Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Prispôsobenie tempa zaznamenanej reči

Diplomová práca

2016

Bc. Rudolf Krumpál

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Prispôsobenie tempa zaznamenanej reči

Diplomová práca

Študijný program: Aplikovaná informatika
Študijný odbor: 2511 Aplikovaná informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: RNDr. Marek Nagy, PhD.

Bratislava, 2016
Bc. Rudolf Krumpál

48495254

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Rudolf Krumpál
Študijný program: aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: aplikovaná informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Prispôsobenie tempa zaznamenanej reči
Rate adaptation of recorded speech

Cieľ: Každá reč sa vyznačuje svojím tempom, ktoré možno reprezentovať počtom slabík za sekundu. Cieľom práce je vytvoriť automatickú konverziu tempa bez straty identity rečníka.
Vstupom je text a jeho doslovná nahrávka, čo umožňuje jednoduchý prepočet koeficientu slabík za sekundu. Nahrávka prečítaného textu môže byť rýchla, alebo pomalá.

Navrhnutý postup zmeny tempa bude zakomponovaný do Multimediálnej čítanky. Deti v rámci hodiny počúvajú a sledujú čítaný príbeh. V záujme zachovania individuálneho prístupu bude možné deťom nastaviť primerané tempo.

Poznámka: Octave, Linux

Vedúci: RNDr. Marek Nagy, PhD.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: prof. Ing. Igor Farkaš, Dr.

Dátum zadania: 16.09.2014

Dátum schválenia: 26.11.2014 prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

Pod'akovanie

Rád by som veľmi pekne poďakoval môjmu vedúcemu RNDr. Marek Nagy, PhD. za jeho pomoc a rady.

Abstract

This thesis sumarizes the research, which dealt with analysis and synthesis of recorded speech. With help of speech analysis and synthesis we slowed down speech rate maintaining talker's identity. At first we deeply analysed digital signal of recorded speech. We used several techniques accompanying statistic functions. Such as cross correlation, zero-crossing rate, signal energy and other functions. After analysis we synthetised speech using pitch enhanced synchronous overlap and add algorithm.

Keywords: speech, digital signal, cross correlation, zero-crossing rate, adjusting rate of speech, PSOLA reč, digitálny signál, autokorelácia, zero-crossing rate, nastavovanie tempa reči, PSOLA

Abstrakt

V tejto práci je zhrnutý výskum riešenia problému, ktorý sa zaoberal analýzou a syntézou digitálneho signálu reči. Cieľom tejto práce bolo pomocou analýzy a syntézy zaznamenanej reči, spomaliť rýchlosť reči a zachovať pri tom identitu čitateľa. Problém sme riešili podrobnou analýzou vstupu pomocou rôznych funkcií. Použité funkcie boli autokorelácia, zero-crossing rate, energia signálu a iné. Následne sme signál syntetizovali pomocou vylepšeného PSOLA algoritmu.

Kľúčové slová: reč, digitálny signál, autokorelácia, zero-crossing rate, nastavovanie tempa reči, PSOLA

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

Predložená diplomová práca je súčasťou väčšieho projektu multimediálna čítanka. Tento projekt slúži deťom ako pomôcka pri učení čítania. Jednou časťou tohto projektu je cvičenie, kde deti počúvajú predčítanú nahrávku a na obrazovke sledujú jej text. Zvýrazňujú sa im slová, ktoré sú práve predčítavané. Niektoré deti toto cvičenie nezvládali, lebo text bol čítaný prirýchlo. Cieľom našej práce bolo predčítané nahrávky spracovať a spomaliť. Pričom sa mala zachovať identita rozprávača. Teda hlas rozprávača nemal byť zmutovaný. V nasledujúcich kapitolách sa čitateľ dočíta, ako sme reč analyzovali a potom syntetizovali, aby sme sa dopracovali k výsledným spomaleným nahrávkam, ktoré poskytneme žiakom využívajúcim túto čítanku.

V kapitole 1 najprv zadefinujeme základné pojmy, ktoré sú pre pochopenie našej práce nevyhnutné. V podkapitolách tejto kapitoly najprv zadefinujeme reč, následne digitálny signál a napokon digitálny signál reči a jeho vlastnosti. Potom popíšeme pracovné prostredie, ktoré sme používali. Neskôr príde na rad teória skrývajúca sa za štatistickými funkciami a algoritmami, ktoré sme v našej práci využili. Ako normalizácia signálu, autokorelácia, zero-crossing rate, PSOLA a iné.

V kapitole 2 popisujeme naše riešenie problému praktickým využitím teórie z prvej kapitoly. Tiež popisujeme naše vylepšenia algoritmov, respektíve správne využitie štatistických funkcií aby sme sa dopracovali k správnym výsledkom.

V kapitole 3 porovnáme naše riešenie, s voľne dostupným softwarom. Popíšeme chyby, ktoré sme v software objavili pomocou reverzného inžinierstva a popíšeme iné možné riešenie časti problému, ktorý sme spracovávali v tejto práci.

V kapitole 4 zosumarizujeme naše výsledky.

Cieľom tejto práce je vytvoriť nástroj, ktorý bude študentom poskytovať možnosť spomaliť rýchlosť s akou je im text predčítavaný. A to tak, aby sme zachovali identitu čitateľa.

Kapitola 1

Prehľad problematiky

1.1 Motivácia

Úvodom by sme radi vysvetlili našu motiváciu a problematiku v ktorej sa budeme pohybovať. Hlavnou motiváciou vzniku tejto práce je rozšírenie už existujúceho projektu Multimediálna čítanka. Tento nástroj slúži deťom v predškolskom, respektíve skorom školskom veku ako pomôcka pri učení čítania. Tento projekt vznikol okolo roku 2005. Čítanka poskytuje viacero rôznych aktivít pre deti. Nás zaujímala hlavne aktivita čítania. Študent si otvorí webstránku a vyberie si príbeh, ktorý si chce prečítať. Následne sa mu na obrazovke zobrazí text a nasadí si na hlavu slúchadlá. Text je už dopredu predčítaný. Počas toho ako je študentovi príbeh predčítavaný, na obrazovke sa mu zvýrazňujú slová v texte, ktoré sú práve čítané. Jeho úlohou je v podstate sledovať tento text. Táto práca sa bude zaoberať vyriešením problematiky, ktorá vznikla u mnohých študentov. Tento problém bol v tom, že študenti občas nestíhali text čítať. Keďže študenti, ktorí nahrávky vytvárali už v súčasnej dobe rozhodne nie sú deti, nie je možné nahrávky spraviť nanovo a pomalšie. Cieľom tejto práce je vytvoriť nástroj, ktorý bude študentom poskytovať možnosť spomaliť rýchlosť s akou je im text predčítavaný. A to tak, aby sme zachovali identitu čitateľa. Teda aby sme sa vyhli rôznym mutáciam hlasu, ktoré pri spomaľovaní reči vznikajú. Najrozšírenejší z týchto mutačných efektov je zhrubnutie hlasu čitateľa až do nezrozumiteľnej miery.

V tejto kapitole postupne popíšeme základné pojmy, použité metódy a štatistické funkcie, ktoré sa používajú na zisťovanie atribútov digitálneho signálu.

1.2 Základné pojmy

1.2.1 Reč

S rečou sa stretávame každodenne, je všade okolo nás. Vzniká pomocou úst, nosa, jazyka a hlasiviek. Základnou stavebnou jednotkou reči sú hlásky, tie rozdeľujeme na samohlásky a spoluhlásky.

Všetky samohlásky sú vytvárané len hlasivkami a perami. Vieme ich vyslovovať aj samé o sebe, bez pomoci iných častí úst ako je napríklad jazyk alebo zuby. Vznikajú len pomocou pravidelného sťahovania a rozťahovania našich hlasiviek, ktoré vytvárajú hlasivkový tón. Tento tón je pre každý hlas špecifický. Frekvencia sťahovania a rozťahovania hlasiviek spôsobuje tón hlasu. Hlasivky, ktoré sa sťahujú a rozťahujú s vyššou frekvenciou, spôsobia tenší, vyšší hlas. Naopak „pomalšie" hlasivky spôsobujú hlbší hlas. Spoluhlásky sa vytvárajú o čosi zložitejšie a podľa charakteristických znakov ich vznniku spoluhlásky rozdeľujeme na explozíva, frikatíva a nosové spoluhlásky.

- Explozíva (napr. b, g, p, t, ť)
- Frikatíva (napr. ch, f, s, z)
- Nosové spoluhlásky (napr. m, n)...

Explozíva sú takzvané výbušné spoluhlásky. Vyznačujú sa tým, že pri ich vyslovovaní sa veľmi často pery spoja a nasleduje prudký presun vzduchu cez naše hlasivky. Tento prechod vzduchu je často sprevádzaný ďalšou hláskou v slove. Napríklad pri vyslovení slabiky „ba" je časť explozíva B veľmi malá. Ďalším znakom explozív je, že pri niektorých dokonca musíme zovrieť pery, prípadne pritlačiť jazyk o zuby. Zvuk hlásky je následne vytvorený len vytlačením vzduchu. Nosové spoluhlásky sa vyznačujú tým, že pri ich vytváraní sa vzduch prechádzajúci cez hlasivky prechádza cez nos. Hlasivky pri vytváraní nosových spoluhlások produkujú pravidelné hlasivkové tóny. Frikatíva vznikajú spravidla vypúšťaním vzduchu z pľúc cez jazykom vytvorenú štrbinu v ústach. Niektoré z nich pri tom hlasivky nepoužívajú vôbec, napríklad spoluhláska s. Naopak pri niektorých hlasivky pracujú a produkujú pravidelné hlasivkové tóny. Sú to napríklad spoluhlásky z alebo h. Ďalšie nemenej dôležité rozdelenie spoluhlások je na znelé a neznelé spoluhlásky. Z pohľadu signálu by sme hlásky mohli deliť na explozíva, hlásky bez hlasivkového tónu a hlásky s hlasivkovým tónom. [3]

- Explozíva (napr. b, g, p, t, ť)
- Hlásky bez hlasivkového tónu (napr. c, ch, f, s, z)
- Hlásky s hlasivkovým tónom (napr. a, h, l)...

Explozíva sú špecifický prípad hlások z pohľadu digitálneho signálu. Spôsob akým sa vytvárajú totiž spôsobí vytvorenie úseku signálu, ktorý je veľmi podobný osamotenému hlasivkovému tónu. Toto delenie bolo pre nás asi najdôležitejšie, nakoľko sme na jeho základe volili rôzne prístupy k syntéze jednotlivých hlások.

1.2.2 Digitálny signál

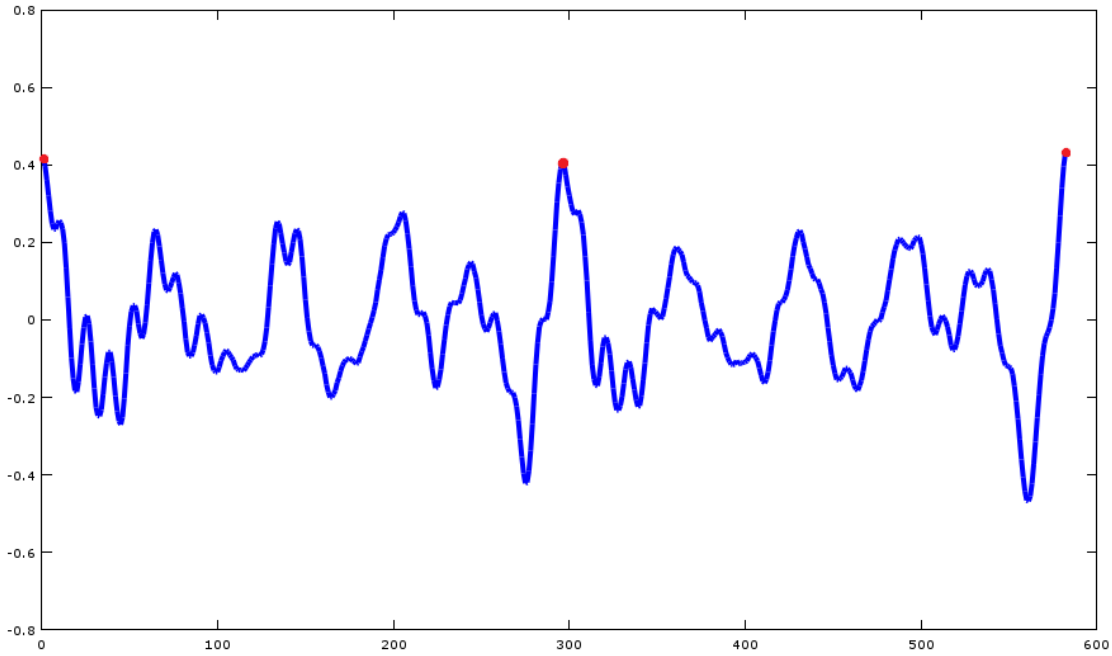
Digitálny signál je postupnosť diskrétnych hodnôt reprezentujúci nejaký jav. Existuje viacero druhov signálu. Napríklad striktne binárny, ktorého bity nadobúdajú hodnoty 1 alebo 0. Ďalšie typy vznikajú napríklad zložením rôznych sínusoviek. V mojej práci som sa zaoberal audiosignálom. Špecifický je tým, že hodnoty jeho bitov sú reálne čísla z intervalu $< -1, 1 >$. Slúži na reprezentáciu zvuku a jeho frekvencia sa pohybuje od 20 Hz do 20000 Hz. Využitie rôznych signálov má vo svete nesmierny význam. Od obyčajného rádia, či mobilného telefónu. Cez využitie v námorníctve ako sonar alebo letecké či dopravné radary. Až po zisťovanie histórie a zmien vo vesmíre. [5]

1.2.3 Signál reči

Signál reči zvyčajne vzniká zaznamenaním hlasu cez mikrofón alebo iné vstupné médium. Ľudský hlas je spravidla zložený z niekoľkých sínusoviek. Frekvencia tohto signálu priamo ovplyvňuje farbu hlasu, či melódiu viet.

Pojem hlasivkový tón v digitálnom signáli definuje špecifickú časť zaznamenaného zvuku, ktorá je lokálnym maximom v zhruba 25ms úseku nezašumeného signálu. Zdôrazňujeme, že signál musí byť nezašumený. Inak šum môže vytvoriť cinknutie, či ťuknutie, ktoré bude lokálnym maximom napriek výskytu hlasivkového tónu. V reálnom živote tento tón počujeme v reči veľmi pravidelne. V podstate ide o zvuk, ktorý vydávame s pomocou hlasiviek. Táto špecifická časť signálu je pre syntézu zvuku zo zaznamenanej reči veľmi dôležitá.

Ak chceme zachovať identitu hlasu a nespôsobiť jeho mutácie, nemá význam signál naťahovať alebo stláčať. Pri reči natiahnutie signálu spôsobí neprirodzene hlboký hlas, naopak stlačenie signálu by spôsobilo veľmi vysoký hlas. Je to z toho dôvodu, že v signále reči, je dôležitá vzdialenosť medzi hlasivkovými tónmi. Čím dlhšie sú vzdialenosti, tým je hlas hlbší a analogicky tenší. Frekvencia hlasivkových tónov v podstate definuje farbu ľudského hlasu. Preto ak chceme zachovávať identitu rozprávača, musíme zvoliť iný prístup.

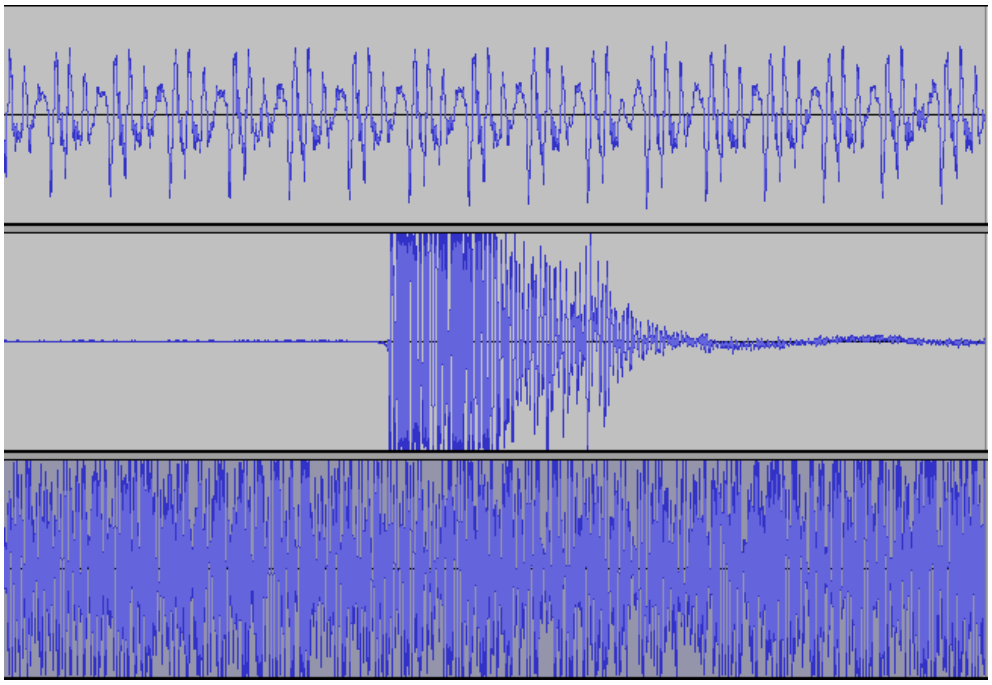


Obr. 1.1: Vektor zaznamenanej reči obsahuje dva hlasivkové tóny, ohraničené lokálnymi maximami naľavo, v strede a napravo

Na obrázku 1.1 môžeme vidieť 59ms rečového signálu. Tento úsek obsahuje práve tri hlasivkové tóny, prvý je na nultom bite, druhý zhruba na 290-tom a posledný na poslednom bite. Naše riešenie sa bude snažiť čo najdôveryhodnejšie napodobniť toto správanie signálu pri úsekoch ktoré obsahujú hlasivkové tóny.

Keď už sme definovali čo je to digitálny signál, môžeme si ukázať ako vyzerá signál jednotlivých typov hlások. Na obrázku 1.2 môžeme postupne vidieť signály hlások „a” , „k” a „s”.

Prvá spomenutá hláska je reprezentantom triedy hlások s hlasivkovým tónom. Na signále sa pravidelne opakujú hlasivkové tóny. Takýto signál tiež nazývamé pulzný signál, lebo pripomína svojou periodicitou pulz srdca. Druhá spomenutá hláska „k” je zástupcom explozív. Ako aj názov naznačuje, signál je intenzívny na veľmi krátkom časovom úseku. Predchádza a nasleduje za ním ticho. Za signálom tejto hlásky môžeme vidieť trošku odlišné správanie ticha ako pred ním. Toto správanie signálu spôsobujú hneď dva elementy. Prvým je človek a druhým vstupné zariadenie. Aj keď sme sa pri vytváraní vzorky veľmi snažili, nedokázali sme zabrániť dodatočnému výdychu po vyslovení hlásky. Výdych spôsobil slabé zašumenie. Šum dychu je ale relatívne pravidelný a jeho priemer by mal byť zhruba nulový. Na



Obr. 1.2: Porovnanie signálov rôznych hlások. Odhora na dol: samohláska, explozívum, frikatívum

našom obrázku 1.2 profil signálu ale pripomína pozvoľnú sínusovku. To je spôsobené zase vstupným zariadením, mikrofónom. Ten sa po explózii snaží automaticky vyrovnať s nasledujúcim tichom, ale chvíľu mu táto regenerácia trvá. Preto je vidieť sínusovku. Posledným písmenom na obrázku je frikatívum „s". Ako môžeme vidieť, signál je v podstate len intenzívny šum. Profil je úplne chaotický ale v porovnaní s tichom pred hláskou „k" je značne odlišný.

1.3 Pracovné prostredie, algoritmy a štatistické funkcie

V tejto časti sa budeme najskôr venovať zadefinovaniu nášho pracovného prostredia. Následne zadefinujeme štatistické funkcie a algoritmy, ktoré sme využívali. Pri algoritmoch a štatistických funkciách uvedieme v názvoch podkapitol zároveň aj príznaky využitia týchto nástrojov. Keďže niektoré popísané algoritmy sme používali pri analýze a iné zase pri syntéze signálu.

1.3.1 Octave

Octave je matematicky orientovaný programovací jazyk podobný Matlabu. Na rozdiel od Matlabu je však Octave voľne dostupný a aj to bol jeden z mnoha dôvodov prečo sme si vybrali toto pracovné prostredie. Trieda matematicky orientovaných programovacích jazykov

je príznačná tým, že vie optimálne narábať s matematickými štruktúrami. Keďže naše nahrávky zvuku sú v podstate veľmi dlhé vektory čísel, je toto prostredie pre nás vyhovujúce. Octave navyše poskytuje vynikajúce možnosti práce so signálom. Tieto možnosti podporuje najmä knižnica „signal". Od jednoduchého načítania zvuku do vektora až po jeho zápis do počuteľnej formy vo formáte „.wav" súborov. Ďalej Octave poskytuje riešenia mnohých štatistických funkcií. Ďalej Octave ako aj iné matematické jazyky poskytuje veľmi dobré možnosti vykresľovania grafov. Výsledné pozitíva výberu tohto pracovného prostredia teda sú: Knižnica signal, optimálna práca s veľkými matematickými štruktúrami a voľná dostupnosť tohto produktu.

1.3.2 Normalizácia signálu

Je celkom bežné, že mikrofón, nahrávacie médium signál mierne nadsadzuje alebo potláča. V nahranom signále sa to prejaví tak, že všetky jeho dáta budú o nejakú konštantu vyššie alebo nižšie ako bola skutočnosť. Táto konštanta môže spôsobiť niekoľko problémov, napríklad pri detekcii zmeny znamienok pri zero crossing rate funkcii popísanej v kapitole 1.3.4. Zbavenie sa tejto konštanty je pomerne jednoduché, najprv získame priemernú hodnotu na celom signále. Získavame ju klasickým aritmetickým priemerom hodnôt. Tento priemer potom pripočítame, ak je priemer záporný pripočítame zápornú hodnotu, ku každému bitu signálu a tým jeho hodnoty naškálujeme tak, aby mali základ v nule. Je nesmierne dôležité aby sme tento krok urobili ako prvý v poradí. Poskytneme tým rovnaké podmienky pre každý z nasledujúcich algoritmov a štatistických funkcií.

1.3.3 Autokorelácia

Autokoreláciu budeme využívať na hľadanie hlasivkových tónov vo vstupnom signále[4].

Korelácia vyjadruje vzťah medzi dvoma alebo viacerými náhodnými veličinami nejakého javu. V našom prípade sme na hľadanie hlasivkového tónu použili autokoreláciu, ktorá vyjadruje podobnosť v rámci úseku nejakého signálu. V tejto kapitole postupne odvodíme vzťah, ktorý autokorelácia popisuje v matematickom ponímaní. Následne uvedieme a popíšeme vzorec ktorým sa autokorelácia počíta v našej problematike.

Aby sme mohli dobre vysvetliť autokoreláciu, potrebujeme postupne odvodiť jej matematický vzorec. Jeho základom je kovariancia. Vstupom do kovariancie sú dve premenné, alebo veličiny. Výstupom je štatistický údaj ako sa tieto dve veličiny navzájom menia.

$$Cov(X, Y) = E[(X - E[X]) * (Y - E[Y])] \tag{1.1}$$

Vzorec 1.1 nám popisuje vzorec kovariancie. $Cov(X, Y)$ je zápis kovariančnej funkcie, kde X a Y sú náhodné premenné a E je stredná hodnota. Ďalej nám vzorec hovorí, že výsledkom kovariancie je stredná hodnota súčinu náhodných premenných. Aby sme zistili presnejší údaj o vzájomnej zmene náhodných premenných, odčítame od týchto ešte ich stredné hodnoty.

Strednú hodnotu náhodnej premennej vypočítame z nasledovných predpokladov a vzorca. Predpokladajme, že premenná X popisuje nejaký jav, či experiment. Môže teda nadobúdať hodnoty x_1, ... x_n . Pričom pre každú z hodnôt poznáme pravdepodobnosť s akou nastane. Potom je stredná hodnota X definovaná nasledujúcim vzorcom 1.2.

$$E[X] = \sum_{i=1}^{n} x_i * P[X = x_i] \tag{1.2}$$

Teda ako sumu jednotlivých hodnôt, ktoré náhodná premenná X nadobúda, vynásobenú o pravdepodobnosť s akou túto hodnotu nadobúda ($P[X = x_i]$). Keď už o výpočte kovariancie vieme dosť, môžeme sa konečne dopracovať ku vzorcu korelácie 1.3.

$$\rho(X, Y) = \frac{E[(X - E[X]) * (Y - E[Y])]}{\sigma_X * \sigma_Y} \tag{1.3}$$

Kde $\rho(X, Y)$ je zápis korelačnej funkcie a σ_X je skrátený zápis pre nasledujúci vzorec.

$$\sigma_X^2 = E(X^2) * E^2(X) \tag{1.4}$$

A teda σ je odmocnina z tohto vzorca 1.4. Rozpísaním sigmy do vzorca korelácie sprehľadníme vzorec a jednoduchšie s ním vieme počítať matematickú koreláciu. Sprehľadnený vzorec bude vyzerať nasledovne.

$$\rho(X, Y) = \frac{E[(X - E[X]) * (Y - E[Y])]}{\sqrt{(E[X^2] - E^2[X])} * \sqrt{(E[Y^2] - E^2[Y])}} \tag{1.5}$$

Dosadenie vzorcov pre výpočet strednej hodnoty E nechávame pre prehľadnosť na čitateľa. Podotkneme len niekoľko vlastností matematickej korelácie. Korelačný koeficient má obor hodnôt $H_\rho =< -1, 1 >$. Ak sú náhodné premenné nezávislé, tak korelačný koeficient (výsledok korelácie) týchto dvoch náhodných premenných je 0. Čo znamená, že sa nijako neovplyvňujú.

Keď sme odvodili a vysvetlili matematickú koreláciu, môžeme pokračovať na koreláciu, ktorá sa používa v digitálnom signále. Tu ju nazývame krížová korelácia, cross korelácia alebo aj skrátene xcorr. Ako takmer všetky matematické funkcie pre digitálny signál, aj táto má dve verzie. Jednu pre spojitý signál a druhú pre diskrétny digitálny signál. V našej problematike si koreláciu môžeme predstaviť ako podobnosť tvarov dvoch signálov. Ďalej definujeme aj τ, ktoré slúži na určenie časového posunu, v ktorom porovnávame signály.

Pri spojitých signáloch počítame koreláciu pomocou nasledujúceho vzorca 1.6.

$$(f \star g)(t) = R_{fg}(t) = \int f^*(t)g(t+\tau)d\tau \tag{1.6}$$

Pričom $\star$ je znak krížovej korelácie funkcií v čase t s posunom τ. Ďalej $f^*(\tau)$ znamená komplexný konjugát funkcie f pri posune τ. Komplexný konjugát znamená, že zoberieme komplexný tvar hodnoty $f(\tau)$ a hodnote pri imaginárnej zložke zmeníme znamienko. Príklad komplexného konjugátu môžete vidieť v nasledujúcom príklade.

$$\begin{aligned} f(y) &= x + 42i \\ f^*(y) &= x - 42i \end{aligned} \tag{1.7}$$

Problém s týmto vzorcom 1.7 je, že sa používa skôr v teoretickej časti spracovania digitálneho signálu. Nakoľko my sme pracovali so zaznamenanou rečou, tento signál bol diskrétny. Teda potrebovali sme využiť krížovú koreláciu pre diskrétny signál. Keďže sa bavíme o diskrétnom signále, vzorec bude obsahovať sumu jednotlivých diskrétnych hodnôt vstupných signálov. Celý výpočet korelačného koeficientu môžeme vidieť v nasledujúcom vzorci 1.8.

$$(f \star g)[n] = R_{fg}[n] = \sum_m f^*[m]g[m+n] \tag{1.8}$$

Pričom vstupný parameter do krížovej korelácie f a g je n, ktoré nám hovorí, v ktorom čase chceme korelačný koeficient získať. Zároveň si vopred stanovíme konštantu n, ktorá nám zase hovorí o časovom posune signálu g. Diskrétnu krížovú koreláciu získame ako sumu z komplexnéhých konjugátov hodnôt funkcie f a normálnych hodnôt funkcie g v čase m, pričom funkcia g je posunutá v čase o konštantu n.

Ďalej by sme radi poznamenali súmernosť krížovej korelácie. Platí nasledovná rovnica 1.9. Teda krížová korelácia je symetrická podľa bodu m.

$$
\begin{aligned}
(f \star g)[n] = R_{fg}[n] &= \sum_{m} f^*[m]g[m+n] \\
(f \star g)[n] = R_{fg}[n] &= \sum_{m} f^*[m-n]g[m]
\end{aligned}
\tag{1.9}
$$

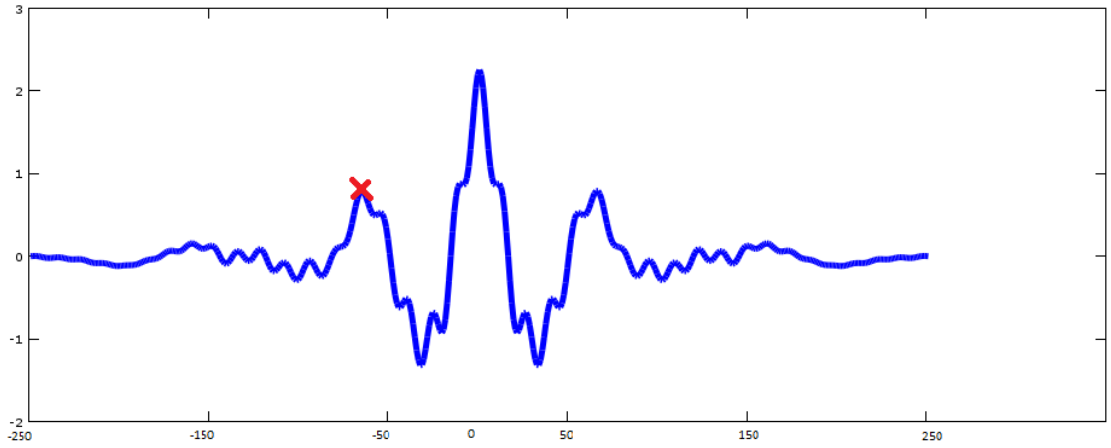
Keď sme už definovali všetko čo potrebujeme, už len stačí poznamenať, že autokorelácia v diskrétnom signále znamená, že vstupnými funkciami do krížovej korelácie je dva krát, tá istá funkcia. Teda autokorelácia signálu f bude vyzerať nasledovne 1.10.

$$
(f \star f)[n] = R_{ff}[n] = \sum_{m} f^*[m]f[m+n] \tag{1.10}
$$

Autokorelácia nám teda poskytuje silný nástroj na porovnávanie signálu so sebou samým. Toto je zdanlivo neužitočná informácia o našom signále ale nie je to tak. Keď zoberieme zo signálu dostatočne malé úseky, povedzme 25 ms (milisekúnd), môžeme predpokladať, že tento úsek obsahuje najviac dva hlasivkové tóny. Toto tvrdenie vychádza z faktu, že priemerná frekvencia hlasiviek dospelého človeka je zhruba 25ms. Naše nahrávky predčítavali študentom deti, ich hlas je vyšší ako hlas dospelého človeka. Ako sme už uviedli v kapitole 1.2.1 Reč, čím je hlas tenší, tým je vyššia frekvencia sťahovania a rozťahovania hlasiviek. Teda predpokladáme, že v 25ms dlhom kúsku signálu budú najviac 2 hlasivkové tóny. Teraz na takýto signál spustíme autokoreláciu po celej jeho dĺžke. Teda zoberieme náš 25ms úsek a postupne ho budeme porovnávať so sebou. Pričom náš lag bude patriť intervalu $< -25ms, 25ms >$.

Výsledkom bude vektor hodnôt, ktoré nám povedia ako moc sa signál v konkrétnom čase lagu na seba podobal. Takýto vektor môžeme vidieť na obrázku 1.3.

Ako môžeme na grafe 1.3 vidieť, je symetrický a v strede má najvyšší bod. Tento bod nastal v čase, keď bol lag 0 a teda signál sa porovnával sám so sebou v identickom čase. Sústreďme sa teraz len na polovicu z intervalu $< 0, length/2 >$. Na tomto intervale môžeme vidieť dva konvexné body, ktoré sú výrazne vyššie ako ostatné konvexné body krivky. Práve toto sú body, ktoré by mohli reprezentovať hlasivkové tóny v signále zaznamenanej reči. Ako z autokorelácie zistíme kde naozaj hlasivkové tóny sú, si ukážeme v kapitole 3, naše riešenia. [7]



Obr. 1.3: Výstupný vektor z autokorelácie s označeným lokálnym maximom, ktoré môže znamenať výskyt hlasivkového tónu vo vstupnom signále

1.3.4 Zero-crossing rate

Názov tejto funkcie v preklade frekvencia prechodov cez nulu. Táto štatistická funkcia robí presne to, čo naznačuje jej názov. Túto funkciu sme používali na detekciu hlások bez hlasivkového tónu, frikatív. [4]

$$zcr = \frac{\sum_{t=1}^{T-1} \mathbb{I}(s_t * s_{t-1} < 0)}{T - 1} \tag{1.11}$$

Vo vzorci 1.11 môžeme vidieť jednoduchú sumu, ktorej definičný obor je interval $< 1, T - 1 >$. V sume môžeme vidieť takzvanú indikátorovú funkciu alebo indikátor splnenia podmienky, ktorú obsahuje. V našom prípade zoberieme dve po sebe idúce hodnoty signálu a vynásobíme ich. $\mathbb{I}$ vráti 1 ak vynásobená hodnota je menšia ako 0. Teda jeden z činiteľov súčinu je záporný a druhý kladný. V opačnom prípade vráti $\mathbb{I}$ 0. Suma nám teda vráti koľko krát signál za čas T prešiel nulou. Nakoniec potrebujeme sumu ešte znormalizovať, aby sme získali hodnotu koľko krát prejde signál 0 na jednotku času. Výsledok potrebujeme znormalizovať dĺžkou meraného vektora. Normalizáciu robíme preto, že zero crossing rate nepoužívame vždy na rovnako dlhých vektoroch. Tiež sme si potrebovali stanoviť určitú hranicu, ktorá naznačuje, že nejde o ticho respektíve o zvuky, ktorým by sme sa pri predlžovaní signálu radi vyhli. Tento údaj ale ešte nebude stačiť. Môže nastať situácia kedy, či už mikrofón alebo šum spôsobí podobnosť signálu s frikatívom.

1.3.5 Rozptyl, sigma

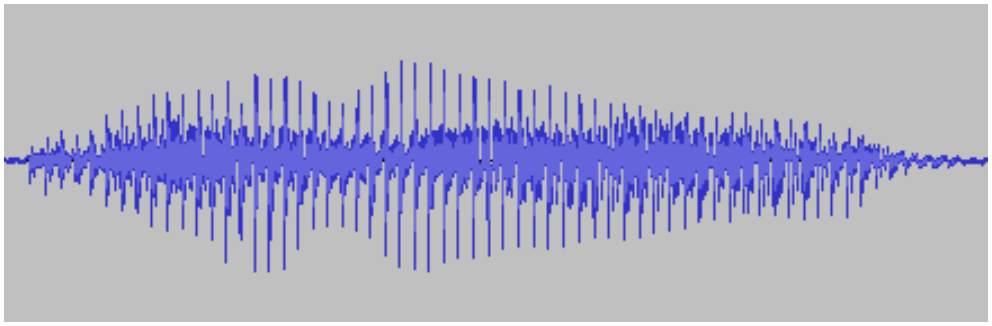
Aby sme získali informácie zo zero crossing rate, ktoré boli pre nás dôležité, potrebovali sme počítať aj rozptyl. Rozptyl alebo sigma náhodnej premennej nám určuje interval, v ktorom by mala skončiť väčšina hodnôt náhodnej premennej.

$$\sigma^2(X) = E[(X - E[X])^2] \tag{1.12}$$

Vo vzorci môžeme vidieť σ^2 je rovná strednej hodnote štvorcov rozdielov hodnoty náhodnej premennej X a strednej hodnoty tejto náhodnej premennej. Teda zjednodušene, keď poznáme strednú hodnotu náhodnej premennej X. Tak výsledok z tohto vzorca nám určuje nasledovné. Ak sme namerali nejakú hodnotu náhodnej premennej X, tak väčšina hodnôt, ktoré premenná nadobudne bude ležať v intervale $X \pm \sigma^2(X)$.

1.3.6 Energia

Amplitúda signálu sa mení. Môžeme to vidieť napríklad na obrázku 1.4. Na obrázku môžeme vidieť signál slova „Ahoj". Túto amplitúdu vieme dobre sledovať pomocou krátkodo-



Obr. 1.4: Signál slova „Ahoj" , jeho amplitúdu budeme sledovať pomocou energie

bej energie signálu. Najjednoduchším výpočnom energie signálu, je suma druhých mocnín jednotlivých bitov vektora digitálneho signálu v čase 1.13. Samozrejme nemá zmysel počítať energiu na celej dĺžke našej nahrávky. Táto informácia by nám nič nepovedala. Sledovaný signál, potrebujeme rozdeliť na kratšie vektory.

$$E_X = \sum_{m=-\infty}^{\infty} X(m)^2 \tag{1.13}$$

Vo vzorci 1.13 teda vidíme sumu druhých mocnín jednotlivých bitov. Túto štatistickú funkciu budeme využívať na odlíšenie ticha a frikatív. Teda tiež pri detekcii hlások bez hlasivkového tónu. Keď už budeme mať signál rozdelený podľa hlasivkových tónov, získame

neurčité úseky respektíve bloky signálu. Tieto môže byť frikatívom, explozívom, tichom alebo šumom. Frikatíva by sme predlžovať chceli, naopak explozíva rozhodne nechceme duplikovať. Na tieto neurčité úseky postupne spustíme Zero-crossing rate, sigmu a energiu a tým získame informáciu o tom, či pôjde o frikatívum alebo o niečo čo by sme predlžovať nechceli.

1.3.7 PSOLA

Skratka PSOLA znamená pitch synchronous overlap and add. Ide o spôsob syntézy signálu. Tento spôsob sa používa hlavne pri syntéze reči, keďže sa pri syntéze synchronizuje vzdialenosť tónov, v našom prípade hlasivkových tónov. Ako sme už spomínali, vzdialenosť hlasivkových tónov je veľmi dôležitá pri zachovaní identity rozprávača. Čím su hlasivkové tóny od seba viac vzdialené, tým je hlas rozprávača dlhší. Tento algoritmus je tiež dôvodom prečo sme volili tak malé úseky signálu. Keď môžeme predpokladať kde sú v signáli hlasivkové tóny, sme v podstate pripravení na syntézu reči pomocou tohto algoritmu. [8] [9]

PSOLA zoberie vždy práve jeden blok a z neho vytvára ďalšie hlasivkové tóny. Najprv signál bloku rozdelí na ľavú a pravú stranu podľa stredného hlasivkového tónu. O týchto vieme, že sa dajú bez akýchkoľvek úprav správne pripojiť do výsledného toku signálu. Avšak treba dávať pozor, že jednotlivé hlasivkové tóny sa opakujú až v troch blokoch za sebou. Teda treba si vybrať, ktorú časť signálu budeme nadpájať na celkovú reč. V podstate je jedno či budeme nadpájať ľavú alebo pravú, zmení to len podmienku pri prvom alebo poslednom bloku. My sme si vybrali nadpájanie ľavou stranou. To znamená, že po tom čo si vyberieme blok a rozdelíme ho na ľavú a pravú stranu, výstupom bude kus signálu kde jeho začiatok bude totožný s pôvodným vstupným blokom. Nasledovať bude nasyntetizovaný signál a pravá strana signálu sa nepridá na koniec výstupného bloku. Keby sme ju totiž pridali, vznikne nám na pravej strane výstupu dva krát po sebe idúca ľavá strana nasledujúceho vstupného bloku, čo sme nechceli dosiahnuť.

Syntéza hlasivkového tónu prebieha nasledovne. Porovná sa veľkosť ľavej a pravej strany, tento krok sa robí preto, že sa snažíme zachovať vzdialenosť nášho prostredného hlasivkového tónu a nášho predchádzajúceho, hlasivkového tónu v bloku. Ak je ľavá strana inej dĺžky ako pravá, znamená to buď klasické kolísanie ľudského hlasu alebo, že sa začína meniť melódia vety. Ak je ľavá strana kratšia, potrebujeme pravú stranu skrátiť. Skrátime ju od konca o rozdiel ľavej a pravej strany. Ďalej ak je ľavá strana dlhšia, aby sme mohli zachovať vzdialenosť hlasivkových tónov vo výstupnom signále, musíme pravú stranu predĺžiť o rozdiel ľavej a pravej strany. Pre jednoduchosť sme pravú stranu predlžovali nulami. Inak ak je ľavá strana kratšia, musíme pravú stranu bloku skrátiť o rozdiel ľavej a pravej strany.

Teraz keď máme pripravené všetky potrebné časti môžme začať skladať signál. Štandartne sa reč pri PSOLA algoritme syntetizuje tak, že zoberieme pravú stranu a postupne lineárne znižujeme jej vplyv na výsledný signál, počínajúc v hodnote 1 a končiac v 0. Analogicky zosilňujeme vplyv ľavej strany. Štandartne sa tento krok zosilovania a zoslabovania robí podľa vzorca 1.14.

$$hss[i] = hsLS[i] * \left(\frac{i}{n}\right) + hsPS[i] * \left(1 - \frac{i}{n}\right) \tag{1.14}$$

Vo vzorci môžeme vidieť nasledovné. Hodnota stredného signálu je rovná súčtu ľavej a pravej strany pôvodného signálu. Pričom v čase sa vplyv ľavej strany zvyšuje a vplyv pravej strany sa znižuje. Stredná strana signálu vytvorí syntetizovaný hlasivkový tón z pôvodného tónu. Výsledný syntetizovaný signál vyskladáme tak, že zoberieme pôvodnú ľavú stranu, za ňu nadpojíme vektor stredného signálu a následne nadpojíme pravú stranu pôvodného signálu.

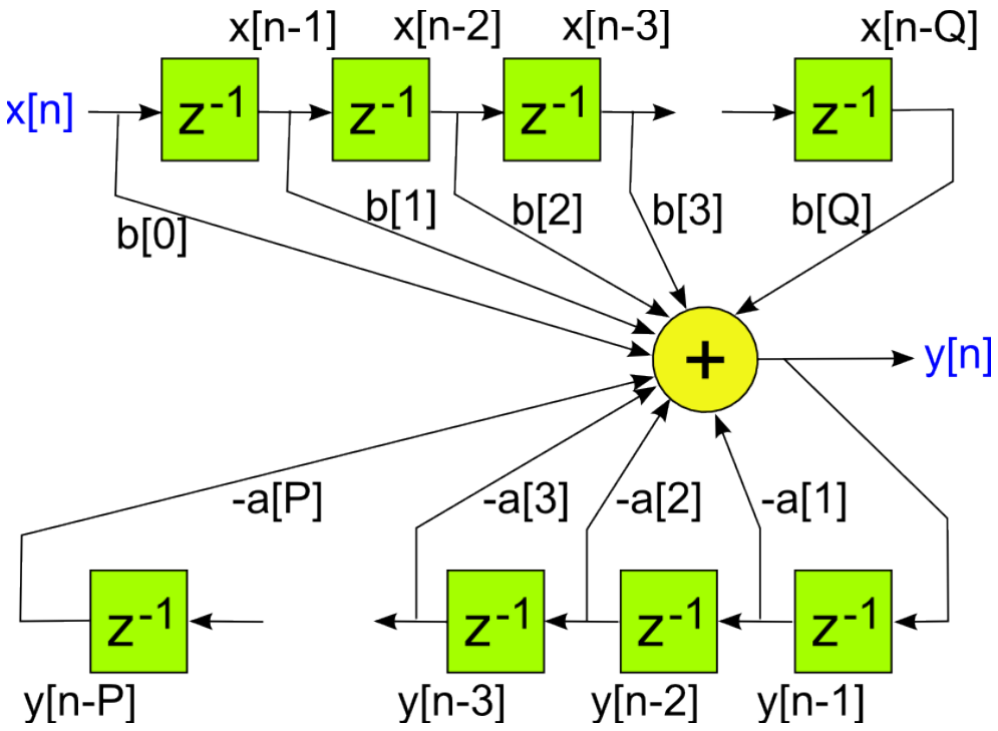
1.3.8 IIR filtrovanie

V predchádzajúcich kapitolách sme ukázali aké funkcie budeme používať pri analýze a následne pri syntéze signálu. Pri analýze sme ale neodstraňovali šum. Jeho odstránenie nie je jednoduché. Na jeho odstránenie by sme museli mať nahrávku tohto šumu respektíve ticha, ktorou sme pri našom riešení nedisponovali. Šum sme sa pre zjednodušenie riešenia teda rozhodli nevynechať. Pri syntéze ale šum značne vystúpil do popredia a rôzne puknutia a otáčanie strán v triede znížilo kvalitu výstupného signálu reči. Do nášho riešenia sme teda pridali ešte aj postspracovanie výstupu aby sme kvalitu znova zlepšili. [2]

Na zlepšenie výstupu sme sa rozhodli použiť filter. V spracovaní signálu sa stretávame s dvoma typmi filtrov. Konečnými a nekonečnými. Ich konečnosť sa určuje pomocou takzvaného impulse response.

Impulz je špeciálny druh signálu. Je to vektor, ktorý začína nulami, nasleduje práve jedna jednotka a pokračuje nulami.

Nekonečné filtre alebo aj IIR (infinite impulse response) sú rekurzívne filtre, ktoré samé seba upravujú na základe výstupu. Výstup z filtra sa znova posiela do filtra a spracováva sa ďalej. Vizualizáciu tohto procesu vidíme na obrázku 1.5.



Obr. 1.5: Výstup z IIR filtra. Horný signál vstupuje do IIR filtra; po získaní výstupu, tento vstupuje do filtra znova [2]

Ako môžeme vidieť, keď do takéhoto filtra pustíme impulz, výstup bude nekonečný. Konečné filtre, alebo aj FIR filtre, majú konečný výstup. Ich funkčnosť si môžeme predstaviť ako potláčanie respektíve zvyšovanie niektorých zložiek signálu. Na vstupe máme opäť signál. Tento signál pustíme na vstup filtra. Výstupom je konečný signál, ktorého niektoré hodnoty sú modifikované pomocou tohto filtra. My sme sa rozhodli pre použitie IIR filtrov, keďže tieto sa v praxi používaju bežne.

Existuje viacero typov IIR filtrov. Každý z nich sa vyznačuje inými vlastnosťami. Low-pass filtre, ktoré sme použili my, sa vyznačujú tým, že na vstup dostanú okrem iného aj vrchné ohraničenie frekvencií, ktoré nebudú modifikované. Zvuky, ktoré prekročia túto hraničnú frekvenciu budú potláčané. High-pass filtre fungujú úplne analogicky k low-pass. Na vstup teda dostanú spodné ohraničenie frekvencií, ktoré budú potláčať. Vyššie frekvencie naopak prejdú cez sito bez modifikácie. Zložením low-pass a high-pass filtra dostaneme band-pass filter, ktorý tlmí signál mimo povoleného intervalu frekvencií. Štvrtým typom filtra je band-stop filter, ktorý naopak signál s frekvenciami zo vstupného intervalu potláča.

Kvalitu nášho výstupu zhoršovali rôzne typy puknutí a tieto sme potrebovali odstrániť. Preto sme sa rozhodli pracovať s low-pass filtrom, ktorý potláča frekvencie vyššie ako je hranica, ktorú dostane na vstupe. Existuje viacero rôznych filtrov. Odlišujú sa najmä tem-

pom potláčania nežiadúcich frekvencií. Na obrázku 1.6 môžeme vidieť štyri rôzne spôsoby potláčania nežiadúcich frekvencií v signále. V ľavom hornom grafe môžeme vidieť najjednoduchší z filtrov. Potláčanie frekvencií je v porovnaní s ostatnými relatívne pozvoľné. Autorom tohto filtra je Stephen Butterworth. Jednoduchosť tohto filtra spôsobila jeho veľké rozšírenie vo svete. Okrem jednoduchosti na naprogramovanie sa v porovnaní s ostatnými filtrami jednoducho fyzicky zostrojiť. Jeho pozvoľné potláčanie ale znamená problém, nakoľko z kvalitatívneho hľadiska frekvencie nepotláča až tak efektívne ako ostatné zobrazené filtre.

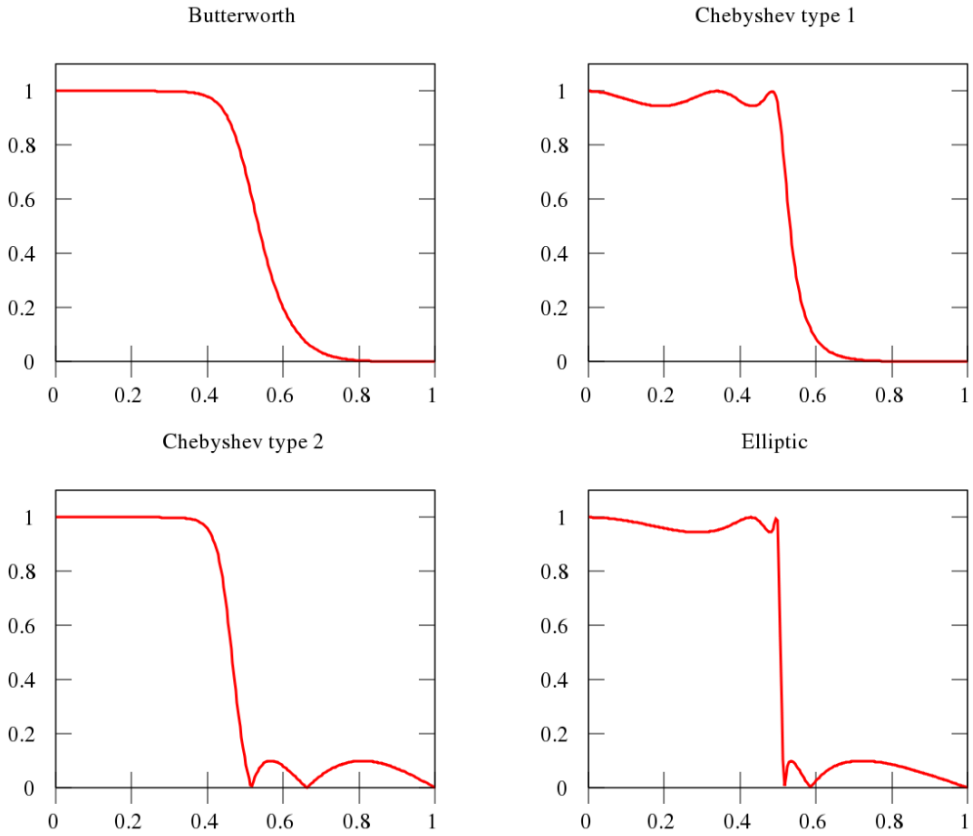
Na obrázku 1.6 na grafe vpravo hore vidíme filter odvodený z Chebyshevovho polynómu. Ako môžeme vidieť, rýchlosť potlačenia nežiadúcej frekvencie je oveľa rýchlejšia. Cena, ktorú za zrýchlenie zaplatíme je mierne zvlnenie pred samotným potlačením. Toto mierne zakolísanie je pre ľudské ucho v reči takmer nepočuteľné. Kolísanie je viac počuť pri signáloch hudobných nástrojov. Na konci potláčania, teda okolo bodu 0,6 na osi x, môžeme pozorovať pozvoľné monotónne klesanie hodnôt. Teda signál po prejdení týmto filtrom v okolí miest výskytu potláčaných frekvencií mierne zakolíše a následne nežiadúcu frekvenciu potlačí.

Na obrázku 1.6 na grafe vľavo dolu vidíme iný filter odvodený tiež z Chebyshevovho polynómu. Čo sa týka poklesu správa sa v podstate opačne k predchádzajúcemu filtru. Pokles začína bez vlnenia a je relatívne rýchly. Na druhej strane sa po potlačení frekvencie sa vytvorí ešte ozvena, ktorú môžeme vidieť napríklad na intervale $< 0.5, 0.65 >$. Túto ozvenu by sme pri filtrovaní reči v signále nechceli mať. Keď by sme na vstupnom signále mali chybu, ktorú by sme sluchom počuli ako nejaké ťuknutie, tento filter by síce túto frekvenciu potlačil relatívne rýchlo, ale ozveny by mohli spôsobiť počuteľnosť tejto chyby v signále aj naďalej. Na obrázku 1.6 vpravo dole môžeme vidieť posledný filter. Potláčanie frekvencií týmto filtrom je relatívne symetrické. Elipsový filter sa preto viac používa pri band-pass a band-stop filtrovaní. Zo všetkých filtrov v ukážke má najrýchlejšie potlačenie nežiadúcich frekvencií. Cena, ktorá sa za to platí je zakolísanie pred a ozvena po potlačení frekvencie.

Jemné zvlnenie pred potlačenou frekvenciou nám teda neprekáža ale ozvenu by sme pri filtrovaní nechceli mať. Keď vylúčime ozvenové filtre, ostanú nam na výber Butterworth a Chebyshev 1. Keďže potrebujeme tiež frekvencie čo najrýchlejšie potlačiť, vybrali sme si filter typu Chebyshev 1.

$$G_n(\omega) = |H_n(j\omega)| = \frac{1}{\sqrt{1 + \varepsilon^2 T_n^2\left(\frac{\omega}{\omega_0}\right)}} \tag{1.15}$$

Vo vzorci 1.15 môžeme vidieť funkciu filtra, ktorú sme použili na vylepšenie výsledkov. Konštanta ω_0 určuje hranicu frekvencie, nad ktorú sa nechceme dostať. V našom riešení sme potláčali frekvencie nad 1000Hz. T_n je Chebyshevov polynóm n-tého stupňa. My sme použili polynóm desiateho stupňa. Podľa stupňa polynómu vieme čiastočne vylepšiť potláčanie nežiadúcich frekvencií, ale výpočet je náročnejší. Premenná ε nám určuje o koľko decibelov chceme danú frekvenciu potlačiť. Pri voľbe tejto premennej sme sa držali štandardu a potláčali sme o 6dB.



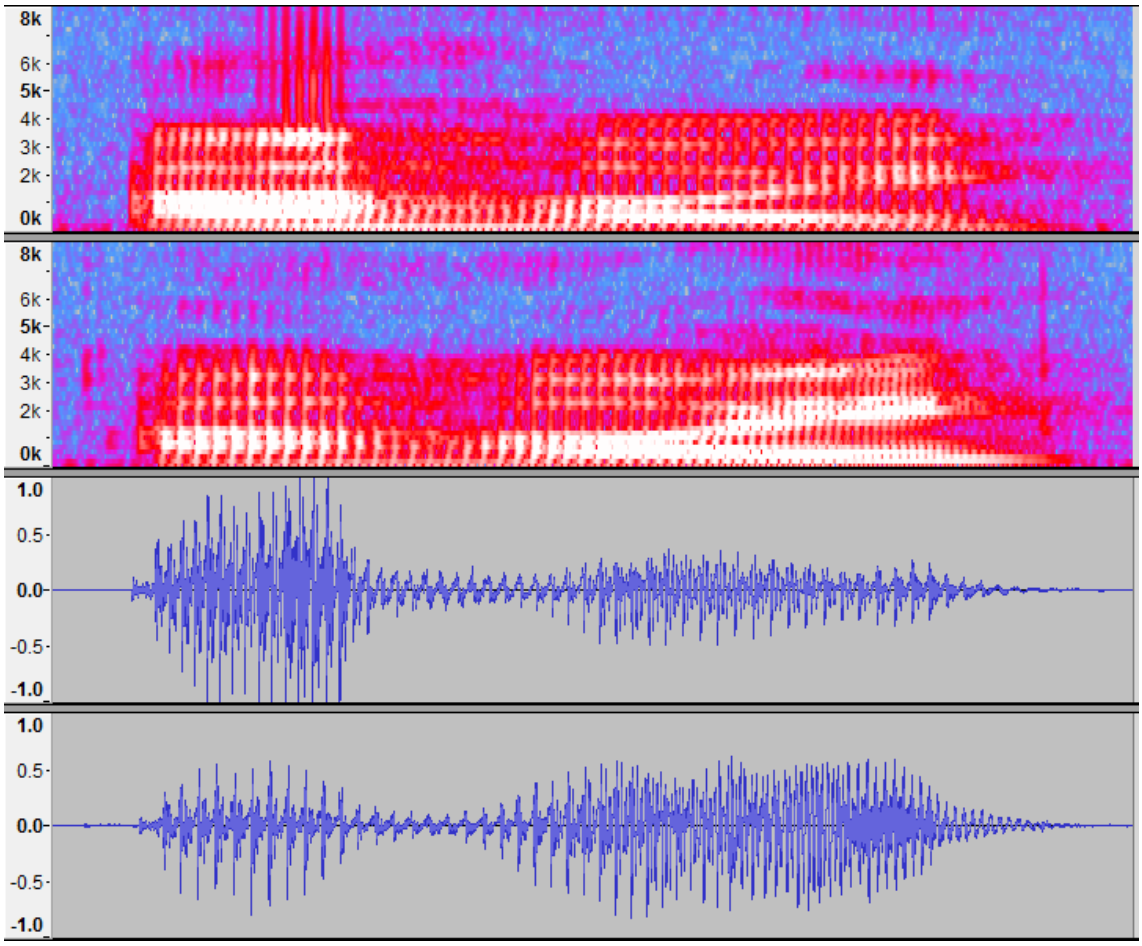
Obr. 1.6: Krivky spôsobov potláčanie nežiadúcich frekvencií pri rôznych typoch filtrov [2]

Kapitola 2

Riešenie problematiky

V tejto kapitole sa budeme venovať podstaným vylepšeniam a praktickému využitiu funkcií spomenutých v predchádzajúcej kapitole. Rovnako popíšeme aj ich správne použitie, aby sme sa dopracovali k výsledným nahrávkam, ktoré budeme môcť poskytnúť našim študentom.

Na 2.1 môžeme vidieť spektrogramy a signály slova „ahoj". Prvý a tretí obrázok reprezentujú slovo povedané v oznamovacom spôsobe. Druhý a štvrtý v opytovacom spôsobe. Spektrogramy sú horné dva obrázky. Spektrogram funguje tak, že na krátkych úsekoch signálu meria jeho frekvencie. Teplejšie farby teda, červená a žltá, reprezentujú zvýšenú frekvenciu signálu, naopak chladné, fialová a modrá, reprezentujú zníženú frekvenciu zaznamenaného signálu. Ako môžeme vidieť pri opytovacom spôsobe, teda druhom spektrograme, sa frekvencia zvyšuje smerom ku koncu slova. Naopak, pri oznamovacom slove je frekvencia sústredená viac na začiatku slova a pozvoľne sa znižuje. Podobne je prízvuk na slove vidieť aj v samotných signáloch slova. V druhom obrázku signálu je amplitúda signálu na konci slova evidentne širšia ako pri oznamovacom spôsobe, ktorý môžeme vidieť na treťom obrázku. Melódia je v slove a vete teda veľmi dôležitá a je často aj jedným z väčších problémov pri syntéze reči z textu. Nám sa tento problém ale pri syntéze opäť stratí. Keďže budeme syntetizovať vždy okolo 25 ms signálu, na spodných grafoch tento časový úsek môžeme vidieť ako vzdialenosť medzi pravidelnými hlasivkovými tónmi. Teda amplitúda signálu a teda aj melódia sa zachová, akurát sa mierne natiahne. Ale to je presne to čo sme chceli v konečnom dôsledku dosiahnuť.



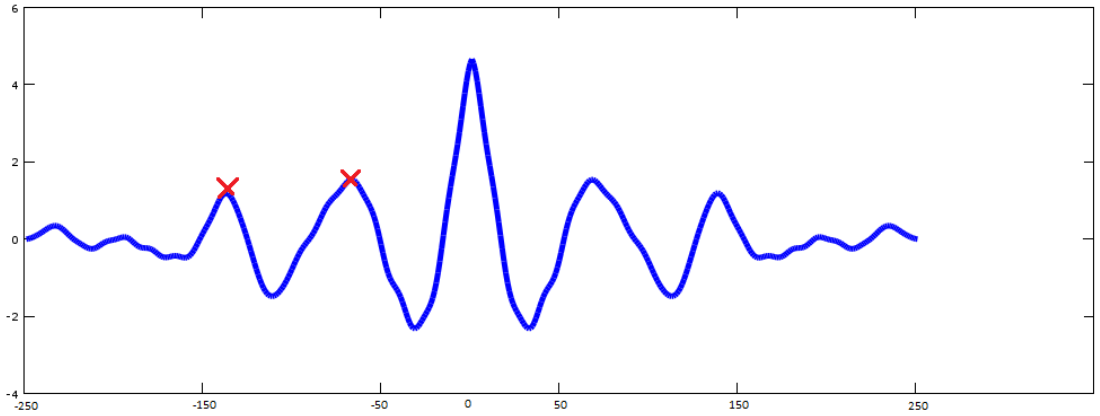
Obr. 2.1: Rozdiely spektra slova „ahoj" povedaného oznamovacím (prvý a tretí graf) a opytovacím (druhý a štvrtý graf) spôsobom

2.1 Autokorelácia

Ako sme už ukázali v podkapitole 1.3.3, autokoreláciu môžeme využívať na porovnávanie signálu so sebou samým. Výsledok autokorelácie môžeme vidieť na obrázku 2.2.

Ako môžeme vidieť, autokorelácia signálu je symetrická podľa stredu. Je to spôsobené tým, že porovnávame signál sám so sebou, pričom prvý porovnávaný je posunutý o takzvaný lag, ktorý je definovaný na intervale $< -ds, ds >$ prčom „ds" je dĺžka porovnávaného signálu. Lag postupne prechádza po celých číslach z tohto intervalu. Teda v momente keď je lag rovný nule, porovnávame signál sám so sebou bez posunu. Teda autokorelácia musí byť v tomto bode globálne najvyššia. Na detekciu hlasivkových tónov nám teda stačí sledovať prvú polovicu výstupného vektora z autokorelácie.

Aby sme zaistili kde nájdeme najvyšší bod mimo globálneho maxima, potrebujeme ešte o niečo skrátiť náš súčasný interval. Tento krok sme robili tak, že sme sa „skotúľali z maxima",

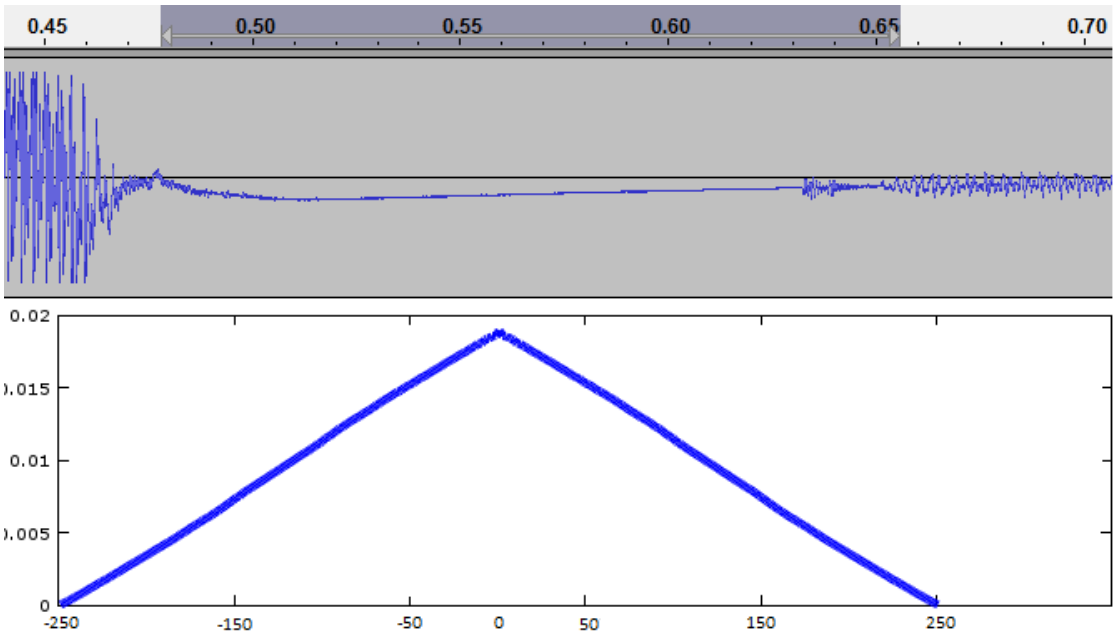


Obr. 2.2: Ukážka výstupu autokorelácie označené body znázorňujú miesta, kde by sa mohol nachádzať hlasivkový pulz

teda sme skracovali vektor autokorelácie od konca dovtedy, kým sme sa nedostali do bodu, kedy začali hodnoty vektoru opäť rásť. Na obrázku Obr. 2.2 sa tento konvexný bod nachádza v bode 0 na osi x.

Počas našej práce sme narazili na neočakávané správanie autokorelácie pri špecifických podmienkach vstupného signálu. V jednej z nahrávok vzinkol kvoli šumu 15 ms dlhý úsek signálu, ktorý sa celý nachádzal v záporných hodnotách. Takáto situácia nastala, keď v nahrávke reči bola chvíľku pauza, kde si čítajúci študent otáčal stranu. Zároveň túto akciu prevádzali všetci študenti v triede, kde sa nahrávka nahrávala. Ticho v spojení so šumom, ktorý pretáčanie strán spôsobilo tento neočakávaný jav. Krivka, po ktorej šum osciloval bola konvexná.

Vzniknutý problém je znázornený na obrázku 2.3. Signál na hornom obrázku je už normalizovaný, teda sme vylúčili globálne nadsadzovanie respektíve podsadzovanie nahrávaného signálu. Tento fenomén teda vznikol lokálne, kvoli šumu. Keďže cieľom tejto práce nebolo odstránenie šumu z nahrávky, museli sme sa s týmto fenoménom vysporiadať inak. Hlavným problémom bolo naše „kotúľanie z maxima". Autokorelácia na takomto kuse signálu bola totiž striktne monotónna a rastúca. Tak ako môžeme vidieť na dolnom grafe. Teda počas hľadania konvexného bodu vo výsledku autokorelácie, sme tento nenašli. Nakoľko aj syntéza takýchto úsekov signálu by bola príliš náročná, rozhodli sme sa ich preskočiť a neduplikovať. Tento problém by sa dal riešiť pomocou odstránenia šumu. Predpokladáme, že signál na Obr. 2.3 by sa po odstránení šumu pohyboval na vyznačenom úseku okolo nuly a bol by reprezentovaný ako ticho.



Obr. 2.3: Šum na hornom grafe viedol k znemožneniu nájdenia vedľajších maxím autokorelácie

Teraz sa ale vráťme k autokorelácii. Ďalej budeme predpokladať, že sme nejaký konvexný bod vo výsledku autokorelácie a výstupný vektor sme skrátili, aby patril intervalu < 0, pozícia_konvexného_bodu>

Takto skrátený vektor autokorelácie už môžeme považovať sa vhodný pre hľadanie výskytu hlasivkových tónov. Pokračovali sme tak, že sme našli maximum na tomto vektore. Jeho pozícia na časovej osi, nám určila veľkosť „okienka" a aj vzdialenosť, o ktorú ho budeme posúvať. Okienko je vzdialenosť medzi začiatkom vektora, teda bodom 0 a nájdeným maximom. O tomto úseku signálu vieme s určitosťou povedať, že obsahuje aspoň jeden hlasivkový tón. Ale ešte stále ich môže obsahovať aj viac a to treba preveriť. Robíme to tak, že naše okienko poposúvame po celom vektore a skontrolujeme, či sme nenašli ďalšie maximá. Okienko posúvame po intervale o jednu tretinu jeho dĺžky a vždy hľadáme na jeho intervale globálne maximum. Toto ešte stále ale nemusí viesť k získaniu všetkých hlasivkových tónov. Môže nastať situácia, kedy by nájdené globálne maximum bolo na pozícii 650 na obrázku Obr. 2.4 a nie na 275. Vtedy je veľkosť okienka dostatočná a keď ho ľubovoľne budeme dopredu posúvať, maximum bude stále práve bod 650. Preto je dôležité posunúť okienko aj dozadu.

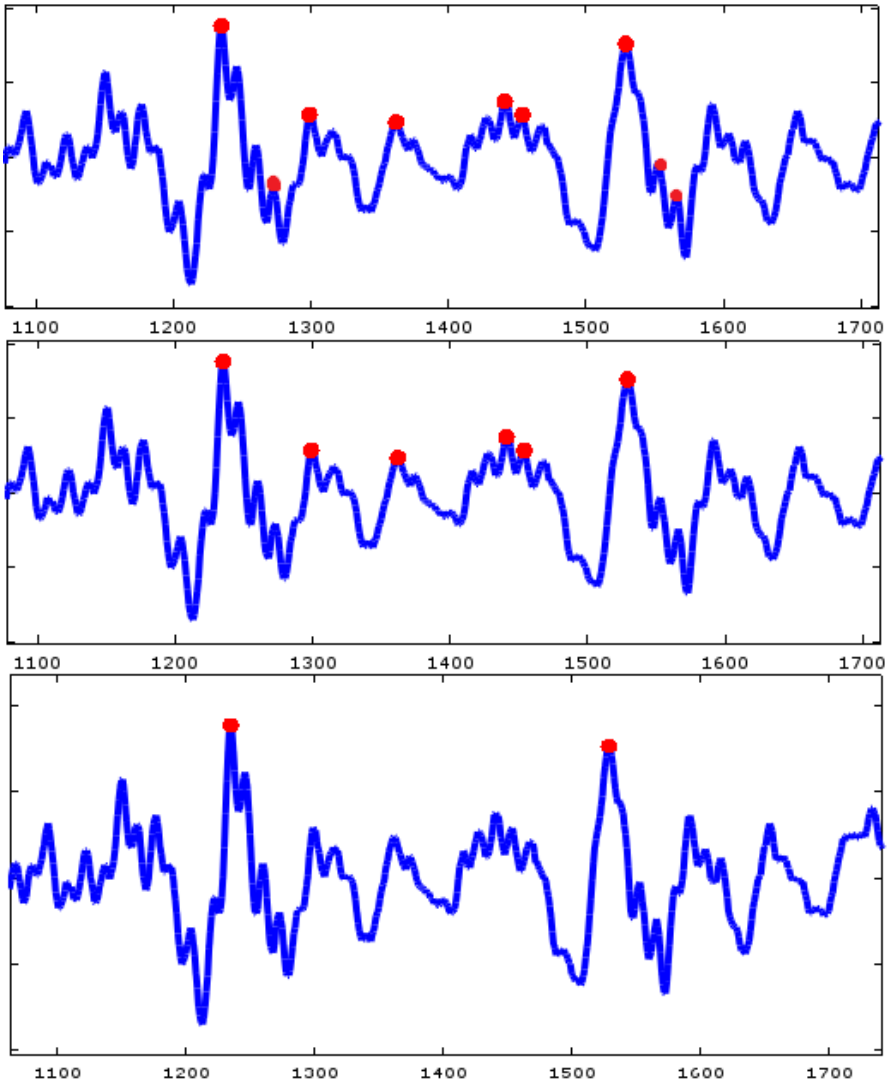
Takto nájdených maxím bude ale veľa, najviac toľko, koľko krát sme okienko po korelácii posúvali. Teraz potrebujeme vybrať len tie, ktoré sú pre nás naozaj dôležité a znamenajú výskyt skutočného hlasivkového tónu. Preto sme vybrali všetky body, ktoré sme zistili po-

mocou posúvania okienka na autokorelácii a zistili sme ich skutočné hodnoty vo vektore signálu nahranej reči. V premennej si udržujeme informáciu o pohyblivom priemere hodnôt už nájdených hlasivkových tónov. Tento priemer je udržovaný len za krátky časový úsek 50ms, aby sme vedeli zabezpečiť, že získavame správne informácie o tom, či nájdené maximum z autokorelácie je naozaj hlasivkovým tónom. Potenciálne hlasivkové tóny porovnáme s načim priemerom, ak sú vyššie ako priemer, tak ich prehlásime za hlasivkové tóny. Ak sú menšie len o menej ako 5% priemeru, tiež ich prehlásime za skutočné hlasivkové tóny. Túto toleranciu sme získali experimentálne. Hľadali sme ju z toho dôvodu, že v reči nehovoríme len jednu hlásku. Pri prechode medzi hláskami, sa hodnota hlasivkového tónu v signále reči môže znižovať. Pri tak malých úsekoch ako sme zvolili t.j. 25ms sa neznižuje o viac ako 5%.

Ešte stále ale môže nastať prípad, kedy sa jeden hlasivkový tón označí viac krát. Nakoľko okienko pre hľadanie tónov neposúvame o celú jeho dĺžku, ale len o jej zlomok. Vždy teda nastane prípad, kedy sa dve po sebe idúce okienka prekrývajú. Prvým riešením bolo kritérium, zakazujúce označiť dva krát ten istý bod signálu. Problém ale je, keď okienko začína napríklad 1ms za už nájdeným hlasivkovým tónom. Vtedy sa môže mylne označiť aj niekoľko po sebe idúcich bodov v signále, kde tón doznieva. Tieto body nemajú rovnaké súradnice a teda dostanú sa do nášho zoznamu hlasivkových tónov. To všetko za predpokladu, že používame spomenuté kritérium, ktoré nepovoľuje viacnásobné pridania rovnakého bodu do zoznamu hlasivkových tónov.

Na obrázku 2.4 môžeme vidieť ilustráciu problému označenia redundantného hlasivkového tónu. Červené bodky označujú hlasivkové tóny, ktoré autokorelácia považovala za hlasivkový tón. Na hornom grafe môžeme vidieť pôvodný výstup autokorelácie bez pohyblivého priemeru hodnôt nájdeného signálu. Na prostrednom obrázku je zobrazený výsledok autokorelácie, kde sme použili redukciu pomocou pohyblivého priemeru popísaného v predchádzajúcom odseku. Niekoľko zle nájdených hlasivkových tónov zmizlo ale stále ich tám ešte dosť veľa ostáva.

Teda prvý priemer, ktorý optimalizoval výšky tónov nestačí rovnako ako aj spomenuté kritérium. Tiež by sme teda chceli, aby sa hlasivkové tóny vyskytovali v zhruba rovnakých intervaloch. Nakoľko vieme predpokladať, že hlasivkové tóny sa vyskytujú zhruba každých 25 ms. Na začiatku pohyblivý priemer frekvencie nastavíme práve na túto hodnotu. Následne nájdeme prvý hlasivkový tón a môžeme priemer upravovať. Keďže každý hlas má svoju vlastnú frekvenciu vydávania tónov, je nutné aby sme tento priemer upravovali. V podstate



Obr. 2.4: Porovnanie vylepšovania autokorelácie. Hore bez kritérií, v strede s jedným spodný graf so všetkými kritériami

tým zabezpečíme oveľa väčšiu presnosť pri hladaní tónov a zamedzíme redundanciu jedného nájdeného tónu. Výsledok po aplikovaní týchto pravidiel môžeme vidieť na spodnom obrázku 2.4. Evidentne sme eliminovali všetky nesprávne nájdené hlasivkové tóny. To je výsledok, ktorý sme chceli dosiahnuť.

Keď sme našli hlasivkové tóny v tomto úseku signálu, môžeme sa posunúť na nový úsek. Nechceme sa však posunúť o celú dĺžku signálu. Stále existuje možnosť, že kvoli pohyblivému priemeru sme niektoré hlasivkové tóny vylúčili omylom. Preto sa posunieme len o jednu tretinu veľkosti kontrolovaného úseku a proces hľadania hlasivkových tónov opakujeme.

	Počet nájdených hlasivkových tónov (signál obsahoval 56 tónov)	**Úspešnosť**
Autokorelácia bez vylepšení	114	49.12%
Zákaz duplikátov + pohyblivý priemer funkčných hodnôt	70	80.00%
Zákaz duplikátov + pp funkčných hodnôt a frekvencie tónov	57	98.25%

Tabuľka 2.1: Štatistika počtov nájdených hlasivkových tónov porovnávajúca úspešnosť využitia pomocných kritérií

V tabuľke 2.1 môžeme vidieť štatistickú úspešnosť hľadania hlasivkových tónov na testovacej vzorke. Testovacia nahrávka použitá pri vytváraní štatistiky obsahovala práve 56 hlasivkových tónov. Bola to hláska „a". Bez kritérií je počet nájdených hlasivkových tónov čisto z autokorelácie, bez žiadneho z vyššie popísaných kritérií. Ako môžeme vidieť, nájdených hlasivkových tónov bolo až 114, čo je viac ako dvojnásobok reálneho počtu. Táto hodnota rozhodne nebola postačujúca a preto sme museli vymyslieť naše kritériá.

Druhý výsledok v tabuľke zahŕňa dve kritériá. Zákaz označenia duplikátov a pohyblivý priemer skutočných hodnôt hlasivkových tónov v signále. Tieto pravidlá priniesli značné zlepšenie. Vo výstupe bolo označených už len 70 pseudohlasivkových tónov. Stále je to ešte o 14 viac ako by bolo optimum označovania. Po pridaní druhého pohyblivého priemeru, sme mali vo výstupe označených 57 hlasivkových tónov. Jediný falošný hlasivkový tón bol označený na konci našej testovacej vzorky. Neskôr sa dozvieme, že to v podstate nevadí. 98,25% úspešnosť bola pre náš projekt postačujúca a tak sme venovali ďalším problémom.

Výstupom z tejto fázy analýzy signálu zaznamenanej reči je zoznam súradníc na časovej osi. Tieto súradnice označujú, v ktorých časových úsekoch sa nachádzajú hlasivkové tóny. Táto fáza ale označí len znelé spoluhlásky a samohlásky. Navyše sa často stáva, že táto fáza označí aj explozíva, ktoré ale rozhodne nechceme v našom procese nijako upravovať. Hlavný dôvod prečo nechceme tieto hlásky upravovať je ten, že pri explozívach je v signáli zvyčajne jeden bod, ktorý je na osi y výrazne vyššie ako ostatné. Keby sme tento bod pri skracovaní odstránili, stratili by sme podstatnú časť explozíva. To by spôsobilo ťažšie rozpoznanie hlásky ľudským sluchom. Predlžovanie explozív by znamenalo zduplikovanie tohto jedného bodu v nejakej vzdialenosti. Ak by bola vzdialenosť medzi pôvodným bodom a syntetizo-

vaným bodom príliš krátka, spôsobilo by to metalický efekt. Hlas by znel plechovo. Naopak ak by boli tieto body od seba príliš vzdialené, viedlo by to k duplikovaniu explozíva v reči. Napríklad pri hláske „k" by vo výstupnom signále bolo počuť niečo ako klikanie. Na druhej strane, nosové spoluhlásky a samohlásky sú pre predlžovanie a skracovanie zaznamenanej reči najdôležitejšie. Dôvodom je to, že signál týchto hlások sa najlepšie syntetizuje. Navyše sa tieto hlásky na základe ich správania vedia predlžovať takmer donekonečna.

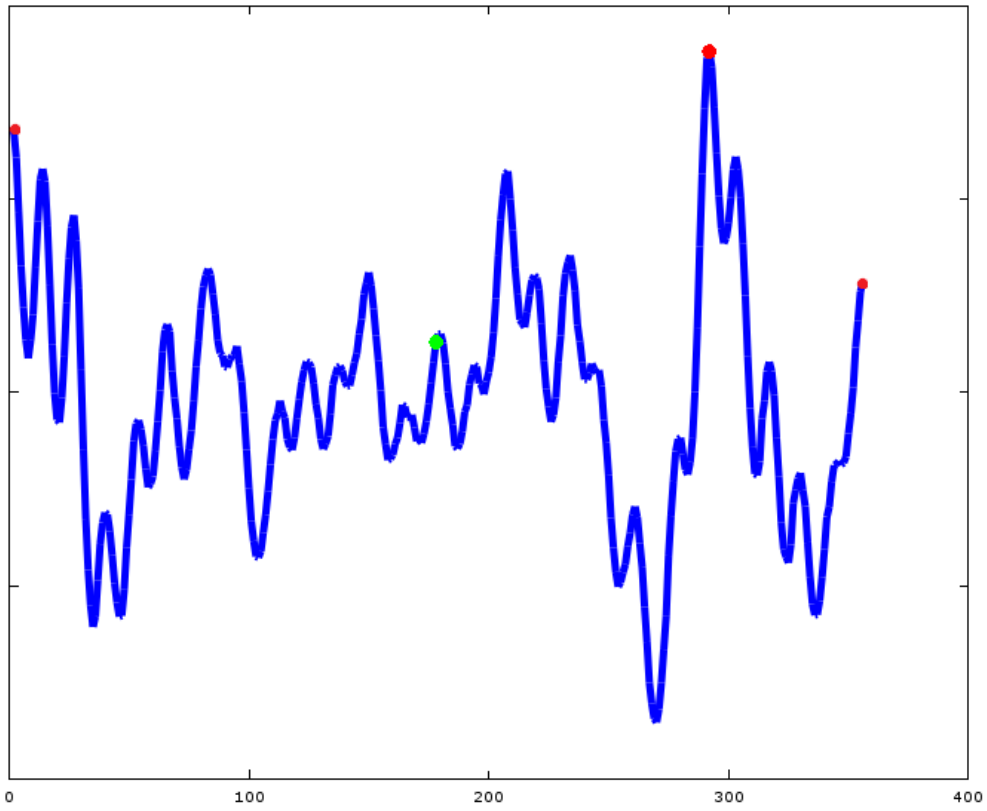
Frikatíva a ostatné hlásky, ktoré vznikajú len správnym vytlačením vzduchu cez hrdlo respektíve cez jazyk a zuby hlasivkové tóny neobsahujú. V nasledujúcej kapitole si ukážeme ako ich odlíšime od ticha a ako sme sa s takýmito hláskami vysporiadali pri predlžovaní signálu.

2.2 Úseky bez hlasivkových tónov

Po tom, čo sme našli hlasivkové tóny vo vstupnom signále, môžeme signál rozdeliť na menšie úseky. Vstup budeme rozdeľovať nasledovne. Pre každé tri po sebe idúce hlasivkové tóny zoberieme vektor signálu ležiaci medzi nimi. Tento vektor od konca o skrátime o jeden bit aby nám nevzniklo akési pretečenie. Na začiatku krátkeho vektora zvuku máme teda hlasivkový tón a na konci máme hlasivkový tón bez najvyššieho bodu. Niekde medzi nimi sa nachádza ďalší hlasivkový tón.

Na obrázku 2.5 môžeme vidieť úsek signálu, ktorý je výstupom z autokorelácie. Červený bod na signále je bod, ktorý označila vylepšená autokorelácia ako hlasivkový tón. Zelený bod na signále je stred výrezu. V samohláskach sú hlasivkové tóny pravidelné a v podobných grafoch výstupu sa vyskytujú takmer v strede. Rozdiel od úplného stredu je rozdiel zvyčajne len niekoľko bitov. Na obrázku je rozdiel cez sto bitov čo je príliš veľa. Takéto výstupné vektory sme poslali do funkcií na detekciu frikatív. Často bola odchýlka aj väčšia. Dlhá až niekoľko stoviek bitov. To znamenalo, že medzi nájdenými hláskami sa mohla vyskytovať nejaká, ktorú sme nedetekovali. V nasledujúcich kapitolách si popíšeme funkcie a ich vylepšenia, ktoré sme vytvorili aby sme zistili, či sa v takýchto blokoch nachádza frikatívum, explozívum alebo ticho. Výskyt samohlásky sme vylúčili vďaka úspešnosti, ktorú sme dosiahli v predchádzajúcej kapitole.

Každé dva po sebe idúce, takto rozdelené vektory budú mať spoločnú časť. Pravá časť prvého z dvojice vektorov bude rovnaká ako ľavá časť druhého z dvojice vektorov. Pri tomto rozdeľovaní sa totiž vždy posúvame práve o jeden hlasivkový tón.



Obr. 2.5: Červený vychýlený hlasivkový tón naznačuje, že na danom úseku sa môže vyskytnúť frikatívum

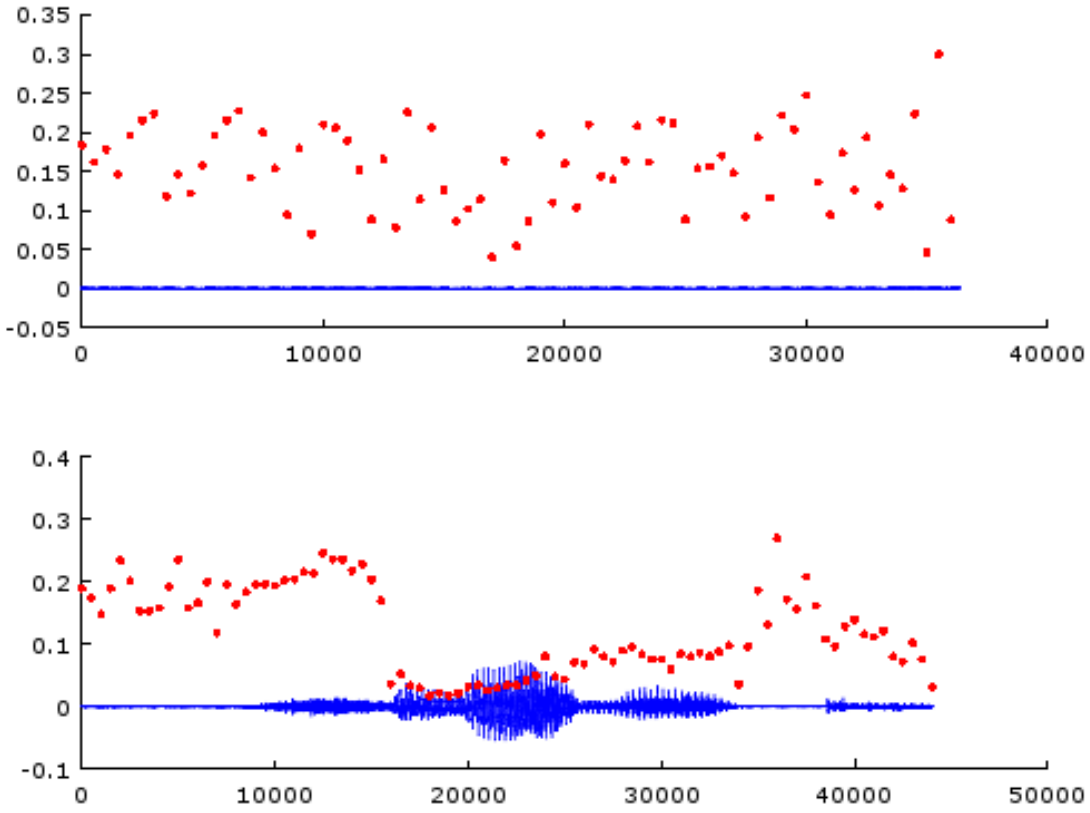
2.3 Zero-crossing rate a sigma

Zero-crossing rate je štatistická funkcia, ktorá sa pri digitálnom signále používa na sledovanie frekvencie, s ktorou signál prechádza cez nulu. V našej práci sme túto funkciu spolu so sledovaním rozptylu nameraných hodnôt využili na detekciu frikatív. Tieto spoluhlásky sú vytvárané pomocou súvislého vypúšťania vzduchu z našich pľúc, pričom ústami vytvárame tomuto vzduchu prekážku. Táto prekážka spôsobí pravidelné vibrácie. Tieto vibrácie následne vytvárajú zvuk frikatív ako je napríklad hláska „s”. Dôležitá je informácia o tom, že tento šum je relatívne pravidelný. Na spodnom obrázku 1.2 v kapitole 1.3.4 môžeme vidieť signál frikatíva. Evidentne tento signál nie je tak pravidelný ako signál samohlásky. Ale čo sa týka prechodov cez nulu je relatívne pravidelný. Narozdiel od ticha alebo šumu, ktoré sú vzhľadom na prechody nulou veľmi nepravidelné. Na obrázku 2.6 môžeme vidieť použité funkcie. Prvá z nich je zero-crossing rate a druhá je funkcia energie. Ako sme spomínali v predchádzajúcom odseku, rozptyl zero-crossing rate pri tichu a šume by mal byť vyšší ako pri frikatívach. Navyše pri hláskach s hlasivkovým tónom sa zero-crossing rate značne zníži.

$$zcr = \frac{\sum_{t=1}^{T-1} \mathbb{I}(s_t * s_{t-1} < 0)}{T-1}$$

$$\sigma^2(X) = E[(X - E[X])^2]$$

Obr. 2.6: Použité vzorce zero-crossing rate a sigma



Obr. 2.7: Porovnanie zero-crossing rate (červené bodky) pri tichu (hore) a pri reči (dolu)

V našej práci sme pri využití zero crossing rate rozdelili vstupný vektor na kratšie pravidelné úseky. Na týchto úsekoch sme potom spočítali zero crossing rate. Na obrázku 2.7 môžeme vidieť ukážky zero crossing rate. Namerané hodnoty sú červené bodky na oboch obrázkoch. Na hornom obrázku je vzorka slabého šumu. Ako môžeme vidieť namerané hodnoty sú značne rozhádzané. Naopak na spodnom obrázku je nahrávka, ktorá začína tichom a nasleduje reč. Reč začína frikatívom asi okolo bitu 8000. Nasleduje samohláska, ďalšie frikatívum, ktoré je tentoraz znelé. Ďalej znova ticho a šum. Poďme si teraz bližšie rozobrať čo sa deje pri meraní zero-crossing rate. Na začiatku môžme vidieť relatívne náhodne rozhádzané hodnoty s veľkým rozptylom. Akonáhle sa ale priblížime k osemtisícemu bitu, namerané hodnoty sa začnú usporiadavať a ich odchýlka náramne klesá. Pri samohláske okolo bitu 17000 sú namerané hodnoty tiež usporiadané, ale sú značne menšie ako pri

frikatívach. Nasleduje znelé frikatívum „z" okolo bitu 26000. Opäť môžeme sledovať usporiadané hodnoty s malou odchýlkou. Akonáhle sa dostaneme k tichu namerané hodnoty sa zase rozhádžu.

Môže sa zdať, že nám zero crossing rate spolu so sigmou nevie dobre odlíšiť samohlásky od frikatív. Ale na druhej strane dobre rozlýši hlásky od ticha a šumu. Nesmieme ale zabudnúť na to, že v tejto fáze už presne vieme povedať kde sa samohlásky v slove nachádzajú. Spojenie tejto informácie s výsledkom kombinácie zero-crossing rate sigma, dostaneme celkom kvalitnú informáciu o tom kde sa frikatíva v slove nachádzajú. Navyše to nepotrebujeme vedieť úplne presne. Pri predlžovaní frikatív nechceme vybrať tú časť signálu, kde frikatívum prechádza do inej hlásky. Predĺženie takéhoto úseku by mohlo znamenať, že vo výsledku by sa nám opakoval zvláštny zvuk tohto prechodu.

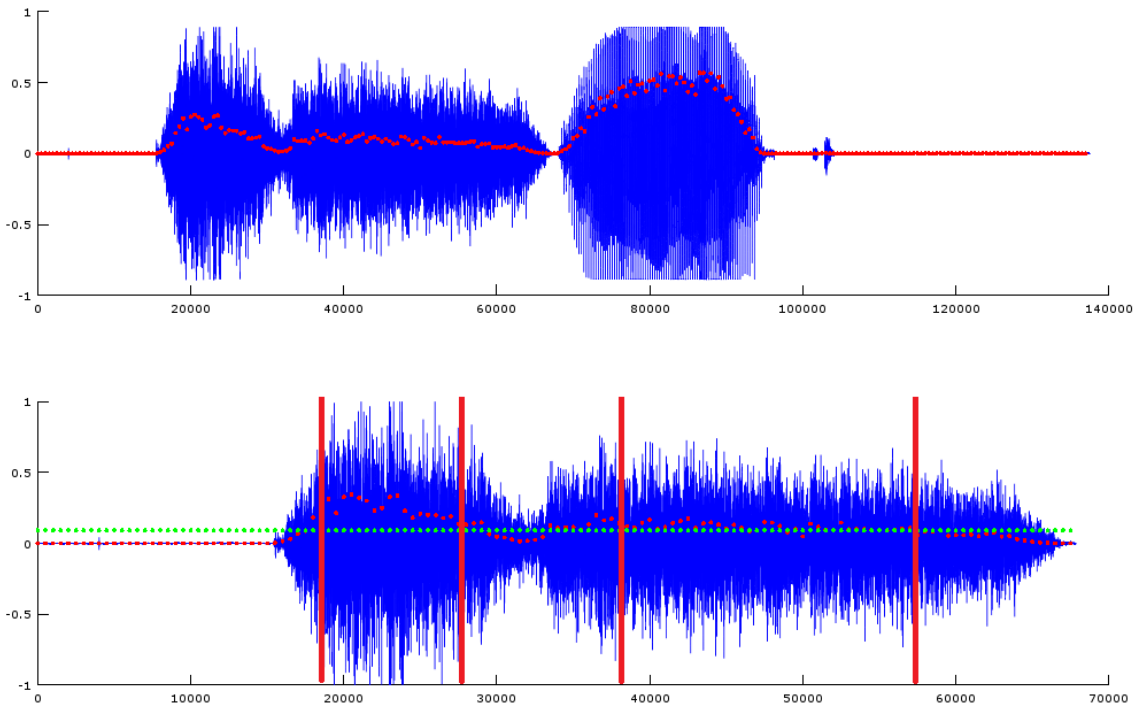
Výstupom z tejto fázy analýzy nášho vstupného signálu sú označené intervaly, na ktorých by sa mali nachádzať frikatíva.

Čo ale so slovami ako „včera" prvé dve hlásky tohto slova sú frikatíva. Potrebovali by sme ich nejako odlíšiť a nespojiť ich do jednej hlásky. Použitím zero-crossing rate a sigmy by sa nám to mohlo stať. Vo výsledku by to viedlo k vytvoreniu slova, ktoré by znelo ako „včvčera". Ako sme tento problém vyriešili sa dozvieme v ďalšej podkapitole.

2.4 Energia

Amplitúda signálu sa mení. Môžeme to vidieť napríklad na hornom z dvojice obrázkov 2.8. Na obrázku vidíme krátku testovaciu frázu, ktorá na začiatku obsahuje ticho respektíve šum, nasledujú frikatíva „čš", samohláska, explozívum a ticho. V testovacom prípade sme frikatíva nevybrali náhodne. Snažili sme sa vybrať také, ktoré sa jednoducho vyslovujú za sebou. Najkratší prechod z jedného frikatíva na druhé mali práve „č" a „š". Červené bodky obrázku znázorňujú výsledok krátkodobých energií tohto signálu. Ďalej si všimnime, že energia pri samohláske je oveľa vyššia ako pri šume, explozívach a frikatívach. Táto informácia by nám ale pri detekcii hlások s hlasivkovým tónom veľmi nepomohla, lebo sme potrebovali nájsť samotné hlasivkové tóny.

Spodný signál z obrázku 2.8 sme získali pomocou predchádzajúcej kapitoly. Zero-crossing rate nám vrátil interval $< 19833, 67493 >$. Pre lepšiu prehľadnosť testu sme nezobrali len túto časť intervalu, ale celý signál od nultého bitu po bit 67493.



Obr. 2.8: Hore porovnanie energie frázy obsahujúcej hlásky „čša" Dolu využitie priemernej energie signálu na detekciu viacerých frikatív vo fráze „čš"

$$E_X = \sum_{m=-1\infty}^{\infty} X(m)^2 \tag{2.1}$$

Amplitúdu signálu reči vieme dobre sledovať pomocou krátkodobej energie signálu. Najjednoduchším výpočnom energie signálu, je suma druhých mocnín jednotlivých bitov vektora digitálneho signálu v čase (vzorec 2.1). V našom riešení sme ju využili na spresnenie výsledkov spojenia Zero-crossing rate a sigmy. Podobne ako pri zero-zero crossing rate, sme na začiatku rozdelili signál na menšie časti a na týchto sme spustili meranie krátkodobej energie signálu. Ako môžeme vidieť na spodnom obrázku 2.8, v okolí bitov 30000 je prechod medzi spomenutými frikatívami. Energia tento prechod krásne zachytila a v tomto čase môžeme vidieť jej náhly pokles a následný rast.

Energiu sme využili na úsekoch signálu, o ktorých sme predpokladali, že obsahujú frikatívum na základe kapitoly 2.3. Na obrázku 2.8, na úseku bitov z intervalu $< 20000, 60000 >$, môžeme vidieť dve po sebe idúce frikatíva. Na spodnom obrázku vidíme ich energiu. Všimnime si teraz, že pri prechode z jedného frikatíva na druhé, energia signálu klesne. Na spodnom z dvojice obrázkov je vidieť naše riešenie detekcie rozdielnych frikatív idúcich po sebe. Signál sme si podobne ako v predchádzajúcej kapitole rozdelili na krátke časové úseky dĺžky 25ms. Na týchto úsekoch sme spočítali krátkodobú energiu signálu. Z vektora výsledkov sme

získali jeho priemernú hodnotu. Túto hodnotu môžeme vidieť v podobe zelených bodiek na spodnom obrázku z dvojice 2.8. Keby sme mali na celom úseku len jedno frikatívum, hodnoty energie by poskakovali okolo priemeru. Podobne, ako to môžeme vidieť na intervale $< 50000, 60000 >$. Spodný obrázok sa na tomto úseku môže zdať mierne klamlivý, lebo zelené bodky priemeru prekrývajú červené bodky energie. Samotná detekcia 2 frikatív prebieha nasledovne. Na vektore energií nájdeme dostatočne dlhý súvislý úsek hodnôt väčších ako je priemer. Ak taký neexistuje, vieme s istotou povedať, že na danom signále je len jedno frikatívum. Je to tak preto, lebo výsledky hodnôt energie len poskakujú okolo priemeru. Ak taký je, mal by za ním nasledovať súvislý úsek hodnôt menších ako priemer. Ak taký nie je, tak sme zase mali na vstupe len jedno frikatívum. Ak nájdeme aj tento úsek, mal by nasledovať ďalší súvislý úsek nad priemerom. Ak nájdeme tento posledný úsek, znamená to, že sme našli dve frikatíva.

Stále však ešte nemáme intervaly, na ktorých sa frikatíva nachádzajú. Pri fáze syntézy by sme nechceli duplikovať prechod medzi frikatívami. Vo výsledku by to neznelo dobre. Lokáciu tohto úseku zhruba poznáme. Nachádza sa v intervale, kde hodnoty energie klesli pod priemer. Na základe tejto informácie vieme dobre definovať interval druhého frikatíva. Zoberieme z energií poslednú hodnotu zo súvislého úseku pod priemerom. Túto hodnotu odčítame od celkovej dĺžky vektora výsledok označme D. Zoberieme 20% z D a bezpečný interval hľadaného frikatíva bude ležať vo vnútorných 60% vektora za podpriemerným intervalom. Na spodnom obrázku 2.8 je tento interval zvýraznený dvoma červenými úsečkami napravo. Teda výsledný interval bude ležať medzi posledným bodom podpriemerného vektora predĺženého o D a koncom vstupu skráteného o D. Vektor sme sa rozhodli kvoli bezpečnosti skrátiť tiež, aby sme sa vyhli prechodu frikatíva na inú hlásku.

Prvé frikatívum budeme definovať podobne. Podotýkame, že výsledok zero-crossing rate bol bez úvodného ticha. Teda náš pracovný vstup reálne začínal okolo bitu 16000. Zoberieme prvý bod z podpriemerného intervalu. Následne získame 20% z jeho hodnoty na osi x. A bezpečná časť frikatíva bude znova ležať vo vnútorných 60% pracovného vstupu. Na spodnom obrázku môžeme výsledok vidieť označený ľavými dvoma červenými úsečkami.

Výsledkom tejto fázy je teda vylepšenie detekcie frikatív v signále. V predchádzajúcej kapitole sme získali intervaly kde sa frikatíva vo všeobecnosti nachádzajú, ale mali sme problém so slovami kde nasleduje viacero firkatív za sebou. Táto metóda tento problém odstránila. Pomerne dlhé vektory rozdelila a skrátila tak aby boli frikatíva od seba dostatočne vzdialené.

Nájdením hlasivkových tónov a bezpečne duplikovateľných úsekov frikatív sa fáza analýzy skončila. Vstupný signál sme v tejto fáze najprv rozdelili na menšie úseky nájdením hlasivkových tónov v kapitole 2.1. Následne sme vstup rozdelili na kratšie úseky, ktoré buď obsahovali hlasivkové tóny alebo boli podozrivé. Podozrivé úseky vstupu sme následne analyzovali pomocou zero-crossing rate a sigmy aby sme zistili či ide o ticho, explozívum alebo frikatívum. Ak išlo o frikatívum, tieto úseky sme ďalej analyzovali v tejto kapitole. Analýzou pomocou vylepšenia funkcie energie sme detekovali, či sa v daných úsekoch nachádza viac ako jedno frikatívum. Ak bolo toto podozrenie pravdivé, daný úsek sme rozdelili tak aby sme získali intervaly, ktoré budeme môcť bezpečne syntetizovať.

2.5 Syntéza signálu s hlasivkovými tónmi využitím PSOLA algoritmu

V tejto kapitole si ukážeme ako sme signál syntetizovali. Vstupom je množstvo malých úsekov signálu, ktoré máme pooznačované v troch skupinách. Prvou najväčšou skupinou je skupina úsekov s hlasivkovým tónom. Tieto vektory sú úseky vstupu medzi tromi hlasivkovými tónmi, skrátené o jeden bit od konca. Vo vektore je vždy od maxima klesajúca časť prvého hlasivkového tónu, celý druhý a rastúca časť bez maxima tretieho hlasivkového tónu. Druhá skupina vektorov je skupina frikatív. Tieto vektory sú skrátené na úroveň kde sa budú dať jednoducho a bezpečne syntetizovať. Treťou skupinou sú odpadové vektory. Teda také, ktoré nechceme syntetizovať. Či už je to ticho, šum, prechody medzi hláskami alebo nebezpečné časti frikatív. Tieto vektory sa nedajú popísať ako predchádzajúce dve skupiny lebo zahŕňajú viacero diametrálne rozličných skupín. Podstatná je informácia, že ich syntézou pomocou našich algoritmov by sme získali hlavne veľa problémov a málo výstupu. Týchto vektorov je totiž najmenej a ich celková dĺžka je najkratšia. Pri týchto tvrdeniach sme vychádzali z výskumu zastúpenia foném [10]. Na základe výsledkov, ktoré výskum priniesol sme zistili, že zastúpenie hlások v slovenčine je nasledovný.

- Samohlásky 37,1%
- Frikatíva 37,09%
- Ostatné 25,89%

Ďalej treba brať do úvahy, že dĺžka samohlások a frikatív je oveľa väčšia ako explozív. V priemere sú explozíva v slovách o 40% kratšie ako ostatné hlásky. Tieto informácie potvrdzujú naše tvrdenie, že syntézou problematických blokov by sme získali veľa problémov za malý prínos pre výsledok.

V kapitole 1.3.7 sme načrtli ako pitch synchronous overlap and add funkcia funguje. V tejto kapitole algoritmus dôslednejšie popíšeme a uvedieme aj naše vylepšenia.

Syntéza hlasivkového tónu Syntéza hlasivkového tónu prebieha nasledovne. Porovná sa veľkosť ľavej a pravej strany, tento krok sa robí preto, že sa snažíme zachovať vzdialenosť nášho prostredného hlasivkového tónu a nášho predchádzajúceho, teraz prvého, hlasivkového tónu v bloku. Ak je ľavá strana inej dĺžky ako pravá, znamená to buď klasické kolísanie ľudského hlasu alebo, že sa začína meniť melódia vety. Ďalej ak je ľavá strana dlhšia, aby sme mohli zachovať vzdialenosť hlasivkových tónov vo výstupnom signále, musíme pravú stranu predĺžiť o rozdiel ľavej a pravej strany. Pre jednoduchosť sme pravú stranu predlžovali nulami. Inak ak je ľavá strana kratšia, musíme pravú stranu bloku skrátiť o rozdiel ľavej a pravej strany.

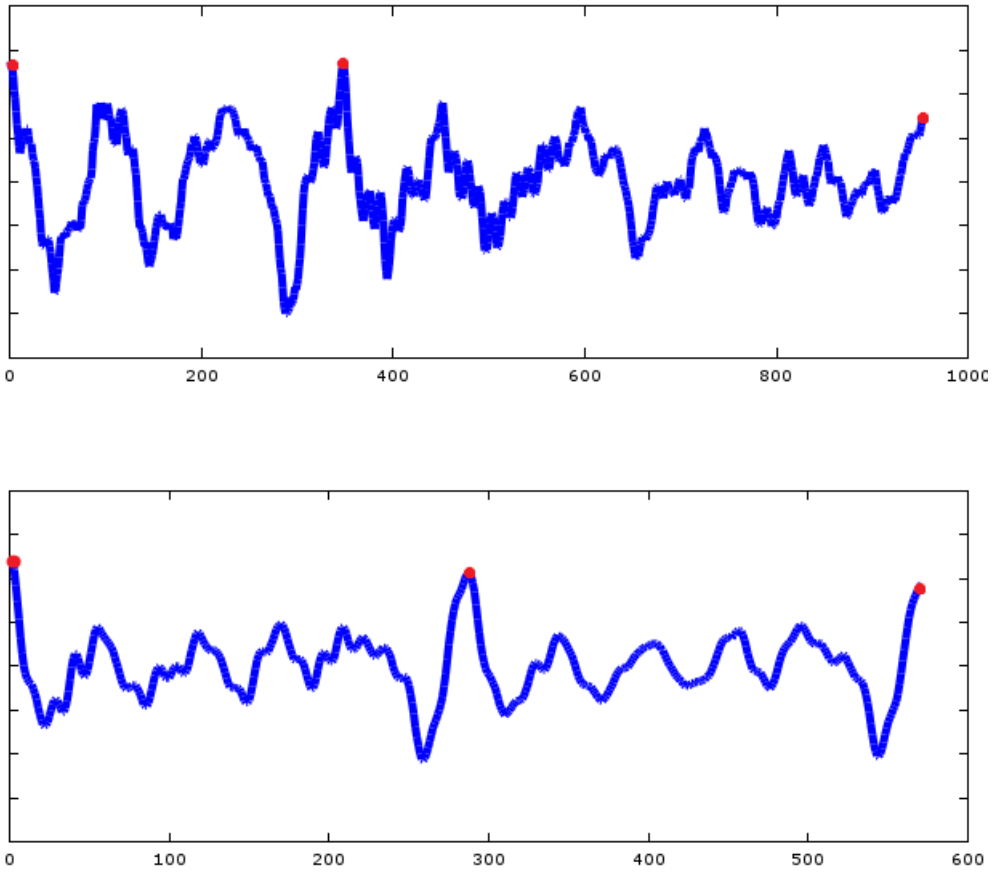
Teraz keď máme pripravené všetky potrebné časti môžme začať skladať signál. Štandartne sa reč pri PSOLA algoritme syntetizuje tak, že zoberieme pravú stranu a postupne lineárne znižujeme jej vplyv na výsledný signál, počínajúc v hodnote 1 a končiac v 0. Analogicky zosilňujeme vplyv ľavej strany. Štandartne sa tento krok zosilovania a zoslabovania robí podľa vzorca 2.2

$$hvs[i] = hsLS[i] * \left(\frac{i}{n}\right) + hsPS[i] * \left(1 - \frac{i}{n}\right) \tag{2.2}$$

Kde i je naša súčasná pozícia a je definovaná na intervale $< 1, rozdiel\_stran >$ a n je rozdiel ľavej a pravej strany. Ďalej hvs znamená „hodnota výstupného signálu" $hsL(P)S$ znamená „hodnota signálu ľavá (pravá) strana".Podľa hlasivkového tónu sme rozdelili blok na ľavú a pravú stranu. Teda výsledný signál vytvorený pomocou PSOLA bude vyzerať nasledovne. Ľavá strana pôvodného bloku, ľavá + pravá strana (upravené pomocou zosiľňovania a zoslabovania), pravá strana pôvodného bloku. Za predpokladu, že máme približne rovnakú dĺžku ľavej a pravej strany, môžeme strednú časť donekonečna duplikovať a bude vždy presne zapadať do frekvencie hlásky.

Takto funguje pitch synchronous overlap and add bez akýchkoľvek vylepšení. Vo výstupe ale syntetizovaný signál neznel veľmi dobre. Občas sa stávalo, že hlas rečníka dostal metalický nádych. Veľmi často sa to stávalo na krátkych úsekoch. Hlas rečníka v podstate na krátko zakolísal. Po analýze problému sme zistili, že hlas kolísal práve na tých miestach, kde mal syntetizovaný úsek signálu vychýlený stredný hlasivkový tón. Skúsili sme nasimulovať vychýlenie doľava aj doprava a tieto vektory syntetizovať. Na základe subjektívneho testu

našim sluchom sme zistili, že ak je odchýlka stredného hlasivkového bodu od stredu vyššia ako 10%, nastane pri syntéze počuteľné zakolísanie.



Obr. 2.9: Vektory, pripravené na syntézu pomocou PSOLA algoritmu. Červené body označujú hlasivkové tóny

Po analýze samotnej PSOLA funkcie sme zistili, že jediná možnosť ako by sa tomuto fenoménu dalo vyhnúť, je nejakým spôsobom upraviť funkciu vytvárajúcu stredný úsek syntetizovaného vektora. Poďme si teraz pripomenuť ako sa signál pomocou tohto algoritmu syntetizuje. Vstupom je vektor signálu, ktorý obsahuje práve 3 hlasivkové tóny. Z toho prostredný je úplný, teda má aj kompletnú ľavú a pravú stranu a ďalšie dva sú polovičné. Na obrázku 2.9 môžeme vidieť dva vektory, ktoré obsahujú hlasivkové tóny. Stredný hlasivkový tón je označený červenou bodkou. Spodný z dvojice vytvorí po syntéze pomocou PSOLA hlásku bez mutácie. Naopak vrchný z dvojice vytvorí spomenutú mutáciu hlasu.

Pri analýze vzorca 2.2 sme zistili, že lineárny pokles a rast vplyvu ľavej a pravej strany nemusí byť vždy prínosom. Tento prístup funguje dobre pri vektoroch ako je spodný na obrázku 2.9. Ak je ale stredný hlasivkový tón vychýlený, linearita vzorca spôsobí vznik

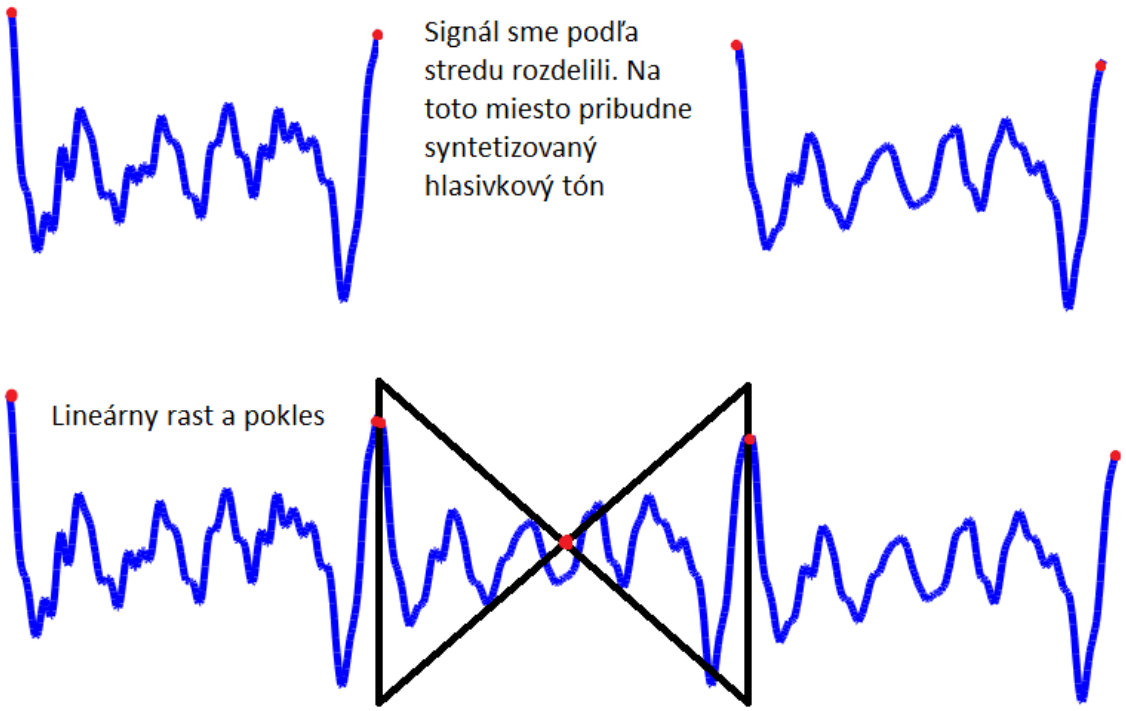
mutácie. Rozhodli sme sa teda definovať funkciu, ktorá bude generovať vzorec na výpočet vplyvov ľavej a pravej strany.

$$\begin{aligned} f(x) &= x^a \\ \log_{S(v)}(S(w)) &= a \\ S(v) &= \frac{v}{length(vektor)} \\ g(x) &= 1 - f(x) \end{aligned} \tag{2.3}$$

V odvodení 2.3 môžeme vidieť odvodenie funkcie, ktorú sme neskôr použili. Potrebovali sme nájsť striktne monotónnu funkciu, ktorá bude prechádzať bodmi 0 a 1, pričom jej funkčná hodnota v bode 0 je nula a v bode 1 je jedna. Tieto parametre môžu mať polynomiálne funkcie. My sme pre zjednodušenie riešenia z množstva polynomiálnych funkcií vybrali hornú funkciu v predchádzajúcom odvodení. V tejto funkcii poznáme premennú x, ktorá je v našom prípade čas respektíve súčasná pozícia na spracovávanom vektore. Potrebovali sme do vzorca zakomponovať vzťah stredu vektora vzhľadom na náš nájdený hlasivkový tón. Na základe tohto vzťahu vypočítame parameter a pomocou druhého vzorca. Pričom $S(v)$ je stred vektora a $S(w)$ je pozícia nášho nájdeného hlasivkového tónu. Tieto pozície vypočítame vzhľadom na celú dĺžku vektora. Potrebovali by sme totiž hodnoty z intervalu $< 0, 1 >$. Túto funkciu môžeme vidieť v treťom vzorci v odvodení 2.3. Keď získame parameter a, pomocou prvého vzorca získavame funckiu na výpočet rastu vplyvu. Funkciu na pokles vplyvu už získame jednoducho odčítaním funkčných hodnôt funkcie $f(x)$ od 1. Ako môžeme vidieť vo štvrtom vzorci odvodenia.

Dôležitá vlastnosť nášho generátora je, že pri nulovej vzdialenosti stredu vektora a nájdeného maxima, nadobúda parameter a hodnotu 1. Teda v prípadoch kedy by bol nájdený hlasivkový tón presne v strede, dostaneme lineárnu funkciu na rast a pokles, ktorá bola použitá v pôvodnom algoritme.

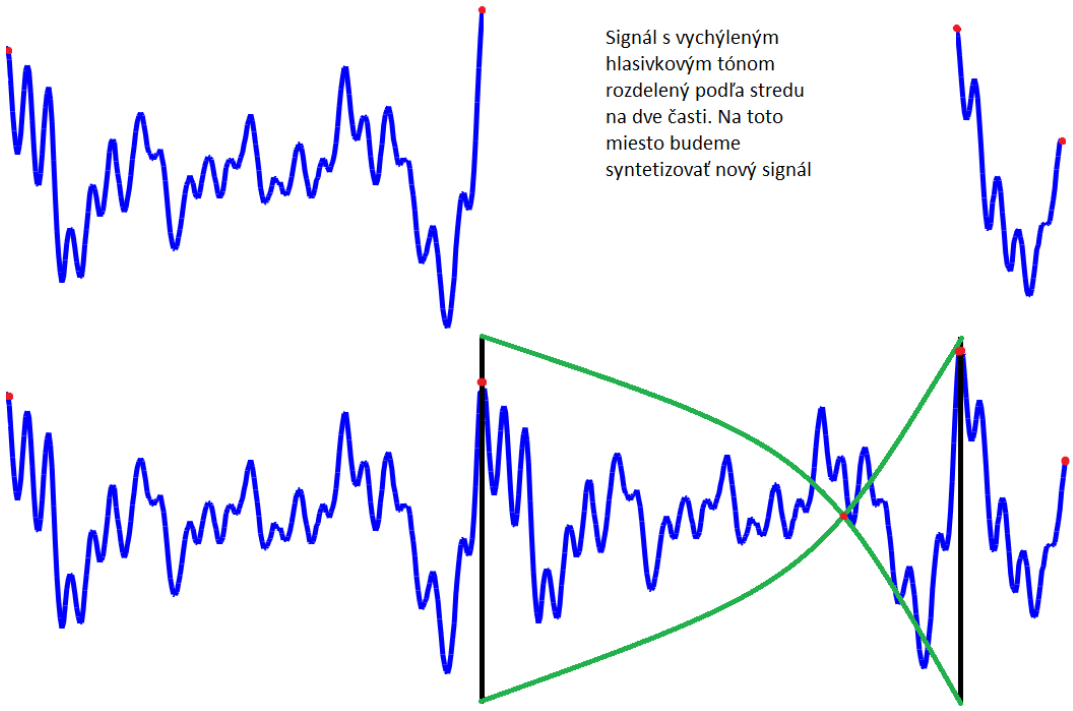
Na obrázku 2.10 môžeme vidieť využitie funkcie 2.3. Na hornom z dvojice zobrazených signálov vidíme pôvodný signál rozdelený na polovice podľa stredného hlasivkového tónu. Pôvodný signál mal hlasivkový tón úplne v strede vektora. Na dolnom obrázku vidíme aká funkcia bola použitá. Výstupom z nášho generátora na funckie rastu a poklesu bola lineárna funkcia. V tomto prípade je vyhovujúce, že vplyv ľavej a pravej strany rastie a klesá lineárne. Pri pozícii hlasivkového tónu v strede sme chceli zachovať pôvodnú funckiu, čo sa nám úspešne podarilo dosiahnuť.



Obr. 2.10: Vizualizácia lineárrneho rastu a poklesu vplyvu jednotlivých zložiek vstupujúcich do syntézy hlasivkových tónov

Na obrázku 2.11 môžeme vidieť ďalšie využitie nášho generátora funkcií rastu a poklesu. Na hornom z dvojice môžeme vidieť vstupný signál rozdelený podľa hlasivkového tónu. Teraz je hlasivkový tón značne vychýlený smerom doprava od stredu. Takže pravá strana signálu tvorí len zlomok dĺžky ľavej strany. Náš generátor vytvoril na základe vstupu nové funkcie rastu a poklesu. Ich vizualizáciu môžeme vidieť na spodnom obrázku. Evidentne už nejde o lineárne funkcie. Tak ako sme predpokladali, vplyv ľavej strany rastie zo začiatku intervalu pomalšie a neskôr rastie rýchlo. Tým získame správanie aké sme pri tejto fáze syntézy potrebovali.

Teraz ozrejmíme prečo je toto správanie dôležité. Ak je stred vychýlený na pravú stranu, znamená to, že pravú stranu budeme musieť predĺžiť nulami. Pri lineárnom raste vplyvu aj tieto nuly spôsobovali, že hlas mutoval. Ak bolo vychýlenie príliš veľké, vplyv pravej strany skreslil jednotlivé bity a tým modifikoval hlas. Keď ale použijeme našu funkciu na rast a pokles, v neskorých bitoch, kde pravá strana mala príliš vysoký vplyv, bude jej vplyv znížený a tým sa výsledok bude viac podobať hlasu rozprávača.

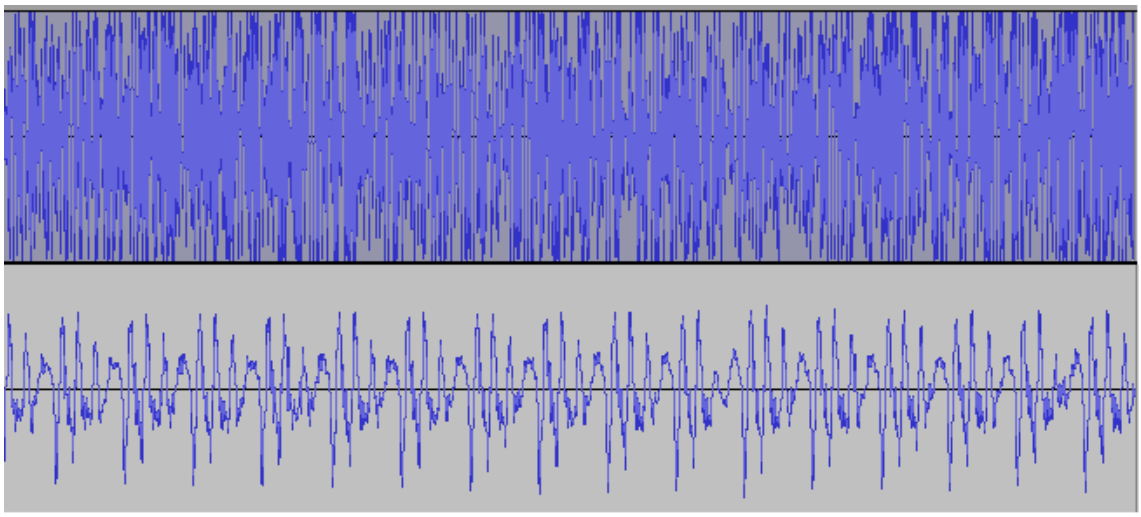


Obr. 2.11: Vizualizácia nelineárrneho rastu a poklesu vplyvu jednotlivých zložiek vstupujúcich do syntézy hlasivkových tónov

Naopak ak je ľavá strana kratšia, pravú stranu skrátiť tiež o rozdiel týchto strán. Táto modifikácia zase môže spôsobiť nepríjemnosti tým, že signál pravej strany bude skrátený natoľko, že bude ovplyvňovať signál ľavej strany príliš dlho a intenzívne. V tomto prípade by sme chceli intenzitu vplyvu pravej strany znížiť čo najrýchlejšie. Táto modifikácia by prebiehala analogicky k tretiemu obrázku 2.10. Avšak by sa vplyv presunul na ľavú stranu od stredu. Tento presun by spôsobil zvýšené tempo poklesu vplyvu pravej strany a zároveň zvýšené tempo rastu vplyvu ľavej strany.

2.6 Syntéza signálu bez hlasivkových tónov

V predchádzajúcej kapitole sme si ukázali ako syntetizovať signál pomocou PSOLA algoritmu. Žiaľ, PSOLA algoritmom vieme syntetizovať len signál s hlasivkovými tónmi. Pri frikatívach nevieme signál syntetizovať pomocou tohto algoritmu, lebo signál frikatív neobsahuje žiadne pravidelné lokálne maximá. Teda by sme nemohli robiť synchronizáciu pomocu hlasivkových tónov v predchádzajúcej kapitole.



Obr. 2.12: Rozdiel medzi signálom frikatíva (hore) a hlásky s hlasivkovým tónom (dolu)

Na obrázku 2.12 môžeme vidieť porovnanie dvoch typov signálov, ktoré sme v tejto práci syntetizovali. Spodný z dvojice signálov reprezentuje signál hlásky s hlasivkovým tónom a spodný z dvojice reprezentuje signál frikatíva. Pri syntéze hlasivkových sme sa mohli oprieť o ich pravidelnosť. Na jej základe, sme signál pomocou PSOLA algoritmu syntetizovali. Ako ale môžeme vidieť na hornom obrázku, frikatíva nie sú nijako periodické a šum je značne chaotický.

To nám ale v podstate veľmi neprekáža, ba naopak. Suchopárny signál frikatív nám poskytuje jednoduchý spôsob ich duplikácie. V kapitolách 2.3 a 2.4 sme si ukázali ako frikatíva v signále detekovať. Zároveň sme z tejto fázy analýzy získali oddelené súvislé vektory signálu. O týchto vektoroch vieme prehlásiť, že zaručene obsahujú práve jedno a len jedno frikatívum. Navyše je tento vektor skrátený tak, aby neobsahoval prechody medzi jednotlivými fonémami.

Postup pri syntéze frikatív je veľmi jednoduchý. Na vstupe dostaneme vektor pôvodného frikatíva. Teoreticky by sme mohli celý tento vektor zobrať a pridať ho dva alebo viac krát do výstupného signálu. Tým by sme získali viacnásobnú dĺžku daného frikatíva a výstup by bol relatívne kvalitný. My sme výstup ešte mierne vylepšili. Pred samotnou duplikáciou si vstupný vektor rozdelíme na dve časti. Na začiatku zistíme, či vstupný vektor na prvých bitoch klesá alebo rastie. Ďalej zistíme funkčnú hodnotu prvého bitu vektora. Označme túto hodnotu A. Teraz ak signál rástol, začneme od konca skracovať a jednotlivé bity si odkladať do pomocného vektora. Pre každý bit skontrolujeme jeho funkčnú hodnotu B. Ak je odchýlka A a B menšia ako 10%, pozrieme sa na bit, ktorý je v pôvodnom vektore na pozícii o jedna menšej ako B. Ak je funkčná hodnota tohto bitu menšia ako B, znamená to, že signál na tomto úseku rástol a môžeme prestať so skracovaním.

Analogicky by sme postupovali ak by sme pri prvom bite pôvodného signálu sledovali klesanie. Kontrola bitu B by potom vyzerala tak, že bit predchádzajúci B by musel mať vyššiu hodnotu.

Vstupný signál máme po tejto fáze rozdelený na dve časti. Prvá časť je porovnateľne väčšia ako druhá. O prvej časti vieme povedať, že pri pridávaní signálu do výstupu, bude prechod medzi prvým a posledným bitom malý. Tým pádom bude aj málo počuteľný. Navyše keď za takto multiplikovaný signál pripojíme druhý kratší vektor, signál bude plynulo pokračovať ako vo vstupe. Takže tento prechod bude úplne nepočuteľný. Ďalšie plus je, že celkový syntetizovaný vektor môžeme vložiť do výstupu bez problémov. Presne zapadne do miesta odkial sme ho vystrihli. Toto je spôsobené tým, že prvý aj posledný bit zostal pri syntéze zachovaný z pôvodného vstupu. Takže presne nadviaže do sekvencie v pôvodnom signále.

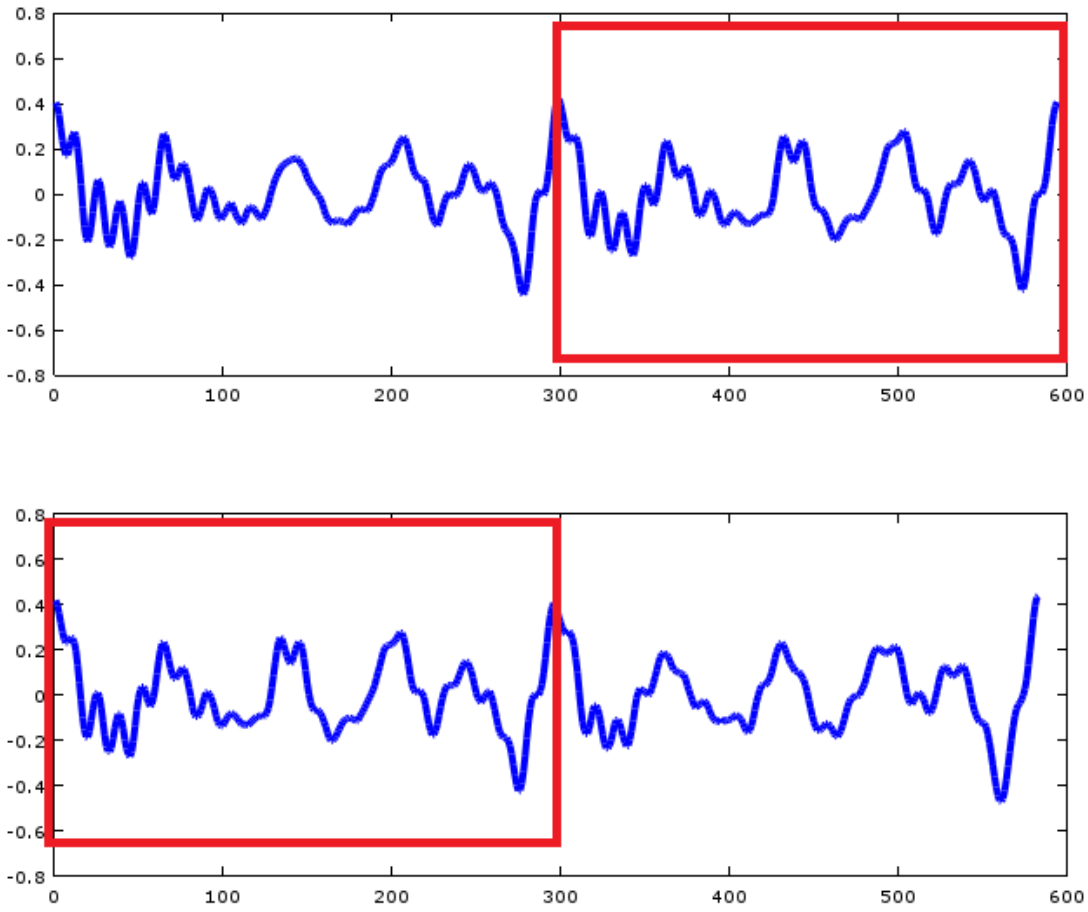
2.7 Spájanie syntetizovaných úsekov dohromady

V predchádzajúcich kapitolách sme si ukázali, ako sme jednotlivé duplikovateľné zložky analyzovali. V tejto kapitole si ukážeme ako sme výsledné vektory spájali dohromady.

Všetky vektory, ktoré sme vytvorili mali spoločné to, že ležali medzi troma hlasivkovými tónmi. Tento fakt budeme využívať na ich celkové spojenie do výsledného signálu.

Na obrázku 2.13 môžeme vidieť dva po sebe idúce vektory vstupujúce do syntézy. Pre prehľadnosť obrázku sme vybrali vektory pred syntézou. Zvýraznené časti vektorov označujú prekryv. Teda každé dva po sebe idúce vektory majú tieto časti rovnaké. Pri syntéze teda treba jeden z dvoch syntetizovaných vektorov pri skladaní výstupu skrátiť. My sme si pri syntéze vybrali skracovanie zľava, teda ľavú časť skladaných vektorov vždy skrátime o úsek medzi začiatkom vektora a prvým hlasivkovým tónom.

Skladanie syntetizovaných vektorov bude teda prebiehať nasledovne. Prvý výstupný vektor pridáme do výsledku tak ako je, bez skrátenia. Pre všetky ďalšie vektory skrátime ľavé strany a jednoducho ich zložíme do výstupného vektora. Keďže sa strany prekrývali, nevzniknú žiadne problémy s hlasom rečníka. Tým získame v podstate pôvodný signál, rozšírený o syntetizované časti z predchádzajúcich častí. Keby sme pre toto skladanie použili vstupné úseky vektorov, dostaneme pôvodnú nahrávku.

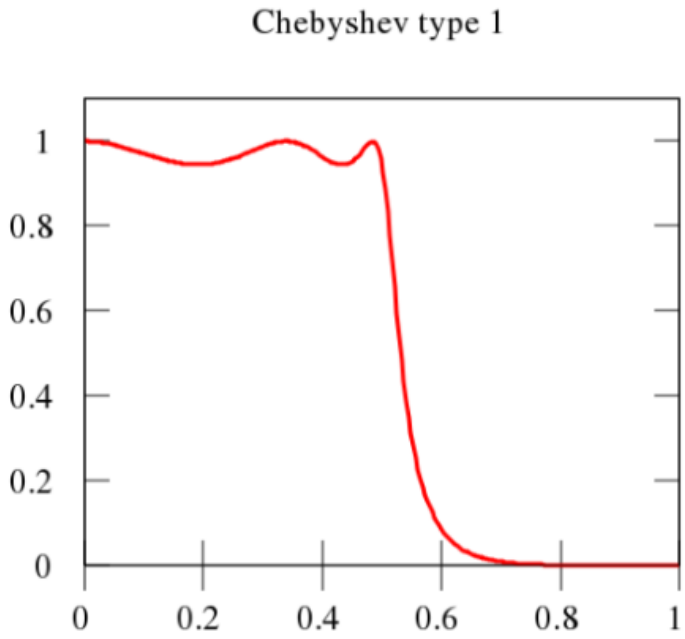


Obr. 2.13: Dva po sebe idúce vystrihnuté vektory. Vyznačené časti označujú prekryv

2.8 IIR filtrovanie signálu

V predchádzajúcich kapitolách sme si ukázali, ako sme signál reči analyzovali a následne syntetizovali. Počas syntézy signálu vznikli vo výsledku chyby spôsobené šumom. Napríklad syntéza časti slova, kde na vstupe celá trieda preložila stranu v čítanke, spôsobila zvýraznenie šumu. Aby sa tento šum stratil v hlase rečníka, na výstup sme použili IIR filter. Filtre fungujú tak, že určitú časť signálu potláčajú a tým pádom sa vo výstupe môže stratiť. IIR filtre sú takzvané nekonečné filtre. Nekonečné sú z toho dôvodu, že na špeciálny typ signálu impulz majú nekonečnú odozvu. Impulz je typ signálu, ktorého vektor začína nulami, potom nasleduje práve jedna jednotka a opäť samé nuly. IIR filtre sú nekonečné z toho dôvodu, že ich výstup sa znova posiela do filtra. Teda na impulz nemajú konečnú odozvu. Z tejto vlastnosti vznikla aj skratka IIR teda infinite impulse response.

IIR filtrov existuje veľké množstvo a každý z nich má rôzne vlastnosti v potláčaní nežiadúceho signálu. My sme si vybrali filter typu Chebyshev 1, lebo mal pre nás najlepšie vlastnosti.



Obr. 2.14: Spôsob potláčania nežiadúcich frekvencií pomocou IIR filtra typu Chebyshev 1. Potlačeniu predchádza krátke zakolísanie a po potlačení nenasleduje ozvena

Na obrázku 2.14 môžeme vidieť spôsob potláčania nežiadúcich frekvencií pomocou IIR filtra typu Chebyshev 1. Predtým ako nastane relatívne prudké potlačenie signálu (okolo 0,5 na obrázku), nastáva mierne zakolísanie, ktoré ale nie je pre poslucháča počuteľné. Po potlačení signálu nenasledujú ozveny, ktoré by poslucháč už započuť mohol. Tieto vlastnosti sú dôvodom nášho výberu tohto filtra.

Použitie IIR filtra typu Chebyshev 1 s parametrami 1000Hz, rád 10 a 6dB upraví vstupný signál nasledovne. Pred každým výskytom frekvencie 1000Hz a viac signál zakolíše o zhruba 0,5 dB a následne prudko klesne o 6 decibelov. Prasknutie alebo chyba, ktorú sme našli pomocou tohto filtra sa síce neodstráni, ale jej počuteľnosť sa značne zníži. Praskanie a ťukanie sa vo výstupnom signále počas čítania stratí v hlase rečníka. Ďalej sa tiež zachová všetko čo sme doteraz dosiahli. Teda hlas rečníka ostane nepoškodený. To bolo cieľom tejto práce.

Kapitola 3

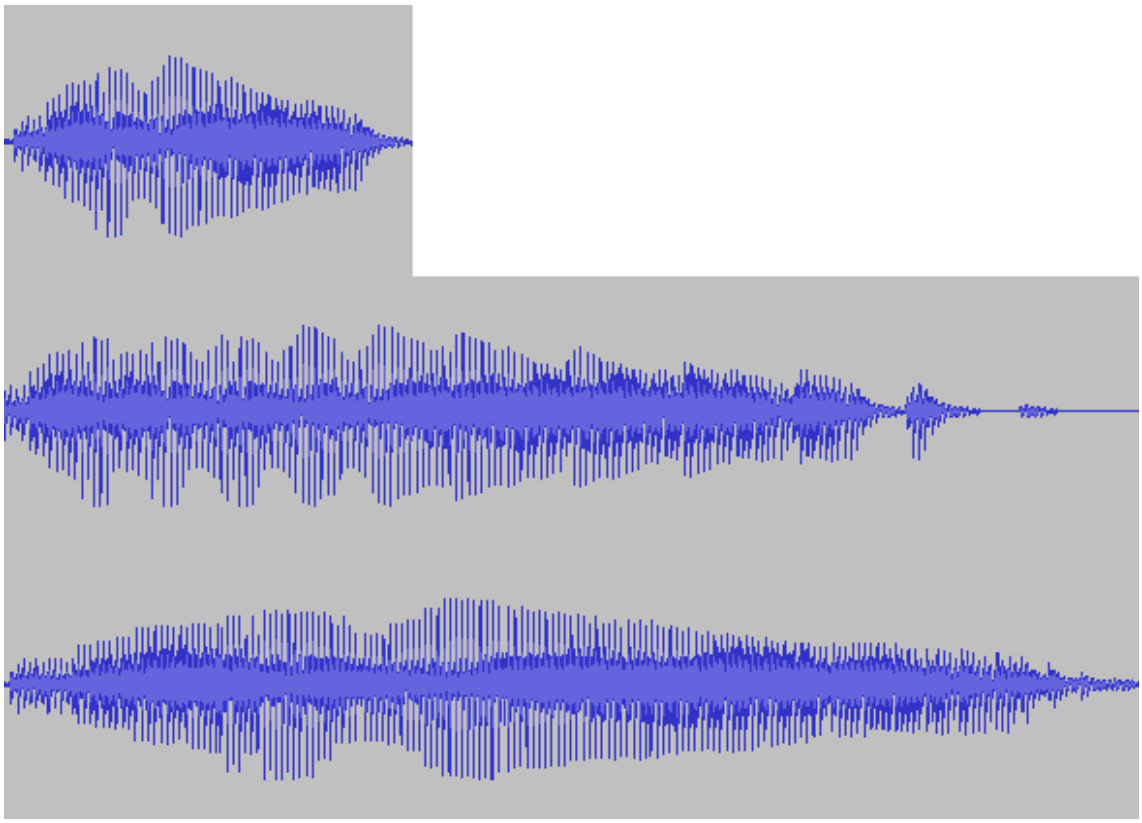
Porovnanie s existujúcimi riešeniami a iné riešenia

3.1 Porovnanie výsledkov s voľne dostupným softwarom Audacity

V tejto kapitole naše riešenie porovnáme s výsledkami, ktoré sme dosiahli v predloženej diplomovej práci. Aby sme sa mali od čoho odraziť, na začiatku sme sledovali ako sa správa pri úpravách reči software Audacity. Audacity je voľne dostupný software, ktorý slúži na úpravu digitálneho signálu. Bohužial, nie sú dostupné zdrojové súbory ani kompletná dokumentácia k tomuto softwaru. Reverzným inžinierstvom sme teda sledovali ako program spracováva signál pri riešení nášho problému. Skúsili sme do programu pustiť testovacie nahrávky aby sme zistili aké problémy tento software má. Výstup bol značne poškodený už pri jednoduchých nahrávkach, ktoré neobsahovali šum. Najväčším problémom riešenia Audacity bolo, že rečníkov hlas významným spôsobom kolísal.

Na obrázku 3.1 môžeme vidieť porovnanie výstupov z testovacieho vstupného signálu. Vstupný signál môžeme vidieť na hornom obrázku. Vstupný signál bol minimálne zašumený a mikrofón, ktorým sme reč nahrávali v podstate nenadsadzoval ani nepodsadzoval signál. Na prostrednom obrázku je vidieť graf výstupného signálu z Audacity a na spodnom grafe náš výstup.

Na začiatku sme si neboli istí prečo v riešení audacity hlas rečníka tak kolíše. Prišli sme na to až pri našom čiastočnom riešení. Všimnime si asympdoty jednotlivých výstupov. V audacity asympdota kolíše a vo výstupe vôbec nepripomína vstupný signál. Naša asympdota



Obr. 3.1: Pôvodný testovací signál(horný graf) porovnaný s výstupom z audacity(prostredný graf) a našim výstupom(spodný graf)

ale signál pripomína oveľa viac. Tvorcovia Audacity zrejme zobrali dlhšie vektory a tým sa reč skreslila.

Všimnime si tiež pravú časť prostredného grafu. Vo výstupe audacity môžeme vidieť niečo ako ozvenu po konci slova kde už by malo byť ticho. Tento fenomén bol zrejme tiež spôsobený nevhodnou voľbou časových úsekov pri duplikácii reči. Ako môžete vidieť na spodnom obrázku, u nás tento problém vôbec nenastal.

Všimnime si teraz amplitúdy prostredných častí všetkých troch signálov. V pôvodnom slove aj v našom signále sa tieto amplitúdy tiež podobajú. Je to z dôvodu správneho využitia nami vylepšeného PSOLA algoritmu. Keď by sme používali lineárne zosilňovanie a zoslabovanie vplyvu ľavej a pravej strany, na začiatku nášho výstupu by amplitúda prostrednej časti signálu v našom výstupe narástla podobne ako pri prostrednom grafe. Využitím tohto vylepšenia, sme získali nástroj na pozvoľnejší rast a pokles tejto amplitúdy.

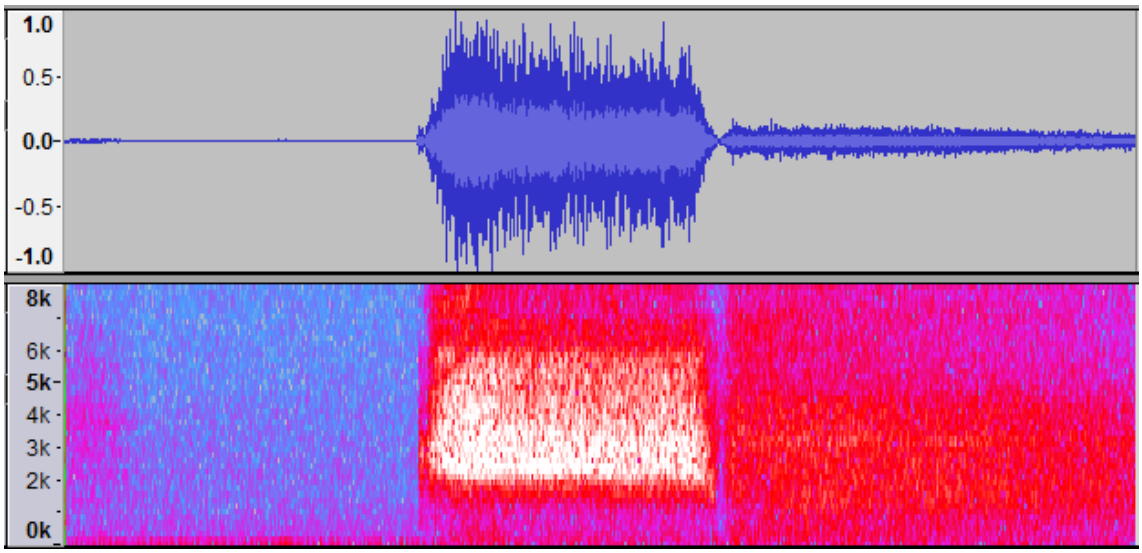
Naše riešenie teda v konečnom dôsledku prinieslo celkovo lepší dojem z výsledneho hlasu rečníka ako malo Audacity. Hlas rečníka sa v našom riešení nijako významne nekolíše a kvalita výstupu je relatívne dobrá. Obrázok 3.1 tieto tvrdenia názorne potvrdzuje. Signál

ľudského hlasu sa totiž na rečníka bude najviac podobať práve vtedy, keď sa budú podobať amplitúdy hlasivkových tónov pôvodného vstupu a syntetizovaného výstupu.

3.2 Iné riešenie detekcie frikatív

Po tom, čo sme už mali našu detekciu frikatív vylepšenú a funkčnú, sme narazili na druhé možné riešenie tejto detekcie [11]. Autori tohto riešenia sa na frikatíva pozerali inak ako my. Pred samotnou detekciou, zistili aké frekvencie majú jednotlivé frikatíva. Pri frikatívach, ktoré sú bez hlasivkových tónov, sa tieto frekvencie pri rôznych rozprávačoch moc nelíšia. Rozdiely sú väčšie iba pri rozprávačoch s rečovou vadou. Problémovejšie sú ale frikatíva s hlasivkovými tónmi, ich frekvencie sa odlišujú podľa farby ľudského hlasu. Ľudia s vyšším hlasom mali tieto frekvencie oveľa vyššie.

Riešenie tohto problému spočívalo v zistení frekvenčných hraníc jednotlivých frikatív a následným použitím spektrogramu na kritické úseky. Spektrogram môžeme vidieť na obrázku 3.2.



Obr. 3.2: Nahrávka frikatív „š" a „f" a ich spektrogram, na ktorom môžeme vidieť frekvencie jednotlivých frikatív

Na tomto obrázku môžeme vidieť porovnanie frekvencií dvoch frikatív pomocou spektrogramu. Červené až žlté farby spektra znázorňujú zvýšený výskyt daných frekvencií na úseku signálu. V testovacej vzorke je na začiatku signálu ticho. Ako môžeme vidieť, ticho

nemá významné zastúpenie v žiadnej frekvencii. Nasleduje frikatívum „š". Ako môžeme vidieť, toto frikatívum má najvyššie zastúpenie vo frekvenciách medzi 2kHz až 6kHz. Za týmto frikatívom nasleduje hláska „f". Toto frikatívum má vyššie zastúpenie na intervale $< 1kHz, 4kHz >$. Každé iné frikatívum má určitý interval, v ktorom sa jeho frekvencie vyskytujú.

Pomocou použitia spektrogramov by sa teda dalo zistiť o aké frikatívum sa na daných úsekoch jedná. Toto riešenie by teda mohlo nahradiť našu fázu analýzy signálu. Problémom s týmto riešení by sme sa zrejme aj tak nevyhli. Ako sme už spomínali, frekvencie frikatív s hlasivkovým tónom sa líšia u rôznych rozprávačov. Táto odchýlka zvyčajne nie je výrazná ale nepravidelnosť intervalov by mohla spôsobiť zlé rozpoznanie frikatíva.

3.3 Možné vylepšenia

Počas našej práce sme narazili na množstvo problémov, ktoré by sa v budúcnosti zrejme dali vyriešiť. Jedným z nich bol šum, ktorý sme napokon schovali do hlasu rečníka, aby nebol veľmi zreteľný. Tento šum by sa pravdepodobne dal odstrániť pomocou filtra, ktorý by bol zostrojený konkrétne pre naše nahrávky. Toto riešenie by bolo vhodnejšie ako použitie Chebyshevovho filtra, nakoľko tento potláča aj niektoré hlasivkové tóny a tým pádom mierne stlmí aj hlas rozprávača.

Ďalší problém, s ktorým sme prišli do kontaktu boli explozíva. Tieto hlásky sa v podstate nedajú jednoducho detekovať ani syntetizovať. Pomocou vylepšenej detekcie by sa ale predsa len niečo s takýmito hláskami dalo spraviť. A to by bolo predĺženie ticha pred nimi. Toto ticho predchádza každému explozívu a je spôsobené tým ako explozíva vznikajú. Pri väčšine musíme na chvíľku zadržať dych aby sme ho potom naraz prudko vytlačili. Moment zadržania dychu je v signále vidieť na niekoľkých bitoch a práve tieto by sa mali dať duplikovať bez problémov. Vylepšená detekcia explozív by tiež pomohla pri probléme, ktorý v našom výsledku ostal. Tento problém nastáva keď nasledujú dve frikatíva po sebe. Z našej analýzy totiž takýto úsek vyjde s výsledkom, že sa jedná o hlasivkové tóny. Táto chybná detekcia spôsobí, že explozíva sa zduplikujú. Napríklad pri slove „tatko" potom syntéza bude znieť ako keby rozprávač prečítal slova „tátktkó". Teda vnútri slova vznikne akési tiknutie, čo neznie veľmi dobre. Vylepšenou analýzou by sa táto chyba dala odstrániť.

Kapitola 4

Záver

V našej práci sme sa venovali analýze a syntéze digitálneho signálu zaznamenanej reči. Na vstupe do nášho programu sme mali krátke príbehy, ktoré predčítavali deti. Tento vstupný signál sme najprv pomocou rôznych algoritmov rozdelili na čo najkratšie úseky. Najprv sme v signále hľadali hlasivkové tóny. Súto pravidelné úseky signálu, ktoré vytvárajú ľudské hlasivky svojím sťahovaním a roztahovaním.

Na hľadanie hlasivkových tónov sme využili autokoreláciu. Táto analytická funkcia porovnáva signál sám so sebou pričom jedna porovnávaná časť sa posúva v čase. Viaceré analýzy ľudského hlasu uvádzajú, že hlasivkové tóny sa opakujú zhruba každých 25ms [3]. Preto sme signál najprv rozdelili na úseky dĺžky 25ms a tieto sme postupne posielali do autokorelácie. Vo výstupoch z tejto štatistickej funkcie sme potom hľadali lokálne maximá, ktoré určovali výskyt hlasivkových tónov. Samotná autokorelácia ale nestačila, museli sme pridať niekoľko vylepšení aby sme dosiahli výsledky, s ktorými sme mohli ďalej pracovať.

	Počet nájdených hlasivkových tónov (signál obsahoval 56 tónov)	**Úspešnosť**
Autokorelácia bez vylepšení	114	49.12%
Zákaz duplikátov + pohyblivý priemer funkčných hodnôt	70	80.00%
Zákaz duplikátov + pp funkčných hodnôt a frekvencie tónov	57	98.25%

Tabuľka 4.1: Štatistika počtov nájdených hlasivkových tónov pomocou rôznych vylepšení

V štatistike 4.1 môžeme vidieť výsledky, ktoré sme dosiahli rôznymi vylepšeniami výsledkov z autokorelačnej funkcie. V testovanej vzorke bolo 56 hlasivkových tónov, samotná autokorelačná funkcia s hľadaním lokálnych maxím našla až 114 hlasivkových tónov, takže sme potrebovali pridať rôzne sitá aby sme počty nájdených hlasivkových tónov znížili. Najprv sme zakázali duplikáty, lebo autokorelácia niektoré hlasivkové tóny označila viac krát. Ako môžeme vidieť v tabuľke, našimi vylepšeniami sme sa zlepšili z presnosti okolo 49% na presnosť **98,25%**, čo bol neočakávane dobrý výsledok.

Keď sme mali označené hlasivkové tóny v signále, skontrolovali sme všetky úseky signálu medzi troma hlasivkovými tónmi. Väčšina vektorov mala lokálne maximá na krajoch a jedno zhruba v strede. Odchýlka od stredu v týchto vektoroch bola zhruba 5%. Ale našli sa aj úseky, ktoré mali odchýlku od stredu väčšiu. Tieto sme podrobili ďalšej analýze.

V tejto fáze sme zisťovali, či daný úsek signálu obsahuje frikatívum. V práci sme sa sústredili najmä na predlžovanie signálu frikatív a hlások s hlasivkovými tónmi. Hlavne z toho dôvodu, že ďalšie zložky nahrávok boli explozíva, ktoré sa nedajú vôbec predlžovať a ticho respektíve šum, ktorého predlžovanie by mohlo viezť k rôznym mutáciám. Druhá fáza analýzy signálu používala zloženie troch štatistických funkcií. A to zero-crossing rate, štatistická sigma a energia signálu. Pomocou týchto troch funkcií sme nielen úspešne rozpoznali frikatívum od šumu ticha a explozív, ale rôznymi vylepšeniami využitia týchto funkcií sme dokázali rozpoznať dve alebo viac frikatív za sebou.

Po dvoch fázach analýzy nasledovala v našej práci syntéza reči so zachovaním identity rozprávača. Na analýzu signálu sme použili algoritmus PSOLA, ktorý sme ale tiež vylepšili, aby sme získali očakávané výsledky. Najväčšou úpravou pitch synchronous overlap and add bol nelineárny rast a pokles vplyvu vstupov na syntetizovanú časť. Nelineárnosť rastu a poklesu sme zabezpečili navrhnutím vlastného generátora rastových funkcií. Na základe rozdielu prostredného lokálneho maxima na vektore a skutočného stredu vektora, generátor vrátil rastovú a poklesovú funkciu. Týmto sme zabezpečili zreteľne lepšie prechody medzi samotnými hláskami. Práve v prechodoch sa vyskytovalo najviac úsekov, ktoré mali vychýlené prostredné maximum.

Počas samotnej syntézy pomocou PSOLA algoritmu sme syntetizovali aj frikatíva, tieto sa syntetizujú ľahšie v podstate bolo treba vektor frikatív skrátiť na správnu veľkosť a následne len zduplikovať celý vektor bez akýchkoľvek úprav.

Po syntéze jednotlivých častí ich jednoducho zložíme do výsledného signálu. Pri syntéze ešte kvôli šumu vznikali chyby. Preto sme sa rozhodli v poslednej fáze rozhodli signál prefil-

trovať pomocou Chebyshevovho filtra. Tento filter sa zaraďuje medzi low-pass filtre a teda potláča frekvencie, ktoré prekračujú hranicu. Túto hranicu dostane filter na vstupe, rovnako aj počet decibelov o koľko chceme frekvencie potláčať. V našom riešení sme frekvencie nad 1000Hz potláčali o 6dB. Nechceli sme aby sa na daných miestach zvuk stratil úplne, skôr sme chceli aby sa nežiadúce frekvencie stratili v reči rozprávača. Toto sa nám vďaka filtrom podarilo docieliť.

Po otestovaní programu na viacerých vstupoch, sme zhodnotili, že práca bola úspešná. Pri viacerých príbehoch tak isto aj našich testovacích vstupoch sme pri predlžovaní signálu vždy úspešne zachovali identitu rozprávača. A to aj pri veľmi veľkom predlžovaní, kedy sa jednotlivé úseky niekoľko krát syntetizovali. Aj keď pri väčšom predĺžení už ľudský hlas znel komicky, nakoľko rozprávačovi vyslovenie jednotlivých hlások, či slabík trvalo značne dlho.

Kapitola 5

Dodatok

V priloženom DVD prikladáme funkčný program napísaný v jazyku Octave. V programe necháme zakomentované príkazy na vykreslenie zaujímavých grafov. Po ich odkomentovaní si čitateľ bude môcť pozrieť napríklad výstupy z autokorelácie, syntetizované hlasivkové tóny, či porovnanie pôvodného signálu s našim výstupom. Tiež prikladáme niekoľko pôvodných, testovaných nahrávok signálu. K týmto prikladáme aj výstupy z nášho programu spolu s výstupmi vytvorenými v software Audacity. Tieto prikladáme aby čitateľ mohol porovnať náš výstup s voľne dostupným riešením.

Literatúra

[1] Marek Nagy: *Multimediálna čítanka*. Online, Dátum: 26.4.2016, https://www.mmcitanka.sk/.

[2] Marek Nagy: *Spracovanie digitálneho signálu*. Online, Dátum: 26.4.2016, https://moodle.uniba.sk/moodle/inf11/enrol/index.php?id=402.

[3] Martin Šukola: *Diplomová práca: Počítačový syntetizér spevu*. Univerzita Komenského, Bratislava, Slovensko, 2010.

[4] Fracois Xavier Nsabimana, Udo Zölzer: *Audio Signal Decomposition for Pitch and Time Scaling*. Communications, Control and Signal Processing, 3rd International Symposium on (pp. 1285-1290), 2008.

[5] Pavol Adam: *Diplomová práca: Úvod do metód spracovania zvuku v súčasnom multimediálnom prostredí*. Univerzita Komenského, Bratislava, Slovensko, 2006.

[6] Sami Lemetty and Matti Karjalainen: *Diplomová práca: Review of Speech Synthesis Technology*. Helsinki University of Technology, 1999.

[7] Jan Jagla, Julien Maillard, and Nadine Martin: *Sample-based engine noise synthesis using an enhanced pitch-synchronous overlap-and-add method*. The Journal of the Acoustical Society of America 132.5: 3098-3108, 2012.

[8] Joshua Patton: *ELEC 484 Project—Pitch Synchronous Overlap-Add*. University of Victoria, BC, Canada, 2012

[9] Tabet Youcef, Mohamed Boughazi: *Speech synthesis techniques*. University of M'hamed Bouguerra Boumerdes, Algeria, 2012

[10] Jozef Štefánik, Milan Rusko, Dušan Považanec: *The frequency of Words, Graphemes, Phones and Other Elements in Slovak* Bratislava, Jazykovedný časopis, 50 No. 2, pp. 81 - 93, 1999

[11] Peter Ladefoged, sandra Ferrari Disner: *Vowels and consonants* Wiley-Blackwell, pp. 55, pp. 68-81, 2005