicí. Budící signál je vytvářen buď jako posloupnost jednotkových pulsů šířky osm vzorků a periodou odpovídající hodnotě znělosti, a nebo, je-li znělost rovna nule, jako šumový signál. Při průchodu budícího signálu filtrem jsou jako počáteční podmínky brány koncové hodnoty předchozího syntetizovaného segmentu. Jelikož procedura realizuje syntézu pouze jednoho segmentu, obsahuje hlavní program dva **for** cykly. Jeden prochází jednotlivé *Bricky* a druhý v každém *Bricku* jeho *Framy*. Jednotlivé syntetizované segmenty jsou skládány s polovičním překrýváním.

Složený výstupní signál je normalizován a pomocí procedury *ulož_wav* je zapsán do výstupního souboru *out.wav*. Tato procedura generuje hlavičku wave souboru a spolu s průběhem syntetizovaného signálu ji binárně uloží do výstupního souboru. Tento soubor pak lze přehrát v libovolné aplikaci pro přehrávání zvukových souborů (např. Windows MediaPlayer).

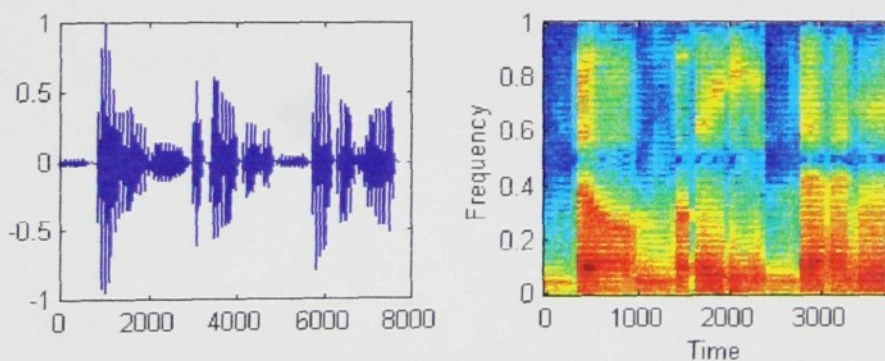
Pro porovnání výstupního signálu zde uvádím dva výstupní signály téže promluvy pro různé stavební jednotky. Na obr. 7.1 je zobrazen průběh a spektrogram slov “dobrý den“. Tyto dvě slova jsem rozdělil a vytvořil z nich dvě tavební jednotky. První obsahovala slovo “dobrý“ a druhá “den“. Syntetizovaný signál složený z těchto dvou stavebních jednotek je na obr. 7.2. Obr. 7.3 popisuje syntetizovaný průběh složený ze základních jednotek mého inventáře. Promluva je složena z jednotek “do“, “b“, “rý“, “de“ a “n“.



Obr 7.1 – Průběh a spektrogram promluvy “dobry den”



Obr 7.2 – Průběh a spektrogram syntetizované promluvy “dobry den”



Obr 7.3 – Průběh a spektrogram syntetizované promluvy “dobry den” složené pomocí základního inventáře

Na spektrogramu na obr. 7.3 jsou dobře patrné jednotlivé stavební jednotky. Spektrogramy syntetizovaných signálů se znatelně liší od spektrogramu signálu přirozené řeči. To je dáno tím, že jako budící funkce je použit jednotkový puls šířky osm vzorků a ne standardní, který má šířku jeden vzorek. Širší budící signál je použit, protože zvyšuje poslechovou kvalitu signálu a hlas nemá "kovové" zabarvení.

Výsledný program lze jednoduše přepsat jako dynamicky linkovanou knihovnu a používat v různých praktických systémech. Lze ho zakomponovat i ve stávající exe formě. Aplikace může pracovat pod operačními systémy Dos a Windows. Díky rychlé časové odezvě jej lze použít jak v off-line, tak v on-line systémech. Základní parametry programu lze jednoduše měnit v textovém souboru *param.txt*. Pro paměťovou a algoritmickou nenáročnost může pracovat i na méně výkonnějších počítačích.

8. Závěr

V úvodu teoretické části této práce je popsána realizace řeči v lidském hlasovém traktu, jelikož řada metod syntézy řeči se snaží tento proces modelovat. Také jsou stručně nastíněny možnosti využití lidské řeči jako komunikačního kanálu mezi člověkem a strojem. Hlavní náplní teoretické části je podrobnější popis všech základních metod syntézy řečového signálu s cílem nalézt tu nejvhodněji realizovatelnou pro výslednou aplikaci. Právě tuto část považuji za jeden z přínosů práce, protože jsem při studiu nenarazil na žádnou literaturu, v níž by byly podrobněji popsány více jak dvě metody syntézy. Čtenář této práce tak může všechny popsané metody prostudovat a utvořit si vlastní názor, usnadňující výběr metody syntézy pro konkrétní aplikaci. Dále jsou popsány jednotlivé části TTS systémů a jejich realizace. Více prostoru je věnováno modelování prosodie, jelikož se však jedná o velmi rozsáhlý okruh problémů, jsou nastíněny pouze základy.

Praktická část této práce se věnuje popisu vývoje a realizace výsledné aplikace. První kapitola je zaměřena na vývoj vlastního bloku syntézy, na experimenty a problémy, které během realizace vyvstaly. Druhá kapitola popisuje vývoj inventáře stavebních jednotek, ze kterých je výsledná promluva sestavována. Zde spatřuji další z přínosů této práce, jelikož z velmi malého počtu stavebních jednotek lze sestavit poměrně kvalitní promluvu. Navíc lze inventář rozšiřovat o další stavební jednotky a tím i zkvalitňovat promluvu v konkrétních aplikacích. Poslední kapitola popisuje výslednou aplikaci.

Můj úkol vytvořit jednoduchou aplikaci TTS systému byl poměrně složitý, jelikož na naší fakultě nemáme s tímto problémem praktické zkušenosti. Proto jsem ke svým závěrům dospěl vlastním výzkumem a studiem velkého množství materiálů. Výsledná kvalita syntetické řeči mé aplikace je znatelně horší než u jiných TTS systémů. Ty však vznikaly po dobu několika let v kolektivech špičkových odborníků, a proto je těžké je srovnávat s mou aplikací. Problém syntézy řeči je velmi rozsáhlý, a proto jsem se snažil vyvinout moji aplikaci tak, aby bylo možné v práci pokračovat, dále ji rozvíjet a zlepšovat možnosti použití v praxi.

Seznam použité literatury

- [1] Psutka, J.: Komunikace s počítačem mluvenou řečí. Academia. Praha 1995.
- [2] Krčmová, M.: Fonetika a fonologie českého jazyka. [skriptum] Fakulta filozofická UJEP. Brno 1982.
- [3] Přibil, J.: Syntezátory řeči pro TTS systémy. Ústav radiotechniky a elektroniky AV ČR, Praha 1995.
- [4] Přibil, J.: Základní principy a metody syntézy řeči. In: Počítačové zpracování řeči. Nouza, J. TU Liberec 2001.
- [5] Přibil, J.: TTS systémy. In: Počítačové zpracování řeči. Nouza, J. TU Liberec 2001.
- [6] Nejedlová, J., Volejník, M.: Transkripce psaného českého textu do fonetické podoby. In: Počítačové zpracování řeči. Nouza, J. TU Liberec 2001.
- [7] Nouza, J.: Řeč jako významný prostředek komunikace. In: Počítačové zpracování řeči. Nouza, J. TU Liberec 2001.
- [8] Myslivec, M.: Parametrizace řečového signálu pro účely rozpoznávání. In: Počítačové zpracování řeči. Nouza, J. TU Liberec 2001.
- [9] Ptáček, M.: Vybrané statě z akustiky řeči. Fonetický ústav KLF FF UK, Praha 1993.
- [10] Dohalská – Zichová, M.: Dynamika verbální komunikace – Mluvené slovo v teorii a praxi. Univerzita Karlova. Praha 1990
- [11] Uhlíř, J., Sovka, P.: Číslicové zpracování signálů. ČVUT. Praha 1995.
- [12] Oppenheim, A.V., Schafer, R.W.: Discrete-time signal processing. Prentice Hall. New Jersey 1989.
- [13] Horák, P.: Modelování suprasegmentálních rysů mluvené češtiny pomocí lineární predikce. [Disertační práce]. Praha 2002. ČVUT FEL.
- [14] Černocký, J.: Diskrétní model pro analýzu a syntézu řeči, LPC. FIT VUT. Brno.
- [15] Považanec, D., Horák, P., Fuchs, P.: Syntezátor ľudskej reči metódou TD-PSOLA. Slovenská technická univerzita. Bratislava 1999.

- [16] Vondra, M.: Kepstrální analýza řečového signálu. In: Elektrorevue 2001/48.
www.elektrorevue.cz
- [17] Frekvence slov, lemmat.
<http://www.fi.muni.cz/nlp/NlpCz/Frekvence%20slov,%20lemmat.html>.
- [18] SAMPA pro češtinu. <http://noel.feld.cvut.cz/sampa>.
- [19] Prášek, P.: Formantová syntéza. FEL ČVUT. Praha 2001.
<http://noel.feld.cvut.cz/vyu/dzr/dzr12/>

Seznam příloh

- A Česká SAMPA – přehled fonémů vyskytujících se v českém jazyce
- B Přehled jednotek použitého inventáře
- C Popis přiloženého disku CD-ROM

PŘÍLOHA A

Czech SAMPA

SAMPA SYMBOL	ORTHOGRAPHIC TRANSCRIPTION	SAMPA TRANSCRIPTION	ENGLISH TRANSLATION
Vowels			
i	myš	miš	mouse
e	les	les	forest
a	pas	pas	passport
ɔ	rok	rek	year
u	kus	kus	piece
i:	pít	pi:t	to drink
e:	lék	le:k	drug
a:	rád	ra:t	glad
ɔ:	móda	mo:da	fashion
u:	půl	pu:l	half
diphthongs			
ɔ_u	mouka	mo_uka	flour
a_u	auto	a_uto	car
Consonants			
plosives			
p	pes	pes	dog
b	bota	bota	shoe
t	tam	tam	there
d	dům	du:m	house
c	tito	cito	these
J\	děd	J\et	grandfather
k	krk	krk	neck
g	kde	gde	where
affricates			
t_s	cíl	t_si:l	aim
d_z	lekdý	led_zgdí	at times
t_s	čas	t_sas	time
d_z	léčba	le:d_Zba	medical treatment
fricatives			
f	forma	forma	form
v	vak	vak	bag
s	sen	sen	dream
z	zub	zup	tooth
Q\	tri	tQ\i	three
P\	rád	P\ a:t	order
š	šaty	Sati	clothes
ž	žal	Zal	regret
j	jas	jas	brightness
k	chata	xata	cottage
h\	had	h\at	snake
liquids			
r	ret	ret	lip
l	led	let	ice
nasals			
m	nák	ma:k	poppy
n	noc	not_s	night
ŋ	banka	baŋka	bank
J	nic	Jit_s	nothing

PŘÍLOHA B

1. a	45. bou	89. gou	133. mé	177. re	221. t'e
2. á	46. ca	90. gau	134. mi	178. ré	222. t'i
3. b	47. ce	91. ha	135. mí	179. ri	223. t'í
4. c	48. ci	92. há	136. mo	180. rí	224. t'o
5. č	49. cí	93. he	137. mu	181. ro	225. t'u
6. d	50. co	94. hé	138. mŭ	182. ru	226. va
7. ď	51. cu	95. hi	139. mou	183. rŭ	227. vá
8. e	52. cŭ	96. hí	140. na	184. rou	228. ve
9. é	53. cou	97. ho	141. ná	185. řa	229. vé
10. f	54. ča	98. hu	142. ne	186. řá	230. vi
11. g	55. čá	99. hŭ	143. né	187. ře	231. ví
12. h	56. če	100. hou	144. ní	188. ři	232. vo
13. ch	57. či	101. ja	145. ní	189. ří	233. vu
14. i	58. čí	102. já	146. no	190. řo	234. vŭ
15. í	59. čo	103. je	147. nu	191. řou	235. vou
16. j	60. ču	104. ji	148. nŭ	192. sa	236. za
17. k	61. čŭ	105. jí	149. nou	193. sá	237. zá
18. l	62. čou	106. jo	150. ňi	194. se	238. ze
19. m	63. čau	107. ju	151. ňí	195. si	239. zé
20. n	64. da	108. jou	152. ňo	196. sí	240. zi
21. ň	65. dá	109. jau	153. ňu	197. so	241. zí
22. o	66. de	110. ka	154. ňou	198. su	242. zo
23. ó	67. dé	111. ká	155. ňe	199. sŭ	243. zu
24. p	68. di	112. ke	156. pa	200. sou	244. zŭ
25. r	69. dí	113. ké	157. pá	201. ša	245. zou
26. ř	70. do	114. ki	158. pe	202. šá	246. ža
27. s	71. du	115. kí	159. pé	203. še	247. žá
28. š	72. dů	116. ko	160. pi	204. šé	248. že
29. t	73. dou	117. ku	161. pí	205. ši	249. ži
30. t'	74. d'i	118. kŭ	162. po	206. ší	250. ží
31. u	75. d'í	119. kou	163. pó	207. šo	251. žo
32. ú	76. fa	120. la	164. pu	208. šu	252. žu
33. v	77. fá	121. lá	165. pŭ	209. šou	253. žou
34. z	78. fe	122. le	166. pou	210. ta	254. cha
35. ž	79. fé	123. lé	167. pra	211. tá	255. chá
36. ou	80. fi	124. li	168. prá	212. te	256. che
37. au	81. fo	125. lí	169. pro	213. té	257. chi
38. ba	82. fu	126. lo	170. přa	214. ti	258. chí
39. bá	83. fou	127. lu	171. přá	215. tí	259. cho
40. be	84. ga	128. lŭ	172. pře	216. to	260. chu
41. bi	85. ge	129. lou	173. při	217. tu	261. chŭ
42. bí	86. gi	130. ma	174. při	218. tŭ	262. chou
43. bo	87. go	131. má	175. ra	219. tou	
44. bu	88. gu	132. me	176. rá	220. t'a	

PŘÍLOHA C

Popis přiloženého disku CD-ROM

Přiložený CD-ROM obsahuje následující programy:

- LPC vokodér
- keprstrální vokodér
- program pro realizaci kódové knihy pro LPC syntézu
- jednotky pro tvorbu kódové knihy nahrané ve zvukových souborech
- výslednou aplikaci – program Mluv

LPC vokodér

Na disku CD-ROM je uložen v adresáři *LPC_vokoder* a zdrojový kód je realizován v programovém prostředí Matlab. Vstupem programu je zvukový soubor typu wave. Syntetizovaný signál je reprodukován přes zvukovou kartu počítače. Dále je zobrazen průběh a spektrogram vstupního a výstupního signálu.

Kepstrální vokodér

Na disku CD-ROM je uložen v adresáři *Kep_vokoder* a je také realizován v prostředí Matlab. Vstupy a výstupy jsou realizovány stejně jako u LPC vokodéru.

Program pro tvorbu kódové knihy pro LPC syntézu

Na disku CD-ROM je uložen v adresáři *LPC_kniha* v souboru *Make_LPC.m*. Program je vytvořen v prostředí Matlab. Tento program převede stavební jednotky uložené ve zvukových souborech wave na parametry potřebné při LPC syntéze výsledné aplikace. Program používá externí funkce *znelost.m* a *korelace.m* pro výpočet znělosti segmentu. Zvukové soubory jsou uloženy s názvem odpovídajícím číslu dané jednotky v seznamu. Výsledná tabulka LPC koeficientů, maxim, znělosti a chyb lineární predikce je uložena v souboru *tab.mat*. Obsah tohoto souboru je převeden pomocí programu *prevod.m* do formátu ASCII. Výstupem jsou pak čtyři soubory obsahující LPC

koeficienty, chybu lineární predikce, znělost a maximum a nesoucí název podle dané jednotky. Tyto soubory pak používá hlavní aplikace.

Jednotky pro tvorbu kódové knihy

Jsou uloženy v adresáři *Jednotky*, ve zvukových souborech typu wave. V tomto adresáři se nachází i jejich seznam. Každé jednotce odpovídá číslo, pod kterým je uložena v souboru.

Koncová aplikace – program Mluv

Vlastní program se nachází v adresáři *Mluv* a skládá se ze souborů *mluv.exe*, *param.txt* a *seznam.txt* a adresářů *tab* a *pravidla*. V adresáři *tab* jsou uloženy parametry všech jednotek pro syntézu a v adresáři *pravidla* jsou uložena jednotlivá prozodická pravidla. Program by měl být spouštěn v příkazové řádce s parametry. První parametr je text promluvy, kterou chceme syntetizovat. Druhý parametr je číslo prozodického pravidla, které chceme použít. Tento parametr je volitelný. Volání programu tedy může vypadat např. takto: *mluv.exe "má ema maso" 02*. Vstupní text je třeba předzpracovat pomocí fonetické transkripce. Výstupem je soubor *out.wav*, který obsahuje výslednou promluvu. Zdrojové kódy k tomuto programu jsou uloženy v adresáři *Mluv_kod*.