TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky a mezioborových inženýrských studií

Studijní program: M2612 – Elektrotechnika a informatika
Studijní obor: 3902T005 – Automatické řízení a inženýrská informatika

Detekce změny řečníka v telefonních záznamech

Speaker change detection in phone records

Diplomová práce

Autor: **Štefan Zverec**
Vedoucí práce: Ing. Jindřich Žďánský, Ph.D.
Konzultant:

V Liberci 14. 5. 2007

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Katedra elektroniky a zpracování signálů Akademický rok: 2006/2007

ZADÁNÍ DIPLOMOVÉ PRÁCE

Jméno a příjmení: **Štefan Zverec**

studijní program: M 2612 – Elektrotechnika a informatika

obor: 3902T005 – Automatické řízení a inženýrská informatika

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

Název tématu:

Detekce změny řečníka v telefonních záznamech

Zásady pro vypracování:

1. Seznamte se s principy detekce bodu změny ve stochastickém procesu.
2. Prostudujte zakladní algoritmy detekce změny řečníka.
3. Vytvořte databázi telefonních záznamů.
4. Navrhněte metodu vhodnou k detekci změn mluvčích v telefonním signálu.
5. Proveďte vyhodnocení navržené metody.

Prohlášení

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

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

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

14.5.2007

.....
Štefan Zverec

Poděkování

Rád bych touto formou poděkoval Ing. Jindřichu Žďánskému, Ph.D. za odborné vedení, poskytnuté cenné rady a zkušenosti, dále všem mým přátelům, kteří mi pomohli s vytvořením databáze telefonních hovorů, především však svým rodičům, bez jejichž podpory při studiu by tato práce nikdy nevznikla.

Anotace

Diplomová práce se zabývá možností využití metody binárního dělení pro potřeby detekce změny řečníka v telefonních hovorech. V úvodních kapitolách nalezneme dva základní přístupy k parametrizaci akustického signálu. Pomocí diskrétní Fourierovy transformace – tzv. melfrekvenční kepstrální příznaky a pomocí lineární prediktivní analýzy – LPC příznaky. Dále je zde stručně popsán princip převedení problému změny mluvčího na změnu parametrů ve stochastickém procesu a popsány algoritmy trénování, testování a vyhodnocení detektoru změn.

Práce popisuje jednotlivé kroky při trénování a testování detekce změn mluvčích a uvádí výsledky vyhodnocení úspěšnosti metody na základě počtu správně nalezených změn. Pro pořízenou databázi telefonních hovorů byla statistickým vyhodnocením stanovena míra úspěšnosti detekce F=72% pro reálné nahrávky, F=96% pro nahrávky uměle vytvořené.

Annotation

This thesis is interested in utilization of binary segmentation method for detection of speaker change in phone dialogue. In opening chapters two basic accesses to acoustic signal parameterization are described. Using discrete Fourier Transformation - so-called melfrequency kepstral parameters and using linear predictive analyse - LPC parameters. Next, there is described transfer of speaker change problem to parameter's change in stochastic process and algorithms of training, testing and change detection evaluation are described.

Thesis describes single steps in speaker change detection training and testing, and introduces results of method success evaluation based on number of correctly founded changes. For created database of phone dialogues was determined rate of detection success by statistic evaluation F=72% in real records and F=96% in prepared records.

Prohlášení ... 3
Poděkování ... 4
Anotace ... 5
Seznam zkratek ... 7
Úvod ... 8
1 Metody detekce změny mluvčího ... 9
1.1 Digitalizace akustického signálu, segmentace, parametrizace, detekce změny ... 9
1.1.1 Vznik akustického signálu formou řeči ... 9
1.1.2 Digitalizace ... 10
1.1.3 Segmentace ... 11
1.1.4 Parametrizace ... 11
1.1.5 Detekce změny ... 12
1.2 Parametrizace signálu ... 13
1.2.1 Melfrekvenční kepstrální koeficienty (MFCC) ... 13
1.2.2 Kepstrální koeficienty LPC ... 14
1.3 Detekce změny ve stochastickém procesu ... 16
1.3.1 Testování hypotéz ... 16
1.4 Detekce jednoho bodu změny ... 17
1.4.1 Metoda maximální věrohodnosti (ML) ... 17
1.5 Detekce více bodů změny ... 18
1.5.1 Metoda binárního dělení ... 18
1.5.2 Efektivní řešení metody binárního dělení ... 26
1.6 Metoda vyhodnocení úspěšnosti detekce změny mluvčího ... 27
2 Implementace metody binárního dělení ... 30
2.1 Pořízení trénovacích a testovacích dat ... 30
2.1.1 Záznam telefonního hovoru, Call Recorder CR 3000 ... 30
2.1.2 Třídění nahrávek, převzorkování ... 32
2.1.3 HTK, HCopy ... 32
2.1.4 Referenční data ... 33
2.2 Implementace metody binárního dělení ... 34
2.2.1 Popis algoritmu, vývojový diagram ... 34
2.2.2 Trénování metody, odhad kritické hranice ... 35
2.3 Zpracování výsledků, algoritmus určení počtu H,I,D ... 36
3 Vyhodnocení metody binárního dělení ... 37
3.1 Trénování metody ... 37
3.1.1 Příznaky pro potřeby trénování ... 37
3.1.2 Trénovací a testovací databáze THSZ07 ... 38
3.1.3 Výsledky trénování ... 39
3.2 Testování metody ... 43
3.2.1 Výsledky testování ... 43
3.2.2 Porovnání úspěšnosti testování na databázích ART a COST 278 ... 44
Závěr ... 46
Literatura ... 48
Příloha A ... 49

Seznam zkratek

COST	CO-operation in Science and Technology
DFT	Discrete Fourier Transform
HMM	Hidden Markov Model
IDFT	Inverse Discrete Fourier Transform
LPC	Linear Prediction Coefficients
MFCC	Mel-Frequency Cepstral Coefficients
ML	Maximum Likelihood

Úvod

Žijeme v době, kdy dostupnost informací, tolik potřebných pro další rozvoj lidské společnosti, nabývá stále většího významu. S tím úzce souvisí i rozšíření a vývoj informačních a telekomunikačních technologií. Dnešní prostředky výpočetní techniky nám poskytují účinné nástroje pro jednoduché třídění a vyhledávání dat, díky kterým se lze ve velkém množství dostupných informací (především textového charakteru) snadno orientovat. Běžným a nejjednodušším způsobem, jak se dorozumívat, jak si sdělovat různé informace, je ale stále mluvené slovo. Jelikož by byl ruční přepis záznamů řeči (například různých jednání, politických debat, radiových či televizních zpráv, dopravních hlášení apod.) časově velice náročný, zabývá se mnoho výzkumných týmů pracujících v oboru rozpoznávání řeči myšlenkou jejich automatického přepisu a indexování, což by mělo usnadnit archivaci a následné vyhledávání těchto záznamů podle různých klíčů.

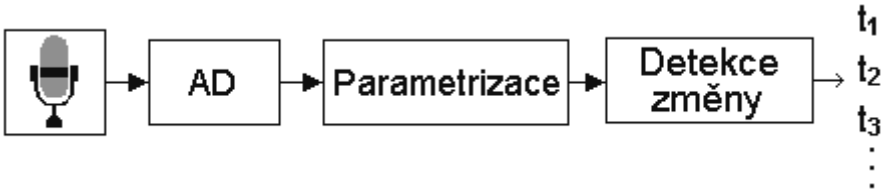
Takovéto systémy, které by byly schopny automatického přepisu zvukových nahrávek se nazývají media mining systémy. Podobný systém pro češtinu je vyvíjen i na Technické univerzitě v Liberci. Tyto systémy se většinou skládají z několika modulů, které mají na starosti například odstranění neřečových složek z nahrávky, jakou je třeba hudba, ruch z ulice, nebo delší pauzy. Jiné nahrávku segmentují podle jednotlivých mluvčí, což je nezbytné pro jejich identifikaci (jde-li o známé osobnosti, tak můžeme rozpoznávat i konkrétní osobu, v ostatních případech nám postačí rozpoznání pohlaví mluvčího), nakonec následuje samotné rozpoznávání textu řeči a indexování záznamů.

Cílem této diplomové práce je ověření robustnosti detekce změny řečníka metodou binárního dělení na reálných datech telefonních rozhovorů dvou mluvčích, vyznačujících se omezenou šířkou pásma, značným šumem, místy, kdy nemluví žádný z řečníků a místy, kdy naopak hovoří oba současně a porovnání její úspěšnosti testované na uměle vytvořených datech. Prvotním úkolem je tedy pořízení dostatečného počtu nahrávek, které poslouží jako trénovací a testovací množina dat, dále pak naprogramování a natrénování off-line metody detekce místa změny a nakonec vyhodnocení úspěšnosti metody při užití různých příznaků.

1 Metody detekce změny mluvčího

1.1 Digitalizace akustického signálu, segmentace, parametrizace, detekce změny

Rozpoznání změny mluvčího předchází několik procesů, které usnadňují matematický výpočet místa změny. Patří mezi ně mimo jiné: digitalizace akustického signálu, segmentace, převod každého segmentu do vhodného příznakového prostoru a až nakonec samotná detekce místa změny řečníka.

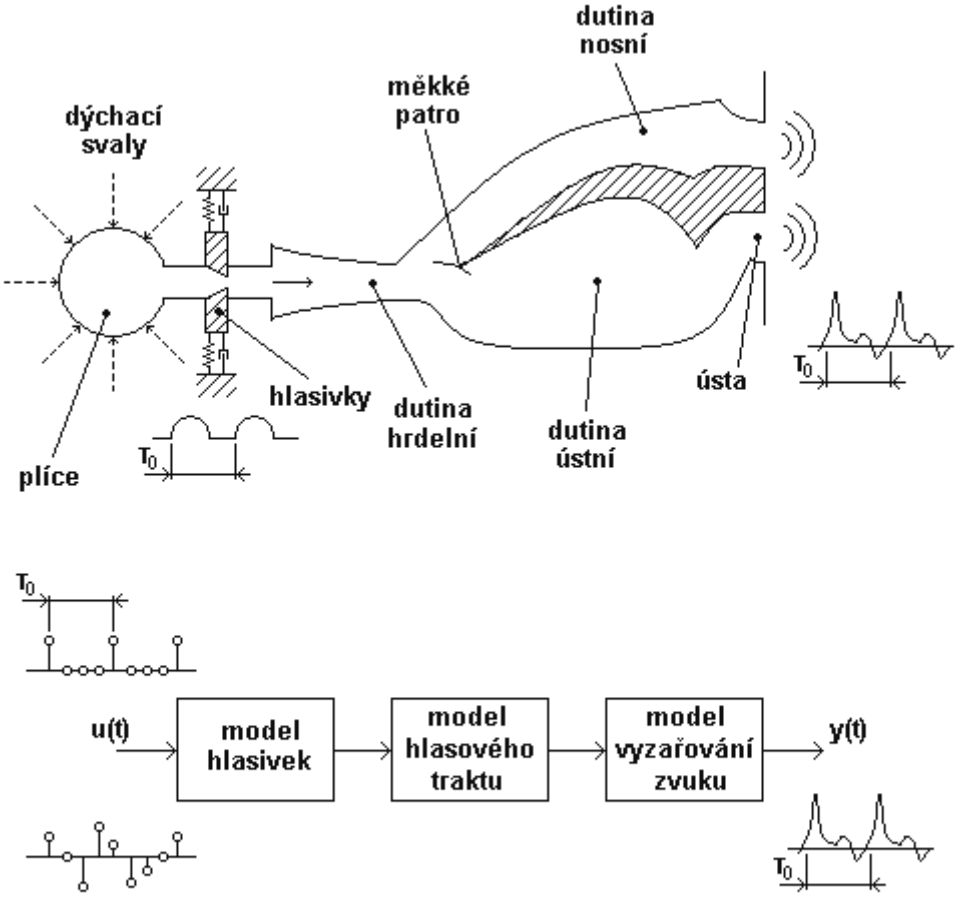


Obr. 1: Blokové schéma detekce změny řečníka

1.1.1 Vznik akustického signálu formou řeči

Mezi nejjednodušší a nejpřirozenější způsob komunikace mezi lidmi patří mluvená řeč. Zdrojem řečových kmitů (jedná se o podélné mechanické vlnění) je hlasové ústrojí, mezi něž patří především: plíce, hrtan, hlasivky, hrdlo, ústní dutina, patro, jazyk, rty a dokonce i zuby. Tyto všechny součásti hlasového ústrojí ovlivňují charakter námi vydávaného zvuku – řeči, mezi kterou řadíme zvuky znělé, neznělé a samozřejmě i šepot. To, že jsme schopni vydávat znělé zvuky, umožňují pružné hlasivky, které se střídavě svírají a roztahují a pod tlakem vzduchu vycházejícího z plic kmitají. Frekvence těchto kmitů je závislá nejen na tlaku vzduchu z plic a momentální pružnosti hlasivek, ale i na tom, zda je mluvčím muž, žena, či dítě. Zpravidla se udává [5] rozsah této frekvence mezi 150 až 400 Hz a ta charakterizuje základní tón lidského hlasu.

Řeč, jako akustický signál, můžeme popsat fyzikálními veličinami, například frekvencí. Frekvenční rozsah zvuku, který člověk dokáže vnímat, je 20 Hz až 20 kHz. Pro řeč je nejvýznamnější rozsah 2-4 kHz, pro který je i lidské ucho nejcitlivější. Dynamický rozsah řeči (rozdíl mezi šepotem a hlasitou mluvou) je cca 50 dB.



Obr. 2: Zjednodušené schéma produkce řeči, matematický model

1.1.2 Digitalizace

Pojmem digitalizace rozumíme vzorkování a následné kvantování zvukového signálu. To nám zajišťuje např. AD převodník běžné zvukové karty. Z vlastností řečového signálu je třeba vhodně zvolit dva základní parametry, kterými jsou vzorkovací frekvence a kvantizační krok [4]. Většina informace v lidské řeči je obsažena v oblasti frekvencí řádu stovek Hz pro znělé hlásky a maximálně několika kHz pro hlásky neznělé. Proto postačuje obvykle vzorkovací frekvence 16kHz i pro nejnáročnější účely, lze však vystačit i s 8kHz, zvláště jde-li o signál telefonní. Co se týče kvantizačního kroku, postačující je 12bitové rozlišení, v praxi však častěji užívané 16bitové. Pro účely rozpoznávání telefonního signálu vyhovuje i 8bitové kvantování.

1.1.3 Segmentace

Jelikož vokální trakt během řeči mění své parametry, hovoříme o řeči jako o nestacionárním signálu. Pokud bychom však zkoumali tento signál v kratších intervalech (běžně 10-30 ms), můžeme uvažovat tyto segmenty jako signál stacionární a tím výrazně snížit počet parametrů, vstupujících do samotného rozpoznávače.

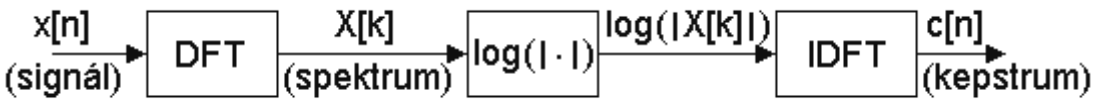
Běžně se používají dvě metody segmentace. Asynchronní segmentace, kdy je pevně daná velikost segmentu (framu) a tzv. pitch-synchronní segmentace, kdy se délka framu mění v závislosti na frekvenci [4]. Obecně lze říci, že pitch-synchronní metoda je lepší, protože poskytuje přesnější popis analyzovaného signálu. Nevýhodou této metody je však větší složitost algoritmu. Abychom zabránili skokovým změnám parametrů na okrajích segmentů při asynchronní segmentaci, používá se překrývání segmentů doplněné váhováním Hammingovým okénkem, kterým minimalizujeme efekt *prosakování*. K tomuto jevu dochází při nevhodné volbě délky okénka, nebo počtu bodů diskrétní Fourierovy transformace (DFT) a to tehdy, nejsou-li tyto délky shodné s celistvým násobkem počtu period harmonického signálu.

1.1.4 Parametrizace

Úlohou parametrizace je extrakce příznaků z akustického signálu. Mezi nejjednodušší patří například energie framu, její první a druhá diference, autokorelační funkce, apod. V oblasti zpracování a rozpoznání řeči se nejčastěji využívá příznaků získaných z kepstra signálu (tzv. kepstrálních příznaků).

Kepstrum $c[n]$ posloupnosti vzorků signálu $x[n]$ je definováno jako inverzní Fourierova transformace logaritmu absolutní hodnoty spektra signálu [4].

$$c[n] = DFT^{-1}\{\log(|DFT\{x[n]\}|)\} \tag{1}$$



Obr. 3: Výpočet kepstra signálu

Mezi hlavní výhody kepstrální analýzy patří možnost oddělit dvě složky signálu, který vznikl konvolucí těchto složek, např. složky signálu buzení hlasového traktu s impulsní odezvou hlasového ústrojí [5].

Před vlastním výpočtem je vhodné jednotlivé vzorky framu upravit. Nejprve aplikujeme tzv. preempfázi, což je zvýraznění vyšších frekvencí v signálu. To se realizuje pomocí jednoduchého číslicového filtru:

$$y(n) = x(n) - a \cdot x(n-1) \tag{2}$$

$x(n)$ ... *vzorky původního signálu*

$y(n)$ ... *vzorky signálu po preempfázi*

a ... *konstanta, jejíž hodnota se obvykle volí v rozsahu 0,95 – 0,98*

Následně se aplikuje Hammingovo okénko, které potlačuje hodnoty vzorků na okrajích jednotlivých framů. Vzorky násobíme příslušnými hodnotami váhové funkce $w(n)$:

$$w(n) = 0{,}54 + 0{,}46 \cdot \cos\left[\left(\frac{1}{2}N - n\right)\frac{2\pi}{N}\right] \tag{3}$$

n-tý vzorek po aplikaci Hammingova okénka tedy vypočteme:

$$y(n) = x(n) \cdot w(n) \tag{4}$$

Pro každý segment je vypočítán jeden vektor, který se skládá z několika různých příznaků. Výsledkem parametrizace je tedy posloupnost příznakových vektorů, na základě kterých je možné provádět trénování rozpoznávače, nebo i samotné rozpoznávání. Rozpoznávání řeči nejlépe vyhovují MFCC kepstrální příznaky, které budou podrobněji popsány v kapitole 1.2.1.

1.1.5 Detekce změny

Existuje mnoho metod, specifických pro danou oblast rozpoznávání. V případě změny mluvčího uvažujeme různé přístupy k rozpoznávání. Mezi jednodušší úlohy patří rozpoznávání jednoho bodu změny – tzv. *single change point analýza.* V praxi se však častěji setkáme s problémem rozpoznávání více bodů změny – *multiple change point analýza*. Budeme se zabývat výhradně off-line metodami rozpoznávání, což znamená, že již máme k dispozici nahrávky rozhovorů, které pak následně podrobujeme trénování a testování.

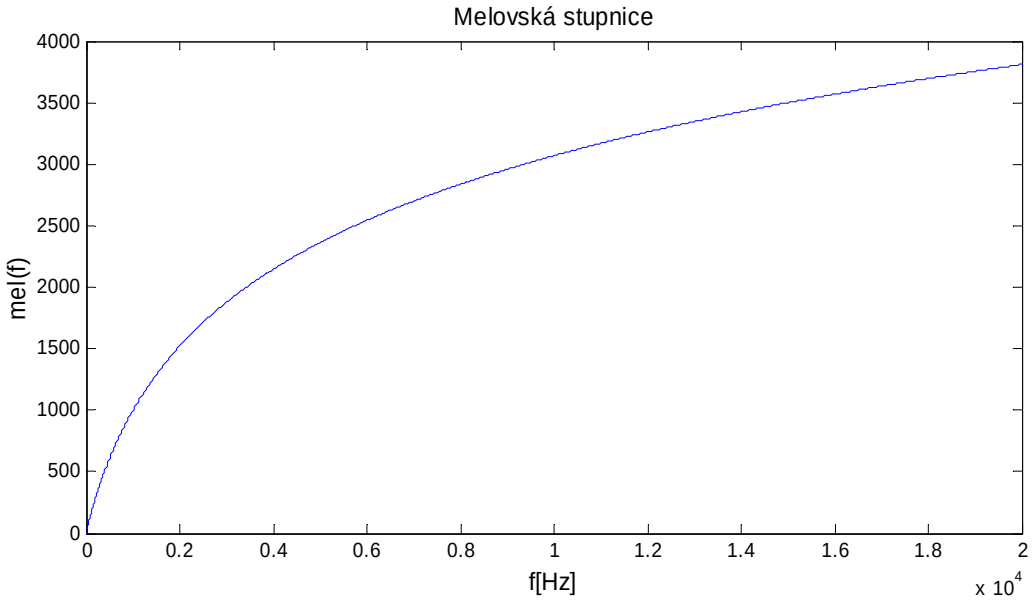
1.2 Parametrizace signálu

Jak již bylo řečeno, parametrizace má za úkol, na základě povahy akustického signálu, resp. jeho jednotlivých segmentů signálu, převést zvukovou informaci na tzv. příznakový vektor, který obsahuje informace významné pro danou metodu rozpoznávání V následujících bodech jsou popsány dva hlavní přístupy parametrizace pro potřeby oblasti rozpoznávání řeči, a to melfrekvenčních kepstrálních koeficientů (MFCC) a LPC kepstrálních koeficientů.

1.2.1 Melfrekvenční kepstrální koeficienty (MFCC)

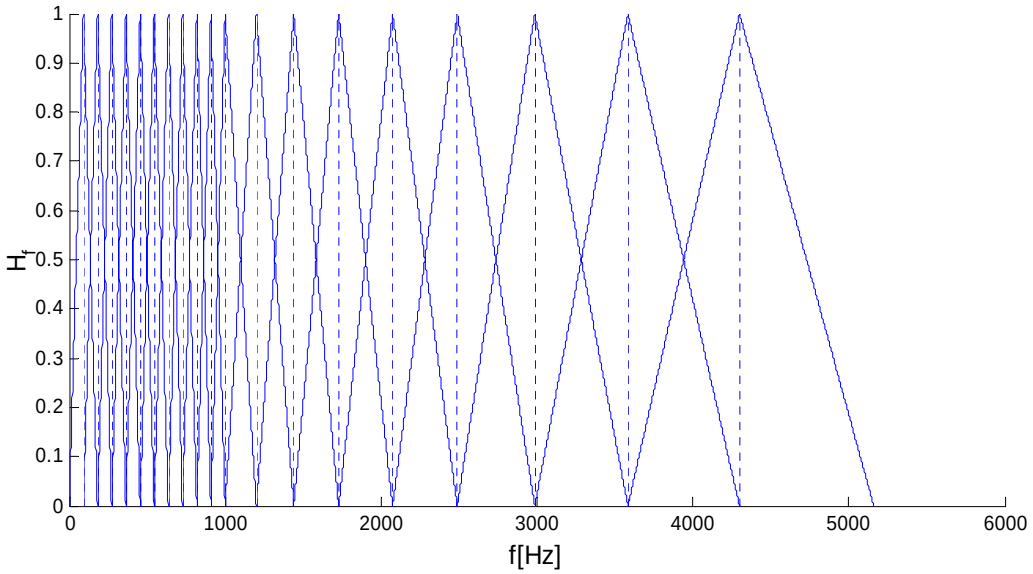
Výpočet těchto příznaků vychází z frekvenční oblasti akustického signálu a ze znalosti, že lidské ucho nevnímá akustický signál různých frekvencí v lineární stupnici, ale spíše v logaritmické (*melovské*) stupnici, dle vztahu:

$$mel(f) = 1125 \cdot \ln(1 + \frac{f}{100}) \tag{5}$$



Obr. 4: Průběh funkce pro přepočet reálných frekvencí na "melovské"

Nejprve je na každý segment akustického signálu aplikována diskrétní fourierova transformace, jejímž výsledkem je jeho amplitudové spektrum. Toto spektrum je filtrováno tzv. melovskou bankou N trojúhelníkových filtrů s logaritmicky rozmístěnými středy.



Obr. 5: Melovská banka trojúhelníkových filtrů

Pro každý frame vzorku takto určíme energii, dostáváme tedy vektor energie $E_1, E_2, ..., E_N$. Užitím logaritmu energie v každém pásmu a následné diskrétní kosínové transformace:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \tag{6}$$

získáváme vektor kepstrálních příznaků. Pro účely rozpoznávání řeči se využívá prvních 13 příznaků, pro rozpoznávání řečníka se první příznak vynechává.

1.2.2 Kepstrální koeficienty LPC

Dalším způsobem, jak dospět ke kepstrálním příznakům, je lineární prediktivní analýzou akustického signálu. Vychází ze znalosti lineárního modelu hlasového traktu a hlavní výhodou oproti předchozímu postupu získávání příznaků je její relativní výpočetní nenáročnost.

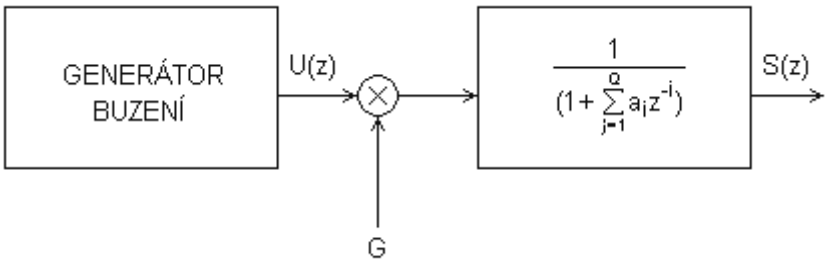
Přenosová funkce hlasového traktu lze zapsat ve tvaru:

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{A(z)} = \frac{G}{1+\sum_{i=1}^{Q} a_i z^{-i}}, \tag{7}$$

kde *S(z)* je obraz výsledného signálu, *U(z)* je obraz buzení, *G* je koeficient zesílení, *Q* je řád modelu a koeficienty a_i charakterizují lineární filtr.

Tento lineární systém lze též popsat pomocí kepstrálních koeficientů [6]. Pro jejich vyčíslení nejprve určíme logaritmus přenosové funkce *H(z)*, tj. $\log(H(z)) = \log(G / A(z))$. Jestliže polynom *A(z)* proměnné z^{-i} je *Q*-tého řádu a dále všechny kořeny tohoto polynomu leží uvnitř jednotkové kružnice a $A(\infty)=1$, lze provést Taylorův rozvoj *log(G/A(z))* v řadu

$$\log(G / A(z)) = c(0) + c(1)z^{-1} + ... = \sum_{k=0}^{\infty} c(k)z^{-k}, \tag{8}$$



Obr. 6: Model vytváření řeči s lineárním číslicovým filtrem

kde *c(k)* jsou tzv. kepstrální koeficienty LPC. Abychom se zbavili logaritmu na levé straně rovnice, budeme obě strany rovnice derivovat. Po úpravě dostaneme

$$-\sum_{i=1}^{Q} i a_i z^{-i} = \left[\sum_{k=1}^{\infty} k c(k) z^{-k}\right]\left[\sum_{i=0}^{Q} a_i z^{-i}\right]. \tag{9}$$

Nyní již můžeme, po roznásobení pravé strany rovnice a porovnáním členů u stejných mocnin *z* (za předpokladu, že $a_0=1$), odvodit vztahy pro výpočet kepstrálních koeficientů LPC

$$\begin{aligned} c(1) &= -a_1, \\ c(k) &= -a_k - \sum_{i=1}^{k-1}\left(\frac{i}{k}\right)c(i)a_{k-i} \quad \text{,pro } 2 \le k \le Q, \\ c(k) &= -\sum_{i=1}^{Q}\left(\frac{k-i}{k}\right)c(k-i)a_i \quad \text{,pro } k = Q+1, Q+2,... \end{aligned} \tag{10}$$

.

1.3 Detekce změny ve stochastickém procesu

Jedná se o oblast zpracování signálu ve statistické analýze. Na základě svého charakteru uvažujeme signál jako určitý stochastický proces a zkoumáním tohoto procesu, resp. jeho parametrů, můžeme určit bod změny procesu. Takto lze jednoduše převést problém detekce změny řečníka na detekci změny parametrů náhodného procesu.

1.3.1 Testování hypotéz

Ze znalostí teorie testování hypotéz můžeme posloupnost příznakových vektorů $x_1, x_2, ..., x_T$, popisujících zkoumaný úsek akustického záznamu, převést na popis nezávislých veličin $X_1, X_2, ..., X_T$ distribučními funkcemi $F_1, F_2, ..., F_T$. Následně pak úlohu detekce změny bodů převedeme na úlohu testování nulové hypotézy [1]:

$$H_0 : F_1 = F_2 = ... = F_T \tag{11}$$

oproti alternativní hypotéze

$$H_1 : F_1 = ... = F_{t_1} \neq F_{t_1+1} = ... = F_{t_2} \neq F_{t_2+1} = ... = F_{t_s} \neq F_{t_s+1} = ... = F_T \tag{12}$$

kde $1 < t_1 < t_2 < ... < t_s < T$, kde s je neznámý počet bodů změn a $t_1, t_2, ..., t_s$ jsou pozice bodů změn, které hledáme. Jestliže distribuční funkce $F_1, F_2, ..., F_T$ náleží do společné parametrické třídy $F(\theta)$, kde $\theta \in R^P$, pak lze problém detekce bodů změny považovat za problém testování hypotéz jejich parametrů $\theta_i, i = 1, ..., n$

nulová hypotéza:

$$H_0 : \theta_1 = \theta_2 = ... = \theta_T = \theta \tag{13}$$

oproti alternativní hypotéze:

$$H_1 : \theta_1 = ... = \theta_{t_1} \neq \theta_{t_1+1} = ... = \theta_{t_2} \neq \theta_{t_2+1} = ... = \theta_{t_s} \neq \theta_{t_s+1} = ... = \theta_T \tag{14}$$

kde s a $t_1, t_2, ..., t_s$ odhadujeme.

1.4 Detekce jednoho bodu změny

Jako nejjednodušší úlohu detekce změny řečníka budeme uvažovat změnu pouze v jednom bodě. Jedná se tedy o případ, kdy na základě vypočítaných parametrů určíme místo s největší pravděpodobností změny řečníka. V souladu s předchozími postupy můžeme problém detekce změny řečníka chápat jako obecnou změnu parametrů gaussovského procesu a tedy jako testování hypotéz:

$$H_0 : \mu_1 = \mu_2 = ... = \mu_T \qquad \Sigma_1 = \Sigma_2 = ... = \Sigma_T \tag{15}$$

oproti hypotéze

$$H_1 : \mu_1 = ... = \mu_t \neq \mu_{t+1} = ... = \mu_T \qquad \Sigma_1 = ... = \Sigma_t \neq \Sigma_{t+1} = ... = \Sigma_T \tag{16}$$

, kde $d < t < T - d$ a d je rozměr náhodného vektoru.

1.4.1 Metoda maximální věrohodnosti (ML)

Na základě testování hypotéz popsaném v předchozích kapitolách je možné postupnými úpravami dospět ke vztahu testování změn parametrů gaussovského procesu [1], který nám charakterizuje pravděpodobnost změny pro každý časový okamžik nahrávky.

$$y = \alpha \sqrt{\max_{d<t<T-d} \left(T \log\left|\hat{\Sigma}\right| - t \log\left|\hat{\Sigma}_1\right| - (T-t)\log\left|\hat{\Sigma}_T\right| \right)} - \beta \tag{17}$$

,kde

$\alpha = a(\log T) = (2 \log\log T)^{\frac{1}{2}}$, $\beta = b_{2d}(\log T) = 2\log\log T + d\log\log\log T - \log\Gamma(d)$ a hodnoty d a T charakterizují rozměr příznakového vektoru.

Kovarianční matice $\hat{\Sigma}, \hat{\Sigma}_1, \hat{\Sigma}_T$ jsou dány vztahy:

$$\hat{\Sigma} = \frac{1}{T} \sum_{i=1}^{T} (x_i - \hat{\mu})(x_i - \hat{\mu})'$$

$$\hat{\mu} = \frac{1}{T} \sum_{i=1}^{T} x_i$$

$$\hat{\Sigma}_1 = \frac{1}{t} \sum_{i=1}^{t} (x_i - \hat{\mu}_1)(x_i - \hat{\mu}_1)'$$

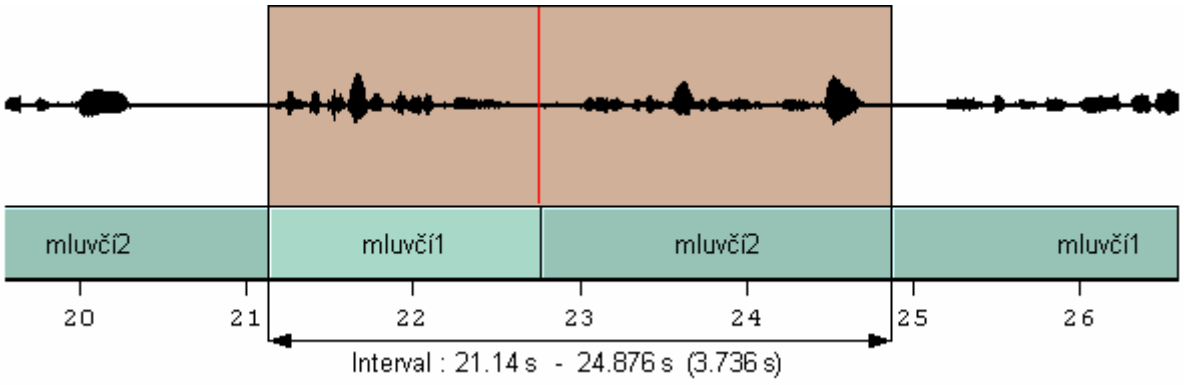
$$\hat{\mu}_1 = \frac{1}{t} \sum_{i=1}^{t} x_i$$

$$\hat{\Sigma}_T = \frac{1}{T-t}\sum_{i=t+1}^{T}(x_i - \hat{\mu}_T)(x_i - \hat{\mu}_T)'$$

$$\hat{\mu}_T = \frac{1}{T-t}\sum_{i=t+1}^{T}x_i$$

1.5 Detekce více bodů změny

Chceme-li detekovat více, než jednu změnu mluvčího v záznamu, je výhodné využít principů detekce jednoho bodu změny, která je vlastně speciálním případem změny vícebodové. Jednoduše aplikujeme předchozí postup (viz. kapitola 1.4) pouze na vybranou část nahrávky. Snažíme se tuto část vybírat tak, aby výsledný ML odhad parametrů byl co nejvěrohodnější – tedy aby v úseku byla právě jedna změna a oba mluvčí byli v tomto úseku zastoupeni po co nejdelší dobu. Jednou z metod, která nám určuje, jak tyto úseky volit, je *metoda binárního dělení* [8].



Obr. 7: Ideální volba úseku vstupujícího do ML odhadu parametrů

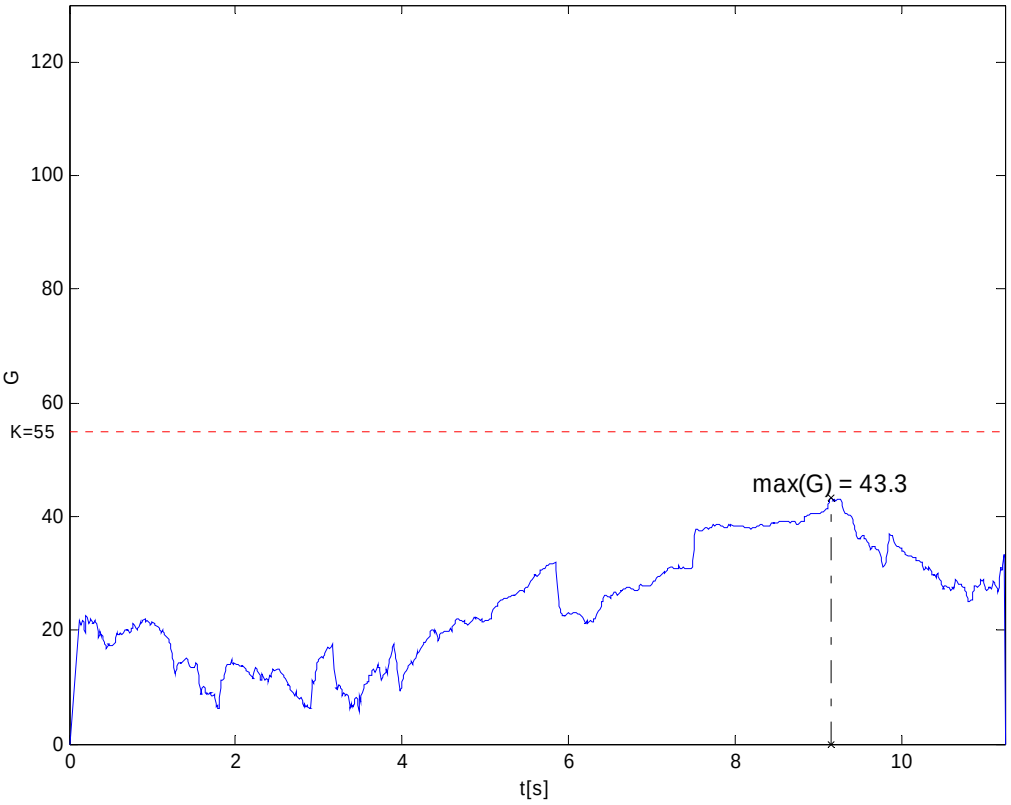
1.5.1 Metoda binárního dělení

Mezi výhody této metody patří především její výpočetní nenáročnost a snadná trénovatelnost, protože jediné, co je třeba určit, je kritická hranice (označme ji K_{opt}) zamítnutí nulové hypotézy H_0. Jedná se o postupné dělení časové osy podle výsledků testování hypotéz. Kritická hranice určuje, jak velká změna parametrů ještě může být označena za změnu řečníka, tedy charakterizuje počet těchto změn.

Na počátku je uvažován první a poslední segment nahrávky jako první body změny. V dalším kroku na základě testování hypotéz H_0 versus H_1 je stanoveno nejpravděpodobnější místo změny a pokud bod změny není nalezen (H_0 přijata),

algoritmus vyhledávání končí, pokud nalezen je (H_0 zamítnuta), je tento nový bod označen, čímž je nahrávka rozdělena na dvě části, které jsou následně podrobeny předchozímu testu. Takto se pokračuje s dělením až do doby, kdy již nejsou v žádném další kroku žádné nové změny nalezeny.

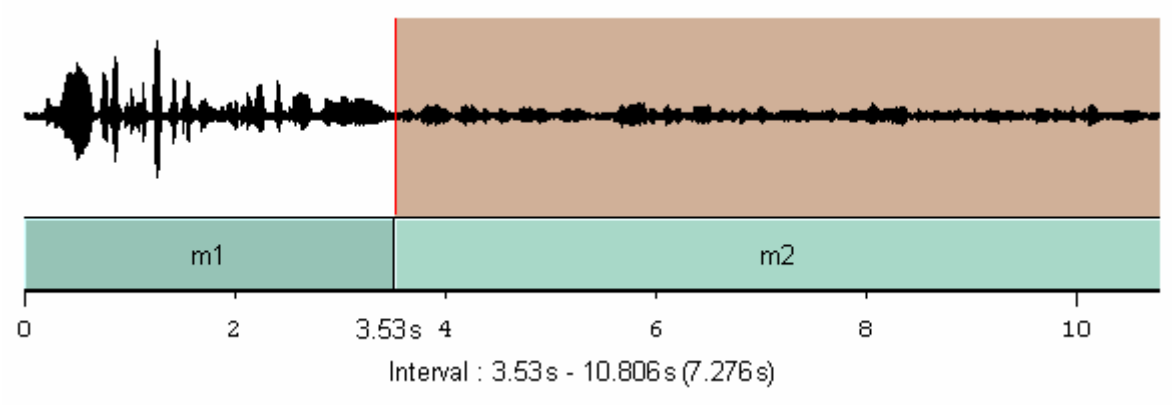
Na obrázku (Obr.8) je zobrazen průběh zisku cesty *G* prvního kroku, který byl vypočten metodou binárního dělení krátké nahrávky, na které hovořil jediný řečník, tedy ke změně nedošlo. Jak je z grafu patrné, ani v jednom bodě funkce zisku cesty nedosáhla zvolené kritické hranice *K=55*. Její maximum - nejpravděpodobnější místo změny má hodnotu *G=43,3* a tedy v souladu s nahrávkou není toto místo označeno jako nový bod změny.



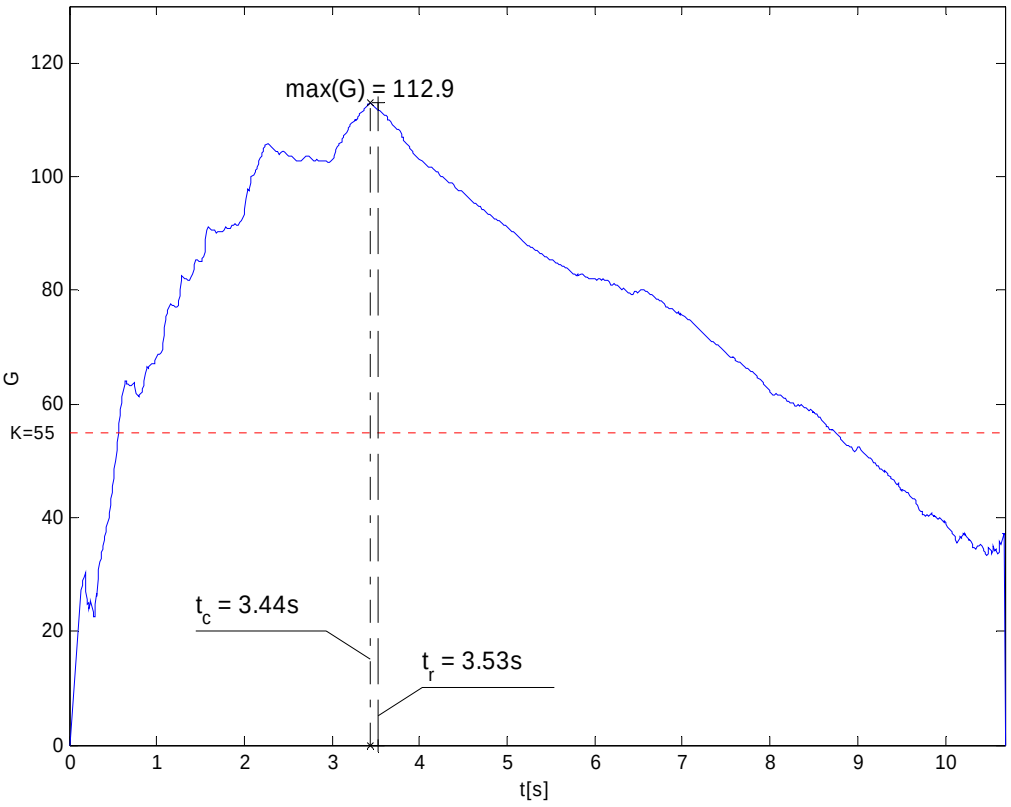
Obr. 8: Průběh zisku cesty G žádné změny

Tento případ je zde uveden pouze jako ilustrační, protože v běžném telefonním rozhovoru se s tímto případem, kdy hovoří jediný mluvčí, nesetkáme.

Dalším příkladem detekce změny je úsek nahrávky rozhovoru dvou mluvčích hovořících bezprostředně po sobě. Opět byl proveden první krok metody binárního dělení, čímž jsme získali průběh zisku cesty G. Zde je již patrný nárůst hodnoty zisku G a nalezením jejího maxima *(G_{max}=112.9)* je i nalezeno místo s nejpravděpodobnější změnou řečníka. Správnost výpočtu - čas změny t_c=*3,4s*, lze ověřit porovnáním s časem získaným ruční segmentací t_r=*3,5s*.

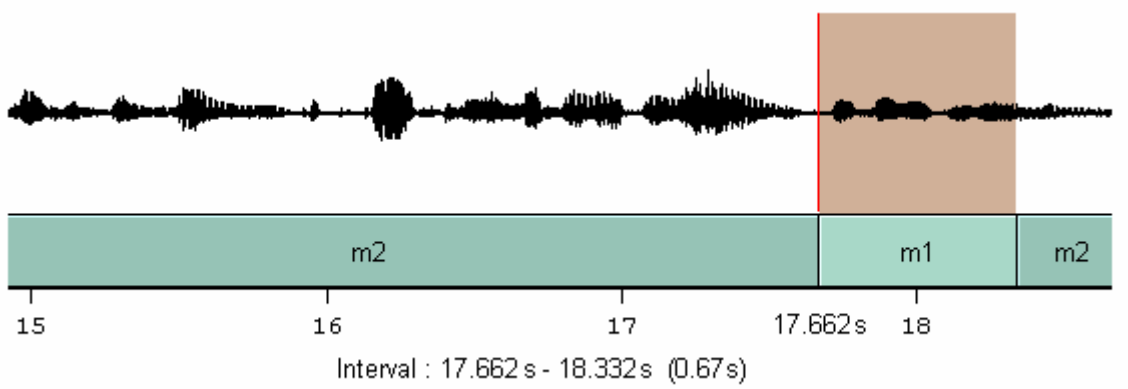


Obr. 9: Změna v jednom bodě

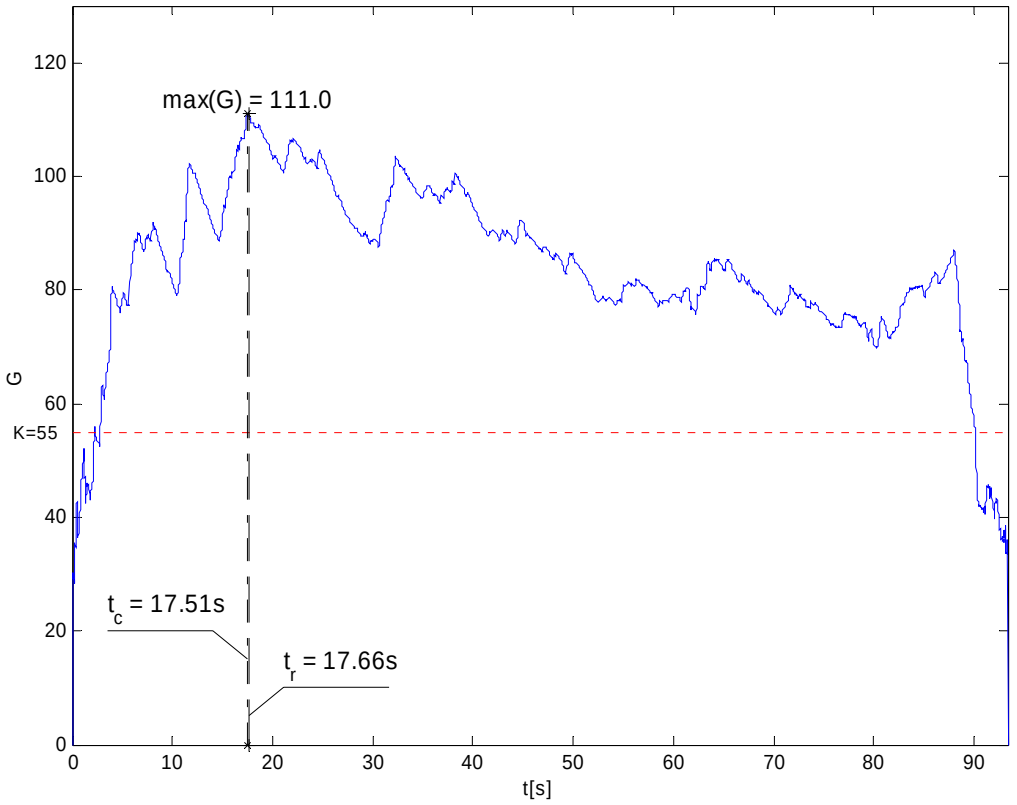


Obr. 10: Průběh zisku cesty G jedné změny, t_c – vypočtená změna, t_r – skutečná změna

V předchozích dvou případech průběhu funkce *G* je nalezení maxima jednoduchou záležitostí. K nalezení méně než jedné změny totiž postačí první krok metody, tedy nalezení nejpravděpodobnějšího bodu změny. V reálném rozhovoru však dochází k mnoha změnám a je zapotřebí je detekovat všechny. Metoda binárního dělení tento problém řeší rozdělením nahrávky (resp. jejího příznakového vektoru) na dvě části

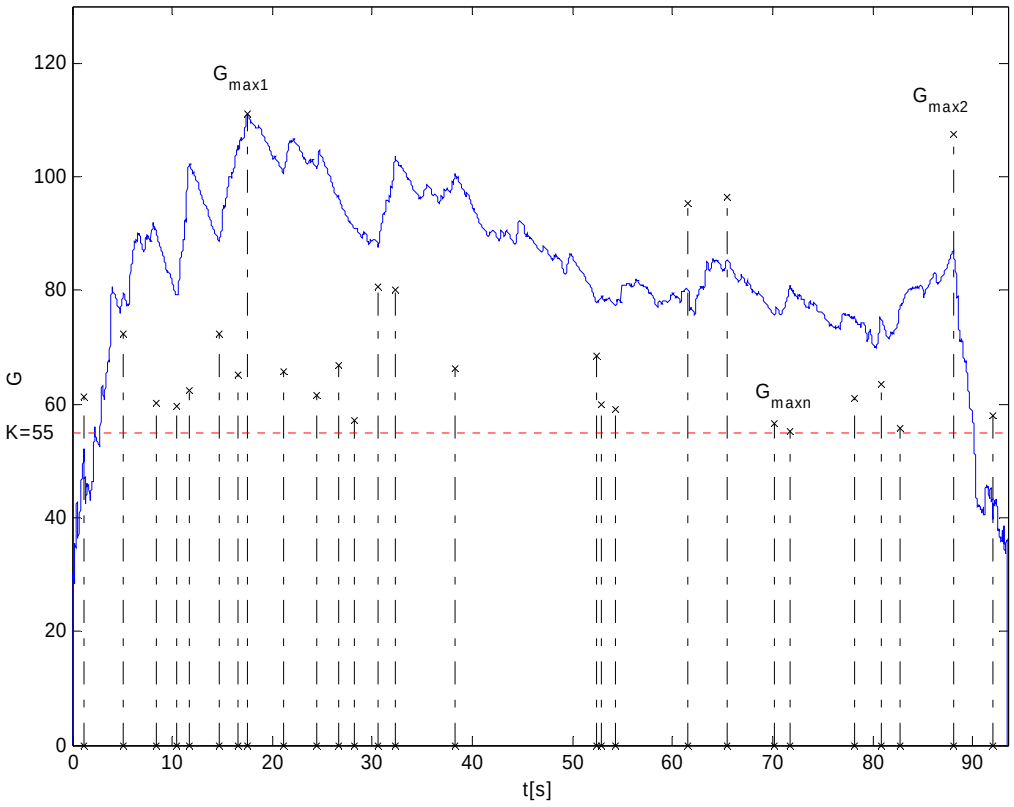


Obr. 11: Záznam hovoru s více změnami



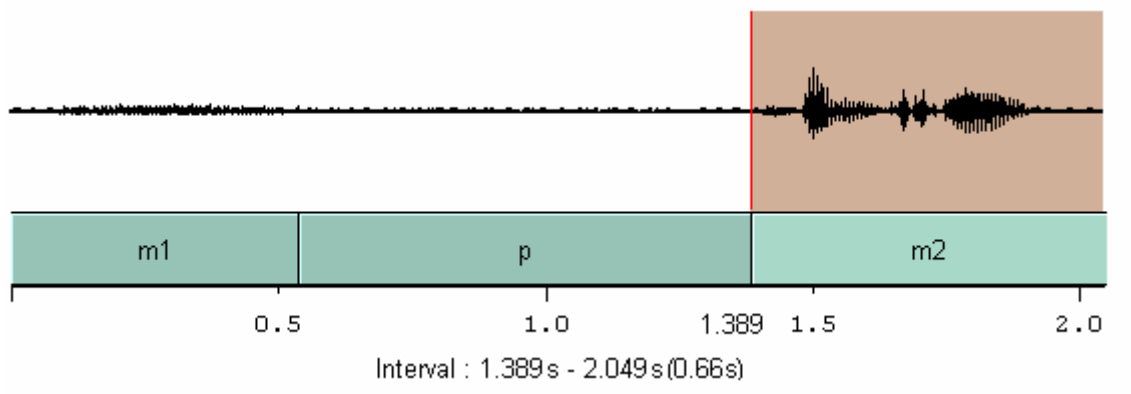
Obr. 12: Průběh zisku cesty G více změn, t_c – vypočtená změna, t_r – skutečná změna

právě v místě naposledy nalezeného bodu změny v prvním kroku a následným hledáním maxima zisku cesty v těchto nově vzniklých prostorech. Tento postup se opakuje až do doby, kdy již metoda nenalezne žádný nový bod změny. Opět je možné porovnat vypočtenou změnu t_c prvního kroku a ručně označenou změnu t_r. Na obrázku (Obr.11) je zobrazen krátký úsek hovoru s dvěma změnami mluvčích. Konkrétně se jedná o zhruba 4 sekundy z nahrávky dlouhé 90 sekund. Pokud srovnáme výsledek detekce změn po všech krocích metody (opět s nastavením $K=55$) a průběh zisku cesty G, můžeme pozorovat, že nalezené změny jsou v místech lokálních maxim této funkce. To, že se neshoduje jejich hodnota, je zapříčiněno tím, že byla hodnota zisku cesty G počítána z již vybraného úseku příznakového vektoru rozděleného předchozím krokem metody.

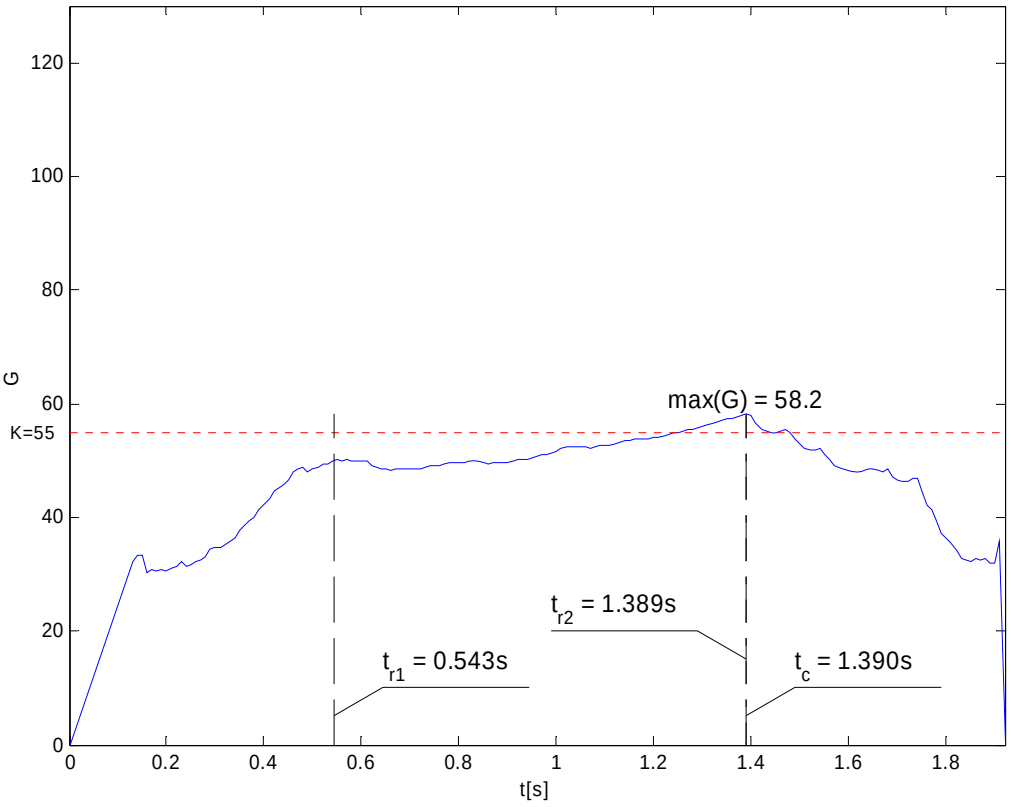


Obr. 13: Zisky cest G_{max} všech kroků metody

Je-li obdobně zkoumán úsek nahrávky, kdy po prvním mluvčím v hovoru nenásleduje ihned mluvčí druhý, ale je mezi nimi delší pauza, předpokládáme také správné označení bodu změny, protože i toto samotné "ticho" má v porovnání s jedním, či druhým mluvčím svůj specifický popis kepstrálními příznaky. Při ruční segmentaci je každé ticho větší než *0,5s* intuitivně označeno jak na jeho počátku, tak i na konci.

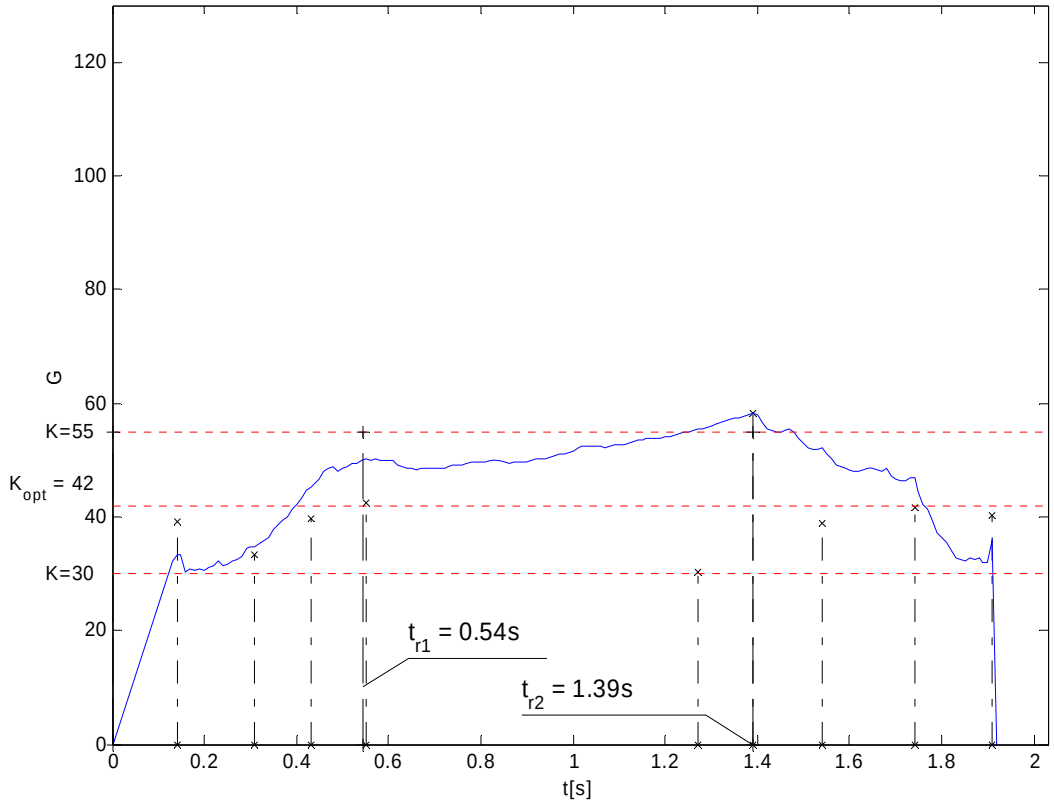


Obr. 14: Záznam hovoru s "tichem" mezi mluvčími větší než 0,5s



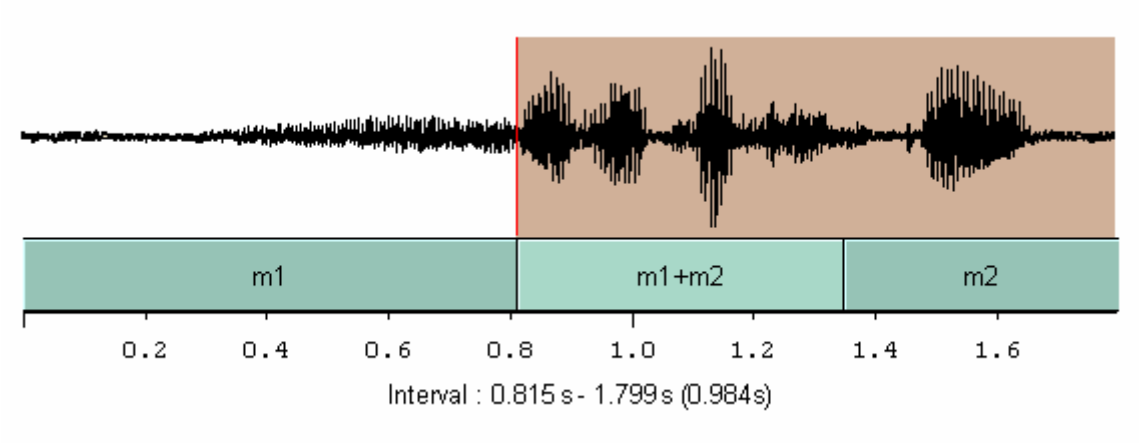
Obr. 15: Průběh zisku cesty G hovoru s "tichem" mezi mluvčími větší než 0,5s, t_c – vypočtená změna, t_{r1}, t_{r2} – skutečné změna

Pomocí metody binárního dělení je v prvním kroku správně odhadnuta změna v čase t_c=*1,39s*, v dalších krocích již ale pro kritickou hranici *K=55* žádné další změny nalezeny nejsou. Pokud bychom snížili kritickou hranici například až na hodnotu *K=30*, nalezneme sice v následujících krocích obě dvě skutečné změny, algoritmus však dále označí i místa navíc, kde ve skutečnosti ke změně nedošlo. Právě nastavením správného K_{opt}=*42* docílíme korektního rozpoznání změn. Hodnoty všech zisků *Gmax* jsou na následujícím obrázku Obr. 16:



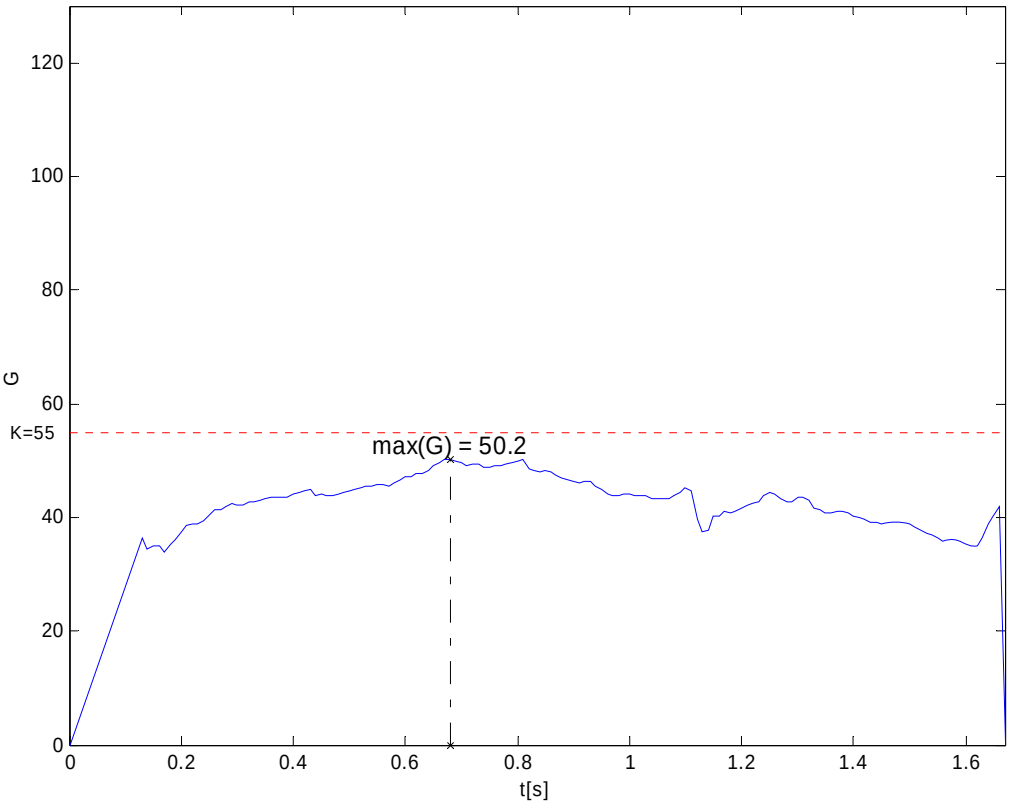
Obr. 16: Zisky cest G_{max} všech kroků metody hovoru s "tichem" mezi mluvčími větší než 0,5s

Za nejméně vhodná vstupní data, co se týče správného určení změny mluvčího, lze považovat nahrávky, kdy hovoří oba dva mluvčí současně. Příklad takové situace demonstruje obrázek (Obr. 17).



Obr. 17: Záznam překrývajícího hovoru dvou mluvčích

V čase *t=0,8 s* , kdy hovoří ještě stále mluvčí *m1*, začne hovořit navíc mluvčí *m2*. Jejich současný projev je značen jako úsek *m1 + m2*. Pokud by metoda binárního dělení měla správně označit tuto změnu, byla by očekávána právě v tomto úseku. Hodnota zisku cesty G však v tomto případě ani v jednom okamžiku nedosáhne kritické hranice, tedy bod změny nebude označen.



Obr. 18: Průběh zisku cesty G – překrývající se mluvčí

Samozřejmě platí, že obsahuje-li záznam více takovýchto míst, kdy hovoří oba dva mluvčí současně, dojde pravděpodobně k více neoznačeným míst bodů změn a tedy k negativnímu ovlivnění úspěšnosti metody.

1.5.2 Efektivní řešení metody binárního dělení

Již samotný postup volby intervalů ovlivňuje, kolik kroků budeme potřebovat k vypočtení všech bodů změn. V prvé řadě si definujeme tzv. *zisk cesty* ***G***. Vyjdeme z výše zmíněného vztahu a upravíme ho pro naše potřeby aplikace na zvolený interval vektoru parametrů

$$G(t|a,b) = \alpha\sqrt{\left((b-a+1)\log\left|\hat{\Sigma}\right| - (t-a+1)\log\left|\hat{\Sigma}_1\right| - (b-t)\log\left|\hat{\Sigma}_T\right|\right)} - \beta \tag{18}$$

kde d je rozměr příznakového vektoru, dále pak platí $(a-d) > t > (b+d)$

$$\alpha = (2\log\log(b-a+1))^{\frac{1}{2}} \tag{19}$$

$$\beta = 2\log\log(b-a+1) + d\log\log\log(b-a+1) - \log\Gamma(d) \tag{20}$$

kde $\hat{\Sigma}, \hat{\Sigma}_1, \hat{\Sigma}_T$ jsou kovariance vektorů $\{x_a,...,x_b\}, \{x_a,...,x_t\}, \{x_{t+1},...,x_b\}$.

Dále definujeme dvě vlastnosti každého nově nalezeného bodu změny – *vrchol* V a to, zdali je *aktivní* – A, či nikoliv a jeho *pozici* – P. V každém kroku, je-li nalezen nový vrchol, jsou přepočítány pozice vrcholů a v závislosti na hodnotě maximálního zisku cesty je nastaven parametr A. Pro zamezení zbytečných kroků se počítá zisk cesty tehdy, jsou-li oba krajní vrcholy označeny jako aktivní.

Kromě minimalizace počtu kroků (počtu hledání změn) můžeme významně zkrátit dobu hledání dopředným výpočtem kovariančních matic. Místo opakovaného počítání těchto matic v každém kroku, je možné tyto matice spočítat již před samotným cyklem.

Vycházíme z příznakového vektoru, který má rozměr $T \times d$. Mějme dvě pole, z_1 rozměru d a z_2 rozměru $d \times d$.

$$\begin{aligned} z_1(t) &= z_1(t-1) + x_t \\ z_2(t) &= z_2(t-1) + x_t x_t' \end{aligned} \tag{21}$$

Pro výpočet střední hodnoty a kovarianční matice platí

$$\begin{aligned} \overline{X} &= E(X) \\ \Sigma &= E\left[(X-\overline{X})(X-\overline{X})'\right] \end{aligned} \tag{22}$$

a tedy

$$\Sigma = E\left[(X - E(X))(X - E(X))'\right] = \\ = E(XX') - \overline{XX}' \tag{23}$$

Po aplikaci na vektory z_1 a z_2 dostáváme vztahy, pomocí kterých si můžeme připravit kovarianční matice pro celý příznakový vektor a není je třeba počítat v každém kroku hledání.

$$\hat{\mu} = \frac{z_1(b) - z_1(a-1)}{b - a + 1} \\ \hat{\Sigma} = \frac{z_2(b) - z_2(a-1)}{b - a + 1} - \hat{\mu}\hat{\mu}' \tag{24}$$

1.6 Metoda vyhodnocení úspěšnosti detekce změny mluvčího

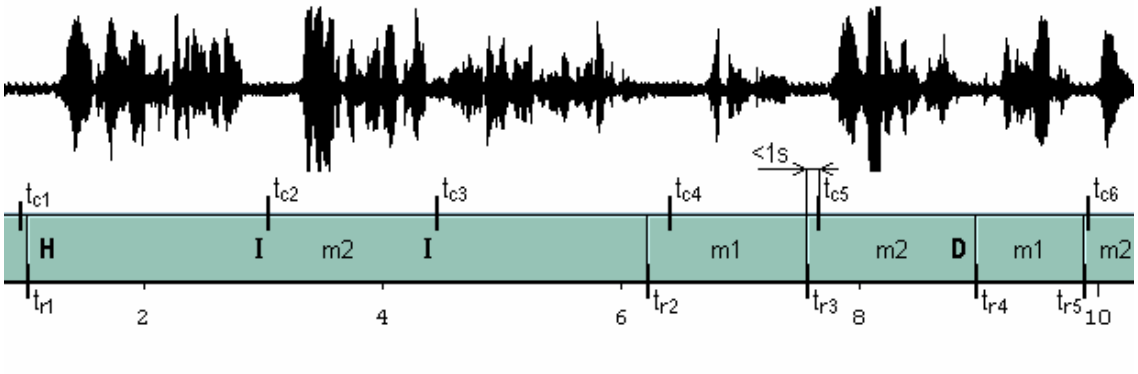
Míra úspěšnosti metody vyhledávání bodů změn se určuje na základě trénovacích nahrávek (ručně indexované změny) a časových značek vystupujících z rozpoznávače. Jak se uvádí v [8], $i-tý$ vypočtený bod změny t_{ci} považujeme za správně nalezený (HIT) a odpovídající $j-tému$ referenčnímu bodu změny t_{rj} tehdy a jen tehdy, když

1. t_{ci} je vypočtený bod změny nejblíže referenčnímu t_{rj},

2. t_{rj} je referenční bod změny nejblíže vypočtenému t_{ci},

3. vzdálenost mezi nimi je menší, než určitá mez $\tau_{\max}$, typicky $\left|t_{ci} - t_{rj}\right| \langle 1s$.

Takto nalezené dvojice časů nazýváme "pár" a označujeme je H. Pokud rozpoznávač v nahrávce objevil nějakou změnu navíc, označíme tuto časovou značku jako "inzerce" – značíme I, pokud naopak nějakou změnu mluvčího nerozpoznal, hovoříme o tzv. "delecích", které označujeme D.



Obr. 19: Označení H...párů, I...inzercí, D...delecí

V závislosti na zvolené kritické hranici pro zamítnutí nulové hypotézy H_0 se nám mění počet inzercí I a počet delecí D. Pokud máme tuto hranici zvolenou jako příliš nízkou (to znamená, že i sebemenší změna parametrů bude vést na označení bodu jako změna), máme sice zaručeno, že spolehlivě odhalíme všechny změny mluvčího, ale výsledkem budou ještě označena místa, kde ke změně mluvčího nedošlo. Je tedy třeba nalézt určitý kompromis, který nám bude nejlépe popisovat daný výsledek.

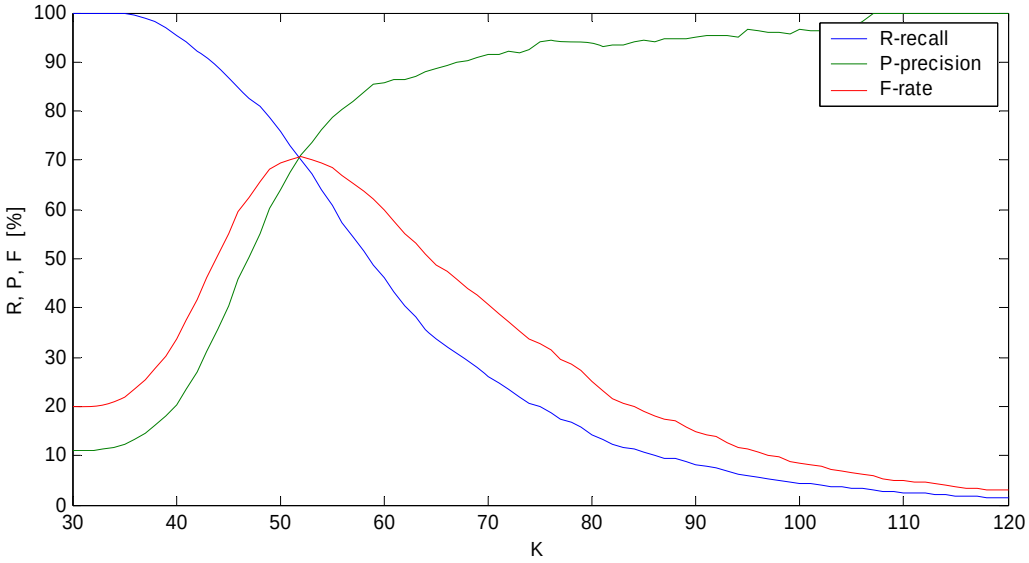
Abychom lépe vyjádřili úspěšnost rozpoznávače, je vhodné vyjádřit počet inzercí I a delecí D procentuálně [8]:

$$R = \frac{H}{N} \times 100\% \tag{25}$$

$$P = \frac{H}{H + I} \times 100\% \tag{26}$$

$$F = \frac{2 \times R \times P}{R + P} \tag{27}$$

kde $N = H + D$ je počet všech referenčních bodů, R je tzv. míra *recall* , která popisuje procento všech správně nalezených změn vůči všem hledaným změnám, P je míra *precision* popisující procento všech správně nalezených ze všech nalezených změn. Funkce těchto dvou měr závislé na zvolené kritické hranici mají opačnou tendenci viz. Obr.20. Poslední míra, tzv. míra *F-rate* vyjadřuje vzájemnou závislost obou předchozích měr a nalezením jejího maxima získáváme optimální kritickou hranici vzhledem k počtu *inzercí* a *delecí*.



Obr. 20: Průběh měr recall, precision, F – rate

2 Implementace metody binárního dělení

Jak již bylo v předchozích kapitolách zmíněno, k rozpoznávání jedné změny bylo využito metody *maximální věrohodnosti*, na jejímž základě je postavena metoda pro rozpoznávání více bodů změn – *metoda binárního dělení*. Tato metoda opakovaně aplikuje testování hypotéz na vektor parametrů a počítá tak nejvýznamnější body změny řečníka do té doby, dokud odhad vyhovuje zvolené kritické hranici. Tato hranice je předem stanovena trénováním na trénovacích datech.

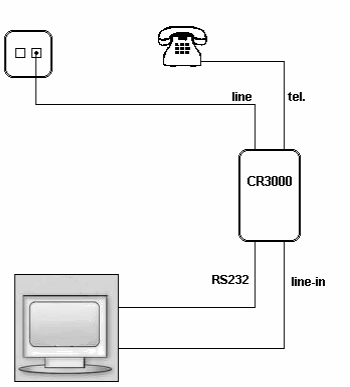
2.1 Pořízení trénovacích a testovacích dat

Abychom mohli metodu binárního dělení testovat, bylo nejprve nutné pořídit dostatečný počet dat k jejímu natrénování a následnému ověření její úspěšnosti. Mezi samotným záznamem telefonního hovoru a přepočtem nahrávky na vektor příznaků je několik nezbytných operací, kam patří například převzorkování, či ruční označení bodů změn mluvčích v nahrávce.

2.1.1 Záznam telefonního hovoru, Call Recorder CR 3000

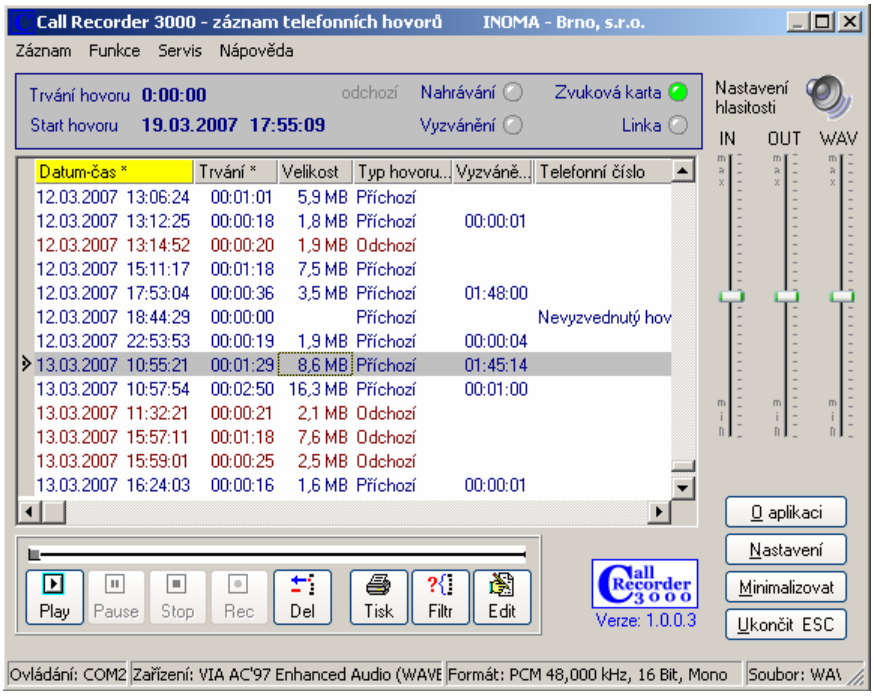
Pokud je třeba zaznamenávat hovory přímo z telefonní linky pomocí AD převodníku zvukové karty, je nutné tento signál upravit na úroveň signálu přijatelného pro analogový vstup karty (střídavé napětí 0-5V). Telefonní linka zpravidla obsahuje stejnosměrnou složku napájecího napětí 60V a střídavé vyzváněcí napětí (obvykle 75V, 25Hz) a při zprostředkování spojení samotný signál rozhovoru. První problém je tedy rozpoznat, kdy je telefon ve stavu vyzvánění a kdy již probíhá samotný hovor a podle toho buď zcela odpojit telefonní přístroj od vstupu zvukové karty (pokud telefon vyzvání), nebo pouze odstranit stejnosměrnou složku napětí na telefonní lince (při hovoru).

Problém detekce příchozího spojení a nahrávání hovoru pomocí AD převodníku zvukové karty řeší přístroj Call Recorder CR 3000. Jak je z obrázku patrné, o připojování signálu hovoru ke zvukové kartě se stará modul CR 3000, který s pomocí sériového rozhraní komunikuje s aplikací obsluhující záznam signálu z analogového vstupu. Pokud je telefon ve stavu "čekání na hovor", nebo pokud telefon "vyzvání", je na telefonní lince napětí vyšší, než



Obr. 21: Zapojení zařízení CR3000

je maximální možné napětí vstupu zvukové karty. Navíc se na lince objevuje i stejnosměrná složka, která napájí telefonní přístroj, modul tedy analogový vstup od linky odpojí. Pokud je ovšem sluchátko telefonu zvednuté (probíhá vytáčení čísla, nebo již samotný hovor), modul připojí linku ke vstupu, aplikace je o tomto stavu informována pomocí sériového rozhraní a spouští záznam hovoru. Po ukončení rozhovoru opět odpojí linku od vstupu zvukové karty a komunikuje s aplikací, která ukončuje nahrávání. Aplikace dokáže rozeznat, jde-li o příchozí rozhovor, či odchozí a zaznamenává informace o času pořízení nahrávky. Hovory jsou zaznamenávány s rozlišením 16 bitů, vzorkovány frekvencí 48 kHz. Tyto parametry je možné měnit pomocí nastavení.



Obr. 22: Okno aplikace Call Recorder 3000

2.1.2 Třídění nahrávek, převzorkování

Záznamy hovorů pořízené aplikací Call Recorder 3000 byly setříděny, pojmenovány a následně převzorkovány na frekvenci 16 kHz. Převzorkování bylo provedeno pomocí programu Sound Exchange. Ten se spouští s následujícími parametry:

sox 01MM_16kHz.wav –r 16000 01MM_16kHz.wav,

Prvním parametrem je vstupní soubor, parametr *–r 16000* označuje proceduru převzorkování (16 kHz) a poslední parametr označuje výstupní soubor.

2.1.3 HTK, HCopy

Parametrizaci signálu provádíme pomocí nástroje HCopy, který je součástí programového balíku HTK. Tento software slouží k trénování a rozpoznávání pomocí skrytých Markových modelů (HMM), dále k již zmíněné parametrizaci řečových signálů a vyhodnocování jejich rozpoznávání. Program HCopy zkopíruje zvukový záznam ze zdrojového souboru a přitom umožňuje provádění různých konverzí, podle předem zadaného konfiguračního souboru, např.:

hcopy -C par06.cfg 01MM_16kHz.wav 01MM_16kHz.par01

Za parametrem *–C* následuje konfigurační soubor, dále je uveden zdrojový a cílový soubor.

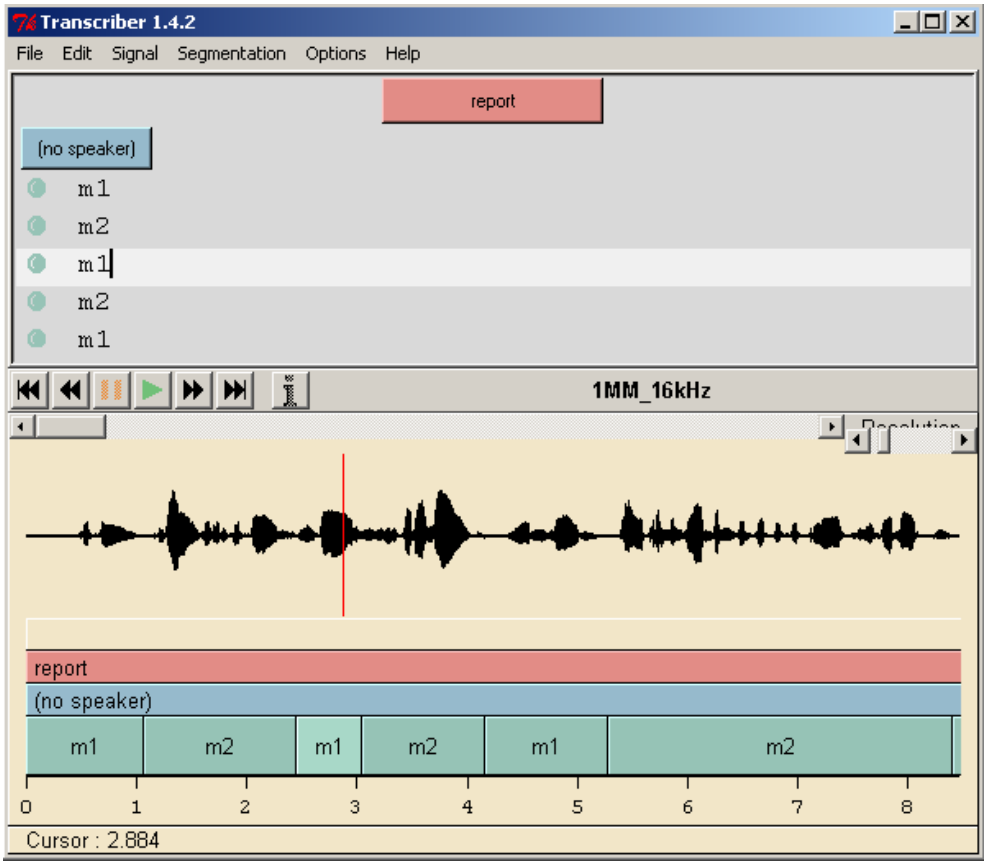
Konfigurační soubor par06.cfg:

```
ENORMALISE = T          -normování energie
EXTENDFILENAMES = T
NUMCEPS = 12            -počet kepstrálních příznaků
NUMCHANS = 24           -počet trojúh. filtrů pro výpočet MFCC příznaků
PREEMCOEF = 0.97        -koeficient preempfáze
SOURCEFORMAT = WAVE     -formát zdrojového souboru
SOURCEKIND = WAVEFORM
TARGETFORMAT = HTK      -formát cílového souboru
TARGETKIND = MFCC_E     -typ výstupních parametrů (_E-log energie, _D-
                         delta příznaky, _A-double delta příznaky,
                         _0-nultý kepstrální koeficient)
TARGETRATE = 100000     -vzorkovací perioda výstupních vektorů (10ms)
USEHAMMING = T          -aplikace Hammingova okénka
WINDOWSIZE = 250000     -délka okénka (25ms)
```

2.1.4 Referenční data

Pro správné vyhodnocení výsledků nalezených bodů změn je potřeba stanovit referenční časy změn. To, jak je metoda v hledání úspěšná, či nikoliv, je závislé jak na kvalitě nahrávky, jejím charakteru (jak často například dochází k situaci, kdy hovoří oba mluvčí současně), ale i na pečlivém "oindexování" trénovací nahrávky. Pokud jsou chybně stanoveny referenční časy, nemůžou ani výsledky metody vycházet v přijatelných mezích.

Indexování je prováděno pomocí programu Transcriber, který je volně ke stažení [10] a umožňuje nejen označení časových značek, ale i popis jednotlivých segmentů společně s popisem jejich obsahu.



Obr. 23: Ruční indexování záznamu

Aplikace ukládá jednotlivé transkripce v XML formátu a je tedy nutné je během vyhodnocování výsledků převádět do podoby, s kterou můžeme v prostředí MATLAB pracovat. K tomuto účelu byla naprogramována funkce pro načítání XML dat ze souboru viz. příloha [A]. Funkce se pomocí skriptu jazyka Perl opakovaně volá se vstupním parametrem cesty k souborům vypočtených bodů změn a ta dále předává

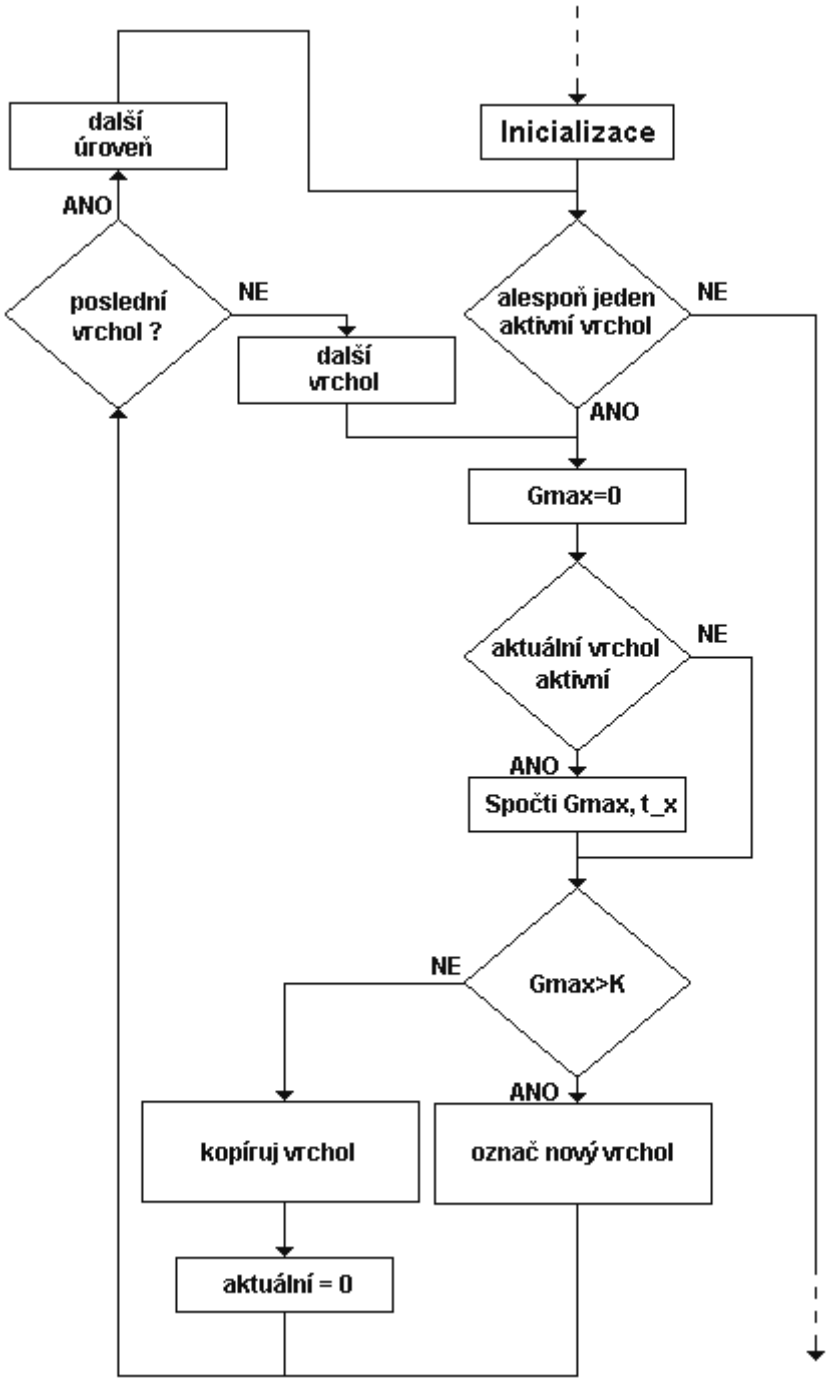
proměnné *tsi* a *trj* další funkci pro výpočet počtu I, D, N, H metodou nejbližšího souseda (*nearest neighbour*).

2.2 Implementace metody binárního dělení

Princip této metody spočívá v opakovaném aplikování výpočtu hodnoty *zisku cesty G* (17) na zvolený interval vektoru parametrů. V kap.1.5.1 byla *metoda binárního dělení* zjednodušeně vyobrazena jako postupné dělení tohoto intervalu, které bylo limitováno kritickou hranicí *K*. Nyní si uvedeme podrobnější popis algoritmu.

2.2.1 Popis algoritmu, vývojový diagram

Před samotným spuštěním programu je v inicializačním bloku načten vektor parametrů, jsou definovány matice pro uchovávání informace o tom, zda-li je aktuální vrchol aktivní - *A(v,u)* a na jaké pozici se v dané vrstvě vrchol nachází – *P(v,u)*, kde $u = 0,...,U$ je číslo aktuální úrovně a $v = 0,...,V(u)$ je číslo vrcholu. Jsou určeny první dva krajní vrcholy, druhý z nich je nastaven jako aktivní. Dále jsou v této části vypočítány kovarianční matice, potřebné pro pozdější výpočet zisku cesty. Pokud algoritmus nalezne v aktuální úrovni alespoň jeden aktivní vrchol (na počátku vyhledávání je jím poslední segment T), hledá první aktuální vrchol a počítá bod s největším ziskem cesty G_{max}. Vyhovuje-li hodnota zisku cesty omezující podmínce $G_{max}>K$, uloží pozici nalezeného aktuálního vrcholu spolu s místy ohraničujícími prohledávaný interval do další úrovně, v opačném případě pouze kopíruje tyto krajní body intervalu a nastavuje aktuální vrchol jako neaktivní. Nachází-li se v dané úrovni ještě další aktivní vrchol, postup hledání G_{max} se opakuje. Poté proces vyhledávání přechází do další úrovně a znovu se vyhodnocuje podmínka cyklu *while* – "Je alespoň jeden aktivní vrchol v dané úrovni?". Po ukončení tohoto cyklu jsou nalezené časy změn předány k dalšímu zpracování a vyhodnocení.



Obr. 24: Algoritmus metody binárního dělení

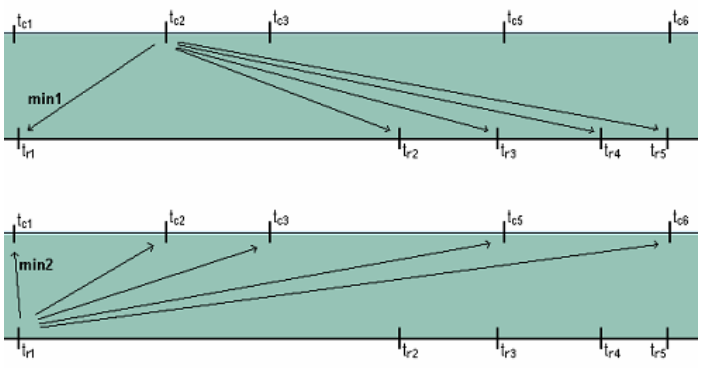
2.2.2 Trénování metody, odhad kritické hranice

Pro dosažení přijatelných výsledků při hledání bodů změn na různých typech nahrávek je třeba nalézt všechny volné parametry metody, tzv. natrénování algoritmu. Především je třeba uvažovat různorodost nahrávek z hlediska rozdílného frázování při hovoru, například většího počtu pauz, oproti hovoru, kdy se segmenty jednotlivých mluvčích překrývají. Jak již bylo zmíněno, metoda binárního dělení má tento volný

parametr pouze jeden. K trénování metody je využit předchozí algoritmus pro hledání bodů změn, který navíc při každém nalezení nového vrcholu zaznamená hodnotu zisku cesty – matice *K(v,u)*. To, jaká hodnota zisku je ještě zaznamenána a která již ne (maximální a minimální počet rozpoznaných změn), je dáno při úvodní inicializaci konstantami K_{min} a K_{max} s tím, že kritická hranice $K=K_{\min}$. Po ukončení algoritmu binárního dělení je pro každou hodnotu K z intervalu $K_{min} \div K_{max}$ vypočten vektor bodů změn t_{ci} na základě kterého jsou stanoveny počty H, I, D. Z průběhu míry *F-rate*, resp. nalezením jejího maxima zjistíme hledané K_{opt}.

2.3 Zpracování výsledků, algoritmus určení počtu H,I,D

Výsledkem algoritmu binárního dělení je vektor vypočtených časových značek - t_{ci}. Je-li k dispozici příslušný vektor získaný ruční segmentací - t_{rj}, je možné jednoduše metodou nejbližšího souseda (*nearest neighbour*) určit počet *párů - H*, *inzercí - I* a *delecí - D*. V algoritmu je v prvním kroku hledán minimální rozdíl časů mezi jednou konkrétní hodnotou času t_{ci} a všemi hodnotami času t_{rj}. Po nalezení minimálního rozdílu je v druhém kroku hledáno minimum z druhé strany, tedy pevnou časovou hodnotou je nyní nalezený čas t_{rj} a k němu hledáme čas t_{ci}. Pokud je hledáním z druhé strany nalezen původní čas t_{ci}, kterým bylo hledání započato, je možné označit značky t_{ci} a t_{rj} jako *pár – H*.



Obr. 25: Obousměrné hledání párů H, inzercí I a delecí D

Protože k výpočtu úspěšnosti metody nejsou potřeba konkrétní páry, inzerce a delece, ale pouze jejich počet, stačí inkrementovat jednu proměnnou, která uchovává počet všech párů a na základě její hodnoty vypočítat počet inzercí a delecí:

$$I = i - H \qquad (28)$$
$$D = j - H$$

,kde i je počet všech vypočítaných časů a j je počet všech referenčních časů.

3 Vyhodnocení metody binárního dělení

Aby bylo možné určit, jak je daná metoda testování změn mluvčích vhodná pro dané nahrávky pořízené z reálných telefonních rozhovorů, či nikoliv, je třeba trénováním metody na dostatečném počtu nahrávek stanovit kritickou hranici K_{opt} a následně ji podrobit testování na testovacích datech. Čím větší počet trénovacích a testovacích dat je k dispozici, tím lze dosáhnout vypovídavějších výsledků.

3.1 Trénování metody

V prvé řadě bylo nutné určit, jaké nahrávky jsou vhodné jako trénovací. Je důležité, aby nahrávky byly různorodé a tím tak byla zajištěna dostatečná úspěšnost hledání změn pro všechny typy nahrávek. Převládat by měly hovory s větším počtem změn. Dále bylo nutné stanovit, jaké příznaky k trénovaní a následnému testování použít. Standardně se pro rozpoznávání změny řečníka využívá melfrekvenčních kepstrálních koeficientů (c_1 až c_{12}). Ty byly vypočítány pomocí nástroje HTK tools. Jednoduchým pozměněním konfiguračního souboru tohoto nástroje, lze vypočítat i další příznaky, jako například LPC příznaky, je možné určit počet použitých trojúhelníkových filtrů MFCC příznaků, šířku pásma vstupního signálu, dále můžeme příznakový vektor doplnit o logaritmus energie, nultý kepstrální příznak (c_0) a další.

3.1.1 Příznaky pro potřeby trénování

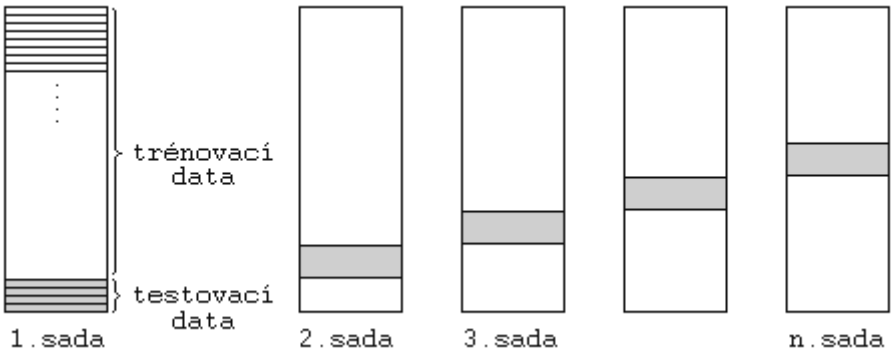
Od základního nastavení konfiguračního souboru nástroje HTK tools jsou odvozeny další varianty parametrizace. V následující tabulce je seznam konfiguračních souborů. První jsou nastavení s různým počtem trojúhelníkových filtrů, pomocí parametru NUMCHANS=24,...,27. Parametry LOFREQ=300 a HIFREQ=3400 je možné stanovit frekvenční pásmo pro výpočet příznaků, čehož se využívá při parametrizaci akustického signálu s omezeným frekvenčním pásmem, jako je např. telefonní signál. Přidáním parametru _E a _0 je možné příznakový vektor rozšířit o logaritmus energie, či o tzv. nultý kepstrální příznak (c_0). Dále následují konfigurace s různým počtem kepstrálních příznaků NUMCEPS=7,...,12. V případě použití LPC kepstrálních příznaků je možné měnit řád lineárního modelu použitého při výpočtu příznaků LPCORDER=14,...,10, počet příznaků a opět doplnění o logaritmus energie.

Konfigurační soubor	Nastavení parametrů
par01.cfg	MFCC NUMCHANS = 27 NUMCEPS = 12
par02.cfg	MFCC NUMCHANS = 26
par03.cfg	MFCC NUMCHANS = 25
par04.cfg	MFCC NUMCHANS = 24
par05.cfg	MFCC LOFREQ = 300 HIFREQ = 3400 NUMCHANS = 24
par06.cfg	MFCC_E NUMCHANS = 24
par07.cfg	MFCC_0 NUMCHANS = 24
par08.cfg	MFCC_E_0 NUMCHANS = 24
par09.cfg	MFCC NUMCEPS = 11 NUMCHANS = 24
par10.cfg	MFCC NUMCEPS = 10 NUMCHANS = 24
par11.cfg	MFCC NUMCEPS = 9 NUMCHANS = 24
par12.cfg	MFCC NUMCEPS = 8 NUMCHANS = 24
par13.cfg	MFCC NUMCEPS = 7 NUMCHANS = 24
par16.cfg	LPC LPCORDER = 14 NUMCEPS = 12
par17.cfg	LPCORDER = 13
par18.cfg	LPCORDER = 12
par19.cfg	LPCORDER = 11
par20.cfg	LPCORDER = 10
par21.cfg	LPC LPCORDER = 14 NUMCEPS = 11
par22.cfg	NUMCEPS = 10
par23.cfg	NUMCEPS = 9
par24.cfg	NUMCEPS = 8
par25.cfg	NUMCEPS = 7

Tab: 1 Tabulka použitých parametrizací pro trénování a testování

3.1.2 Trénovací a testovací databáze THSZ07

Databáze trénovacích a testovací nahrávek obsahuje celkem 58 hovorů v celkové době záznamu 51,3 minut. Hovory byly ukládány ve formátu WAV, se vzorkovací frekvencí 16kHz. Jelikož celkový počet 58 pořízených a ručně transkribovaných nahrávek nelze považovat za dostatečně rozsáhlou databázi, je pro trénování a následné testování využito principu rotace trénovacích a testovacích dat. Z celkové množiny nahrávek je vždy vybráno zhruba 90% dat jako trénovacích, zbylý počet jako testovacích. Tímto výběrem dat nám vznikne jedna sada trénovacích a testovacích dat. Další sady dostáváme výběrem jiných nahrávek jako testovacích a zbylých jako trénovacích.



Obr. 26: Princip rotace trénovacích a testovacích dat

Pro každou sadu lze trénováním určit K_{opt} a na základě takto stanovené kritické hranice vypočítat pro testovací nahrávky procentuální úspěšnost metody. Výsledkem trénování je pak $\varnothing K_{opt}$. Jako výsledek testování metody je směrodatná její průměrná procentuální úspěšnost na všech testovacích nahrávkách. Tímto způsobem lze tedy jednoduše docílit většího počtu trénovacích a testovacích dat a tedy statisticky významnějších výsledků.

3.1.3 Výsledky trénování

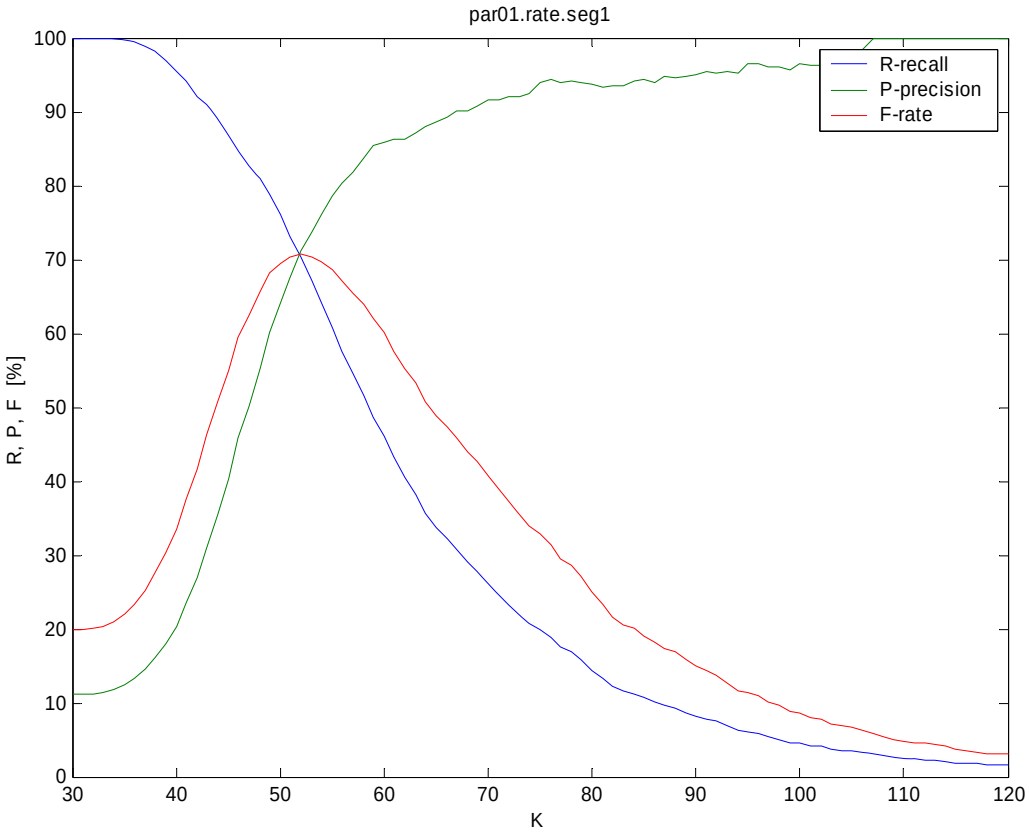
Pro každý druh parametrizace a každou nahrávku byl metodou binárního dělení vytvořen soubor dat, obsahující všechny vypočtené body změn. Pro snadnější orientaci v množství těchto dat byla jména výstupních souborů pojmenována podle názvu nahrávky, pro kterou výpočet proběhl a názvu konfiguračního souboru provedené parametrizace. Z těchto dat a jim příslušných referenčních časů byly dále určeny počty I, D, N, H a po následné maximalizaci míry F-rate bylo pro každou sadu trénovacích dat určeno K_{opt}.

částečný výpis dat souboru par01.rate.seg1:

Kopt = 52 F = 70.6%			
K= 48:	F=65.6	P=55.2	R=81.0
K= 49:	F=68.2	P=60.1	R=78.9
K= 50:	F=69.5	P=64.1	R=76.0
K= 51:	F=70.2	P=67.5	R=73.2
K= 52:	**F=70.6**	**P=71.1**	**R=70.2**
K= 53:	F=70.3	P=73.7	R=67.1
K= 54:	F=69.6	P=76.2	R=64.1
K= 55:	F=68.6	P=78.7	R=60.8
K= 56:	F=67.0	P=80.4	R=57.5

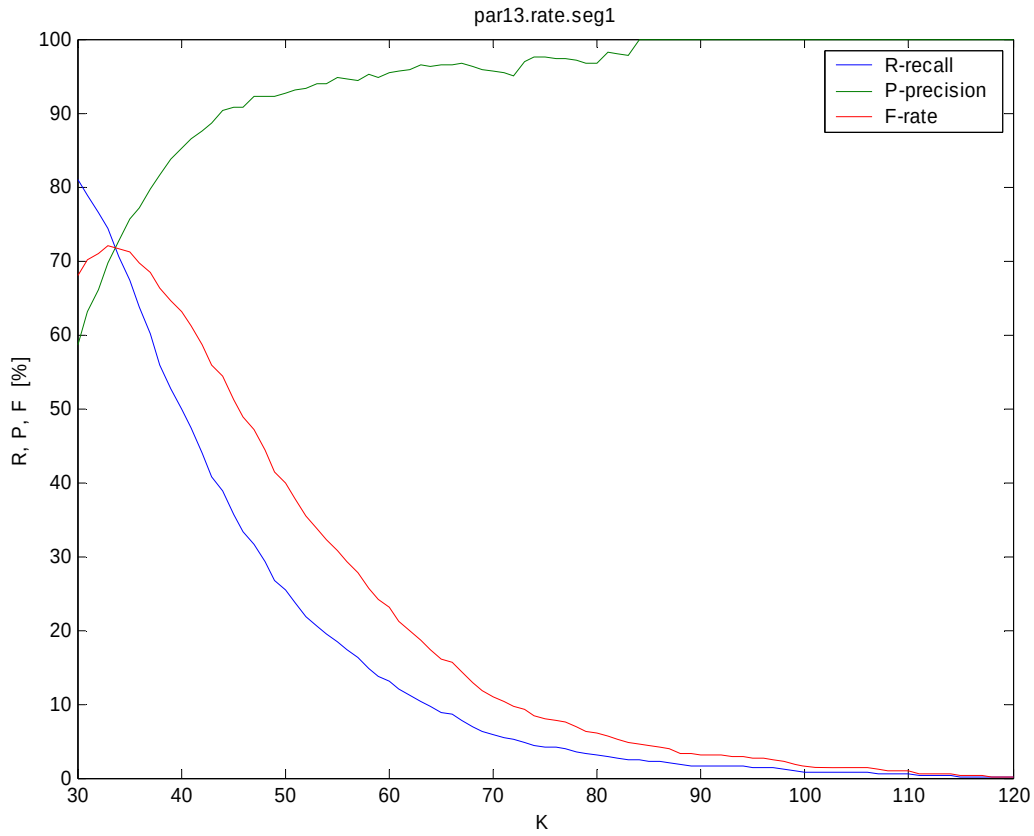
Tab: 2 Příklad výsledku trénování sady seg1 a parametrizace par01.cfg

Tento příklad ilustruje výsledek po trénování na datech sady *seg1* příznakového vektoru vypočítaného dle nastavení *par01.cfg*. Trénování bylo provedeno pro rozsah K=30 až K=120. Graficky zobrazený průběh měr *F-rate*, *P-precision* a *R-recall* je zobrazen na následujícím obrázku:



Obr. 27: Výsledek trénování sady dat seg1 – par01.cfg

Danou parametrizací byl vypočten příznakový vektor MFCC příznaků s počtem 27 kanálů trojúhelníkového filtru. Trénováním bylo stanoveno $K_{opt}=52$ sady *seg1*. Jak se dalo očekávat, snižováním počtu kepstrálních příznaků, ať již u MFCC příznaků, nebo LPC kepstrálních příznaků se snižuje trénováním stanovená optimální kritická hranice metody rozpoznávání, což dokazuje následující graf:



Obr. 28: Výsledek trénování 1. sady dat – par13.cfg

V tomto případě je příznakový vektor složen ze sedmi MFCC kepstrálních příznaků (NUMCHANS=24) a můžeme pozorovat, že optimální hranice pro testování je výrazně nižší, v porovnání s předchozím příkladem. Nižší K_{opt} má za následek nastavenou vyšší citlivost rozpoznávače a s tím je i spojené možné vyšší procento inzercí ve výsledku. Podobně se chová i trénování metody při parametrizaci LPC kepstrálními příznaky. Je-li počet příznaků NUMCEPS=12, pohybuje se K_{opt} okolo hodnoty K=50. Postupným snižováním počtu až k NUMCEPS=7 se K_{opt} pohybuje kolem hodnoty K=32.

Výsledek trénování všech sad trénovacích dat pro jednotlivé parametrizace a vypočítané průměrné hodnoty Kopt, včetně výběrové směrodatné odchylky s_x jsou uvedeny v Tab: 3:

Druh parametrizace	K_{opt}									$\varnothing K_{opt}$	s_x
	seg1	seg2	seg3	seg4	seg5	seg6	seg7	seg8	seg9		
par01	52	52	52	52	52	52	52	52	52	52,00	0,00
par02	52	53	52	52	52	52	52	51	53	52,11	0,60
par03	52	54	52	52	53	52	52	52	53	52,44	0,73
par04	52	52	52	52	52	52	52	52	53	52,11	0,33
par05	49	49	49	49	49	49	49	49	49	49,00	0,00
par06	57	57	57	57	57	57	57	57	57	57,00	0,00
par07	57	57	57	57	57	57	57	57	57	57,00	0,00
par08	62	62	62	62	62	62	62	62	62	62,00	0,00
par09	48	48	48	48	48	48	48	48	48	48,00	0,00
par10	44	44	44	44	44	44	44	44	44	44,00	0,00
par11	41	40	40	41	41	41	41	41	41	40,78	0,44
par12	37	36	36	36	37	37	36	37	36	36,44	0,53
par13	33	33	33	33	33	33	33	33	33	33,00	0,00
par16	49	51	51	49	51	51	49	49	49	49,89	1,05
par17	50	50	50	50	50	50	50	50	50	50,00	0,00
par18	51	51	51	51	51	51	51	51	51	51,00	0,00
par19	46	48	46	46	46	46	46	46	46	46,22	0,67
par20	44	42	42	42	42	42	42	43	42	42,33	0,71
par21	46	47	47	46	46	46	46	46	47	46,33	0,50
par22	43	43	43	43	43	43	43	43	43	43,00	0,00
par23	39	39	39	39	39	39	39	39	39	39,00	0,00
par24	36	37	37	35	37	36	35	36	36	36,11	0,78
par25	32	32	32	32	32	32	32	32	32	32,00	0,00

Tab: 3 Hodnoty K_{opt} pro jednotlivé sady dat a parametrizace

Můžeme si všimnout, že v mnohých případech dopadlo trénování jednotlivých sad shodně. Pro některé parametrizace dokonce vychází výběrová směrodatná odchylka $s_x=0$. Je to způsobeno tím, že se trénovací databáze skládá z nahrávek podobného charakteru. Pokud bychom měli k dispozici nahrávky, které se od sebe, co se týče jejich popisu pomocí příznaků, výrazněji liší, dosahovaly by výběrové směrodatné odchylky natrénovaných dat K_{opt} vyšších hodnot.

3.2 Testování metody

V této kapitole jsou uvedeny výsledky testování zvolené metody rozpoznávání na pořízené testovací databázi při nastavení různých druhů parametrizací. Navíc jsou pro srovnání zmíněny i výsledky rozpoznávání změn mluvčích na základě nahrávek různých televizních a rozhlasových pořadů.

3.2.1 Výsledky testování

Trénováním metody je pro každou sadu dat vypočtena hodnota kritické hranice K_{opt}, na základě které je dále metoda testována na testovacích datech. Pro každou sadu testovacích dat a každou konkrétní nahrávku je takto zajištěn počet H, I, D, N a jsou vypočteny procentuální úspěšnosti (míry *F-rate*, *precision*, *recall*). Ty jsou následně zprůměrovány a ukazují úspěšnost zvolené metody při použití konkrétní parametrizace viz. **Chyba! Nenalezen zdroj odkazů.**.

Kopt	I	D	N	H	F [%]	P [%]	R [%]
52	15	21	62	41	69,5	73,2	66,1
52	4	15	21	6	38,7	60,0	28,6
52	4	6	20	14	73,7	77,8	70,0
52	23	53	117	64	62,7	73,6	54,7
52	6	13	34	21	68,9	77,8	61,8
52	40	39	120	81	67,2	66,9	67,5

Tab: 4 Příklad výsledku testování jedné sady na 6 testovacích nahrávkách (test.seg1.par01)

Tabulka obsahuje jednotlivé úspěšnosti rozpoznávání změn řečníka konkrétních nahrávek 1. sady testovacích dat *seg1* získaných na základě parametrizace *par01.cfg*. Pokud budeme jako určující úspěšnost uvažovat míru *F-rate*, je patrné, že se úspěšnost pohybuje okolo 70%. Jen v případě druhé nahrávky je úspěšnost pouhých 38,7%, což je zapříčiněno velkou nepřesností odhadu, tedy vysokým počtem delecí - *D* vůči správně nalezeným dvojícím – *H*.

V globálním měřítku již lze ale s jistotou hovořit o celkové úspěšnosti metody binárního dělení. V další tabulce můžeme vidět, že se úspěšnost metody aplikované na rozpoznávání změn řečníka v telefonních záznamech pohybuje od 59,2% do 71,5%. Je tedy patrné, že je velice důležité dbát na vhodně zvolené příznaky při trénování a testování metody. Nejlepších výsledků bylo dosaženo při parametrizaci s nastavením

konfiguračním souborem *par12.cfg*. Jedná se o MFCC kepstrální příznaky se 24 použitými trojúhelníkovými filtry, počet kepstrálních příznaků je 8.

Druh parametrizace	trénování		testování					
	ØKopt	ØF [%]	ØF [%]	s_x	ØP [%]	s_x	ØR [%]	s_x
par01	52,00	70,1	70,1	9,02	70,8	10,78	71,0	11,68
par02	52,11	70,0	69,6	9,31	70,3	11,18	70,8	12,20
par03	52,44	70,0	69,7	9,26	71,2	10,92	69,9	12,09
par04	52,11	70,0	70,0	9,04	70,8	10,64	70,9	12,09
par05	49,00	60,8	59,2	10,06	62,3	13,08	58,8	12,97
par06	57,00	70,5	71,2	9,55	70,3	11,00	73,6	11,98
par07	57,00	69,2	70,2	9,34	70,5	10,41	71,2	12,26
par08	62,00	70,0	70,2	8,84	71,4	10,48	70,7	12,07
par09	48,00	70,9	71,2	8,88	71,1	11,13	73,1	11,75
par10	44,00	71,5	71,4	8,74	70,4	11,22	74,3	11,30
par11	40,78	71,7	71,4	8,27	71,2	10,80	73,4	11,39
par12	36,44	72,2	71,5	8,08	68,7	10,88	76,2	10,59
par13	33,00	71,8	71,2	8,80	69,6	10,59	75,3	12,72
par16	49,89	67,7	66,7	7,23	65,2	11,17	70,7	11,57
par17	50,00	68,1	67,3	8,35	65,9	12,15	70,7	10,64
par18	51,00	67,7	66,9	8,33	67,5	11,78	68,6	11,98
par19	46,22	68,3	67,5	7,41	64,4	10,36	73,1	11,10
par20	42,33	69,3	68,7	8,03	64,2	9,70	75,7	11,62
par21	46,33	69,1	68,7	8,17	67,5	10,19	72,1	12,65
par22	43,00	69,9	70,1	7,89	70,2	10,76	71,8	11,49
par23	39,00	70,8	71,1	7,43	69,3	10,06	74,7	11,33
par24	36,11	70,5	69,0	8,61	69,1	9,76	70,6	12,91
par25	32,00	70,8	70,4	9,18	69,4	9,04	73,3	13,34

Tab: 5 Úspěšnost testování pro dané parametrizace

3.2.2 Porovnání úspěšnosti testování na databázích ART a COST 278

Z předchozích výsledků již lze soudit, jak je námi zvolená metoda úspěšná při rozpoznávání změn řečníka na záznamech telefonních hovorů. Může se zdát, že úspěšnost rozpoznání je natolik nízká, že je tato metoda v praxi nepoužitelná. Důvodem takto nízkých hodnot míry F-rate jsou samotné nahrávky vstupující do parametrizace. Jelikož se jedná o nahrávky reálných hovorů, je kvalita těchto akustických signálů ovlivněna šumem a okolním rušením. Významnou měrou je kvalita ovlivněna i omezeným frekvenčním pásmem telefonní linky. Nejvýrazněji však úspěšnost klesá, pokud je v nahrávce příliš míst, kdy nemluví žádný z mluvčích, nebo naopak hovoří-li oba současně. Jak bylo již na krátkých ukázkách demonstrováno, metoda měla největší problémy rozpoznat právě tato místa.

Metoda binárního dělení již byla dříve testována na databázích záznamů televizních a rozhlasových pořadů a to jak v cizích jazycích, tak i pro češtinu [8]. Databáze nahrávek v českém jazyce – databáze ART, byla pořízena z 5346 segmentů pocházejících od 456 mluvčích. Jednotlivé segmenty byly náhodně pospojovány do 100 trénovacích a 100 testovacích úseků, každý o délce přibližně 10 minut. Jelikož tyto nahrávky také obsahovaly rušivé zvuky z okolí a jiné různé neřečové signály, byla vytvořena databáze S-ART, kde byly tyto rušivé zvuky v maximální míře potlačeny. Poslední testovanou databází nahrávek byla databáze COST 278 [9], která vznikla na základě projektu Evropské unie, jehož se účastnilo 10institucí z 9 různých zemí. Každá instituce poskytla 3 hodiny záznamů národních televizních zpráv. Pro srovnání s výsledky testování na základě těchto databází použijeme výsledky testování a trénování při parametrizaci s osmi MFCC kepstrálními příznaky (*par12.cfg*) databáze telefonních nahrávek, kdy byly dosažené výsledky nejlepší. Databázi pro jednoduchost označíme zkratkou THSZ07.

Databáze	trénování		testování		
	Kopt	F[%]	F[%]	R[%]	P[%]
ART	76	92,84	93,36	93,25	93,48
S-ART	72,5	96,51	96,27	95,73	96,82
	∅Kopt	**∅F[%]**	**∅F[%]**	**∅R[%]**	**∅P[%]**
COST 278	90,95	74,16	70,74	74,53	68,62
THSZ07	36,44	72,20	71,50	68,70	76,20

Tab: 6 Srovnání výsledků testování metody na různých databázích

Z tabulky je možno usoudit, že úspěšnost testování nahrávek databází ART a S-ART je v porovnání s úspěšností databáze telefonních záznamů THSZ07 nesrovnatelná. U ideálních nahrávek (uměle sestříhaných s odstraněnými "tichými" místy) databáze S-ART je úspěšnost testování F=96%, u nahrávek zatížených rušivými signály databáze ART F=93%. Hodnoty míry F-rate získané testováním na databázi THSZ07 necelých ∅F=72%. Pokud ale srovnáme výsledek testování s testováním databáze COST ∅F=71%, která byla sestavena také metodou rotace trénovacích a testovacích dat, je dosaženo srovnatelných výsledků.

Závěr

Nedílnou součástí systému pro automatické přepisování zvukových nahrávek různých televizních a rádiových pořadů do textové podoby (tzv. *"Media mining systému"*) je modul starající se o automatickou transkripci nahrávek. Ten se skládá z několika dalších částí, například detektoru řeči, změny mluvčího, jeho identifikaci a nakonec samotného rozpoznání řeči. Tato práce se zabývá možností detekce změny mluvčího pomocí metody binárního dělení a jejího praktického využití při aplikaci na nahrávky reálných telefonních hovorů. Speciálně se tedy zaměřuje na detekci změn dvou mluvčích. Tato metoda byla zvolena především z důvodů jejího snadného trénování.

Metoda binárního dělení využívá principů převedení problému hledání změn mluvčího na problém hledání bodu změny obecných parametrů ve stochastickém procesu. K hledání změny využívá testování jednoduchých hypotéz. Parametry popisující akustický signál nahrávky byly získávány pomocí dvou druhů parametrizací. Parametrizací získanou na základě diskrétní Fourierovy transformace, tzv. MFCC kepstrálních příznaků a pomocí lineární prediktivní analýzy, tzv. LPC příznaků. Navíc byly navrženy a vyhodnoceny i další varianty těchto příznaků. Bylo například použito parametrizací s různým počtem kepstrálních příznaků, u MFCC příznaků byl měněn počet trojúhelníkových filtrů, v případě LPC příznaků zase řád lineárního modelu.

Jako trénovací a testovací databáze nám posloužily nahrávky reálných telefonních záznamů. Záznamy byly postupně ručně segmentovány pro potřeby pozdějšího stanovení úspěšnosti hledání změn a i k samotnému natrénování metody. Pro trénování a testování byl naprogramován detektor změn pracující na principu námi navržené metody binárního dělení.

Výsledkem trénování je soubor dat vypočítaných změn mluvčích pro námi zvolený rozsah volného parametru a jednotlivé druhy parametrizace. Na základě těchto dat je vypočtena míra úspěšnosti detekce změny. Maximalizací této míry dostáváme hodnotu volného parametru detektoru pro jeho následné testování.

Nejlepších výsledků testování bylo dosaženo při použití parametrizace s osmi melfrekvenčními kepstrálními příznaky. Na databázi nahrávek reálných telefonních hovorů (THSZ07) zatížených různými typy rušení dosahoval detektor úspěšnosti F=72% při nastavení volného parametru K=36. Pro lepší představu o dosažených výsledcích byly uvedeny hodnoty úspěšností rozpoznávání na dalších databázích. Na uměle vytvořené databázi ideálních nahrávek televizních a rozhlasových pořadů (S-ART) dosahovala úspěšnost metody až F=96% při K=73, v případě nahrávek blížící se reálným datům (ART) F=93% při K=76. Srovnatelných výsledků dosahujeme při testování na mezinárodní databázi televizního zpravodajství COST F=71% při K=91.

Výsledky testování detekce změn mluvčího na "příliš" reálných datech, skládajících se nejen z užitečného signálu, ale i z velkého počtu rušivých složek, nedopadly, co se týče úspěšnosti definované mírou F-rate, nejlépe. Na nahrávkách upravených záznamů již ale dosahovaly dostatečně přesvědčivých výsledků. Jednou z možností zvýšení rozpoznávacího skóre je tedy vhodná úprava nahrávek před samotnou detekcí. Druhou možností je smíření se určitým typem chyb. Je možné totiž detektor změn natrénovat tak, aby nalezl větší procento správně nalezených změn vůči všem skutečným změnám za cenu chybného označení místa, kde ve skutečnosti ke změně nedošlo. Toto chybné označení je však přijatelné, oproti chybě, která by vznikla neoznačením skutečné změny.

Literatura

[1] Chen, J., Gupta, A.K.: Parametric Statistical Change Point Analysis, Birkauser, Boston, 2000, ISBN 0-8176-4169-6.

[2] Lauro, C., Antoch, J., Vinzi, V.E., Saporta G., Multivariate Total Quality Control, Physical-Verlag, Heidelberg, 2002, ISBN 3-7908-1383-4.

[3] Lehmann, E.L., Testing Statistical Hypotheses, 2nd edition, Wiley & Sons, New York, 1986, ISBN 0-412-05321-7.

[4] editor Nouza J., Počítačové zpracování řeči – cíle, problémy, metody a aplikace.TUL, Liberec, 2001, ISBN 80-7083-551-6

[5] Psutka J., Komunikace s počítačem mluvenou řečí. Academia, Praha, 1995, ISBN 80-200-0203-0

[6] Psutka J., Mluvíme s počítačem česky. Praha, 2006, ISBN 80-200-1309-1

[7] Rashmi Gangadharaiah, B. Narayanaswamy, N. Balakrishnan, A Novel Mothod for Two-Speaker Segmentation, Proc. of Intl. Conf. on Speech and Language, (ICSLP), Jeju, Korea.,September 2004.

[8] Žďánský J., Metody detekce změny mluvčího v akustickém signálu. disertační práce, Liberec, 2005

[9] Žilbert J., Mihelič F., Martens J.P., Meinedo H., Neto J., Docio L., Garcia-Mateo C., David P., Zdansky J., Pleva M., Cizmar A., Žgang A., Kačič Z., Teleki C., Vicsi K., The Cost 278 Broadcast News Segmentation and Speaker Clustering Evaluation – Overview, Methodology, Systems, Results. In Proceedings of 9–th International Conference on Speech Communications and Technology Interspeech 2005, pp. 629-632, Lisboa (Portugal), 2005

[10] Transcriber a tool for segmenting, labeling and transcribing speech, URL: <http://trans.sourceforge.net/en/presentation.php>

[11] MATLAB Help Desk, URL: <http://ccs.ucsd.edu/matlab/>

[12] Kolář, Jazyk Perl, URL: <http://www.kai.vslib.cz/~kolar/perl/>

Příloha A

Funkce převodu XML dat do proměnné – trslab2hid.m:

```
function trslab2hid(filename);

close all
clc
i=1;

infilename=['transkr',filename(10:length(filename)-6),'.trs'];
infilename2=[filename,'.lab'];
outfilename=[infilename2(1:length(infilename2)-4),'.hid'];

fid = fopen(infilename,'r');

while feof(fid)==0
   line=fgetl(fid);
   if length(line)>12 & strcmp(line(1:12),'<Sync time="')
      tsi(i,:) = sscanf(line(13:length(line)-3),'%f') ;
      i=i+1;
   end
end

fclose(fid);
i=1;
fid = fopen(infilename2,'r');
fid2 = fopen(outfilename,'w');

while feof(fid)==0
    K=fscanf(fid,'K= %d:\t');
    trj=fscanf(fid,'\t%f');
    [I,D,N,H]=near_neigh(tsi,trj);
    fprintf(fid2,'K= %d:\tI=%d\tD=%d\tN=%d\tH=%d\r\n',K,I,D,N,H);
end
fclose('all');
```