A Cepstral Domain Maximum Likelihood Beamformer for Speech Recognition

Dominik Raub, John McDonough, Matthias Wölfel

Institut für Logik, Komplexität und Deduktionssysteme
Universität Karlsruhe (TH), Germany
{ draub, jmcd, wolfel }@ira.uka.de

Abstract

Recent work by Seltzer [1] indicates that classical approaches to beamforming, minimizing output power while enforcing a distortionless constraint, do not yield optimal results in terms of *word error rate* (WER) on speech recognition task. This problem can be traced back to the mismatch between the target criterion of classical adaptive beamformers, which is optimization of the signal to noise ratio, and the actual target criterion, which is the reduction of the recognizer's WER. Following an approach by Seltzer [1] we therefore investigate the performance of an alternative error criterion, which attempts to optimize the beamformer weights, so as to improve the likelihoods along the recognizer's Viterbi path for each utterance. This criterion matches the goal of lower WERs more closely and therefore leads to better recognition results.

1. Introduction

While *automatic speech recognition* (ASR) systems perform well on recordings made in a quiet environment using a close talking microphone, distant microphone scenarios, such as meeting room tasks, still present a considerable challenge [2]. A proven method to enhance recognition results in such a scenario is the use of microphone arrays, employing delay and sum beamformers. More complicated adaptive beamforming techniques may yield improvements in *signal to noise ratio* (SNR), but such techniques will usually not improve the *word error rate* (WER) significantly [1]. We believe that this is due to a mismatch between the target criterion of classical adaptive *minimum variance distortionless response* (MVDR) beamformers and the target criterion in speech recognition, which is a low WER.

Recent work by Seltzer [1] indicates that recognition results can be improved dramatically by adapting the beamformer weights so as to maximize the likelihood of the filtered acoustic data under the recognizers best hypothesis and thereby approximating the actual target of minimizing WER more closely. In a procedure similar to the *expectation-maximization* (EM) algorithm one determines a recognition hypothesis, that is used to optimize the beamformer weights. These are in turn used to obtain a better hypothesis.

In Seltzer's work, however, the weights are not optimized with respect to the actual acoustic models. Rather, Seltzer uses auxiliary models in the log-mel-spectral domain. As the recognizer itself uses cepstral models, cepstral auxiliary models should approximate the recognizer better and thus lead to better recognition results. Therefore, the purpose of this work is to investigate the possible advantages of using cepstral auxiliary models.

2. Beamforming

Beamforming exploits the fact that signals impinging on a microphone array will travel different distances to each of the sensors [3]. Imposing delays on the signals received by the microphones such that the arrival delays of signals originating from the *look direction* are exactly compensated, and summing the delayed signals, signals arriving from other directions will undergo destructive interference and thereby be attenuated.

2.1. Narrowband and frequency domain beamforming

If the bandwidth of a signal is small we may approximate a time domain linear filter by multiplying the frequency domain signal with a complex weight [4]. We can thus write the narrowband beamformer with the *frequency domain snapshot* u as input vector, weight vector w and output y as

$$y(t) = w^H u(t) \tag{1}$$

where $(\cdot)^H$ is the Hermitian transpose operator. Frequency domain beamformers [3] are constructed by applying an analysis filter bank, in this work Hamming window and *Fast Fourier Transform* (FFT), to each sensor output, to obtain a discrete spectrum with sufficiently narrow frequency bins. Given a set of frequency bins, one provides a separate narrowband beamformer for every frequency band i in the spectrum, that takes as input the value of band i from each sensor. The outputs of all narrowband beamformers form a full spectrum, that can be transformed back into the time domain or passed to the remaining components of an ASR frontend.

As opposed to time domain beamformers, frequency domain implementations allow for an independent optimization of each subband [4], thus exhibiting better convergence properties.

2.2. Generalized sidelobe canceller

It is often desirable that signals arriving from a specified look direction are not distorted or attenuated by the beamformer, while all other signals are attenuated as much as possible. A beamformer attaining this goal is called MVDR beamformer [3]. Using a simple narrowband filter and sum approach one has to perform a constrained optimization of the weights to minimize output power while maintaining distortionless response. The linear constraint is given by

$$w^H s = 1 \quad \text{where } s = (e^{-j\theta_0}, e^{-j\theta_1}, \ldots, e^{-j\theta_{m-1}})^T. \tag{2}$$

In the *steering vector* s the angles θ_i are the phase differences at the sensors for signals arriving from the look direction.

The general sidelobe canceler (GSC) [5] turns this constrained problem into an easier to solve unconstrained one. We define the *quiescent weight vector* $w_q = \frac{1}{m}s$ and give a *blocking matrix* $B \in \mathbb{C}^{m-1\times m}$, the columns of which span the orthogonal complement of w_q in $\mathbb{C}^m$. Let $w_a \in \mathbb{C}^{m-1}$ be the *active weight vector*, for now arbitrary. The beamformer output $y(t)$ for input $u(t)$ is given by

$$y(t) = w^H u(t) = (w_q^H - w_a^H B^H)u(t) \tag{3}$$

where $w^H = w_q^H - w_a^H B^H$ is the effective weight vector, i.e. the weight vector that would lead to the exact same response as the

GSC yields in a standard narrowband filter and sum beamformer. Note that the GSC always satisfies the linear constraint (2), independent of the choice of the active weight vector w_a. Thus we may solve the constrained MVDR optimization problem by solving the unconstrained problem of finding an active weight vector w_a for the GSC that minimizes the output power given an input $u(t)$. A more detailed derivation of these results can be found in [5].

3. Seltzer's LIMA-beam

Seltzer [1] proposes a new target criterion for the optimization of the beamformer weights w that is designed to better match the objective of reduced WERs. He observes that the recognizer chooses a hypothesis $\hat{h}$ according to

$$\hat{h} = \operatorname*{argmax}_h P(h)P(y|h). \tag{4}$$

The feature vector $y = y(u, w) = (y(t; u, w))$ is a function of the frequency snapshots u and the beamformer weights w. Let h_c be the correct transcript for the input signal $u = (u(t))$. Then the optimal set of weights $\hat{w}$ should be selected to minimize the WER or, in approximation of that goal, to maximize the probability of the features y under the hypothesis h_c:

$$\hat{w} = \operatorname*{argmax}_w P(y(u, w)|h_c) \tag{5}$$

The evaluation of Eq. (5) is central to all algorithms subsequently derived. The differences lie only in the approximations made and the beamformer setup.

To describe Seltzer's algorithm we need to discuss a number of approximations. First, let $s_c = (s_c(t))$ be the state sequence of the *hidden markov model* (HMM) associated with the transcript h_c. Then, ignoring the state transition probabilities, we have:

$$P(y(u, w)|h_c) = \prod_{t=0}^{|s_c|} P(y(t; u, w)|s_c(t)) \tag{6}$$

Generally $P(y(t)|s_c(t))$ will be a Gaussian mixture whose likelihood is difficult to maximize. So, as an approximation, Seltzer uses auxiliary models in the logspectral domain with only a single Gaussian component instead of the recognizer's actual models. The auxiliary models are trained in the same way as the recognizer models, only that they operate on logspectral instead of cepstral features. The single Gaussian case is much easier to handle because we have

$$\begin{aligned} &\operatorname*{argmax}_w \prod_{t=0}^{|S_c|} P(y(t; u, w)|s_c(t)) \qquad (7) \\ &= \operatorname*{argmax}_w \log \prod_{t=0}^{|S_c|} \mathcal{N}(y(t; u, w); \mu_{s_c(t)}, \Sigma_{s_c(t)}) \end{aligned}$$

Thus we have as error function

$$\varepsilon = \sum_{t=0}^{|S_c|} (y(t) - \mu_{s_c(t)})^H \Sigma_{s_c(t)}^{-1} (y(t) - \mu_{s_c(t)}) \tag{8}$$

where the logspectrum $y(t) = (y_m(t; u, w))$ at time t with input sequence u and weight set w is given as:

$$y_m(t) = \log_{10}(\sum_f M_{m,f} |v_f(t)|^2) \tag{9}$$

Here $M_{m,f}$ is the Mel matrix indexed by Mel band m and frequency band f. The beamformer output $v = (v_{f_1}(t), \ldots, v_{f_n}(t))$ is then given by

$$v_f(t) = w_f^H u_f(t) \tag{10}$$

where $v_f(t)$ is the output of of the narrowband beamformer for frequency band f, calculated from the input $u_f(t)$ for that band and the appropriate weights w_f determined by $w = (w_{f_1}, \ldots, w_{f_n})$.

To optimize the beamformer weights we determine the derivative of the error function ε by w^* [5] as shown below. Given these derivatives, we can optimize the beamformer weights using gradient driven procedures [6].

$$\frac{\partial}{\partial w^*} \varepsilon = 2 \sum_{t=0}^{|s_c|} (y(t) - \mu_{s_c(t)})^H \Sigma_{s_c(t)}^{-1} \frac{\partial}{\partial w^*} y(t) \tag{11}$$

$$\frac{\partial}{\partial w_f^*} y_i(t) = \frac{1}{\sum_g M_{i,g} |v_g(t)|^2} M_{i,f} \frac{\partial}{\partial w_f^*} |v_f(t)|^2 \tag{12}$$

$$\frac{\partial}{\partial w_f^*} |v_f(t)|^2 = u_f(t) u_f(t)^H w_f = u_f(t) v_f(t)^H \tag{13}$$

For a fixed mel band k the mel matrix $M = (M_{k,f})$ gives a filter with a triangular transfer function, that has zero entries for all but a small number of frequency bands f. Thus a particular mel component of the error function ε, that is defined in the log mel domain, depends only on a fairly small number of frequency components. This is the main reason Seltzer chose to work in the log mel domain [1]. Of course the mel filters overlap, so the bands are not independent. Thus Seltzer proposes to duplicate the weights, i.e. to keep a separate set of weights for each mel band and fequency band. Using this approach one may optimize the weights under each mel triangle independently. This is a solution to the complexity and convergence problems one faces when trying to optimize all filter weights jointly.

The speech recognizer demands that the signal be stationary during an observation window, thus constraining the window length to about 20ms. On the other hand a filter with a finite duration impulse response of only 20ms cannot compensate for reverberation effects, as typical room impulse responses extend over 150ms [1]. As solution Seltzer proposes to replace the subband weights by short tap-delay-line linear filters. Thus we obtain a longer subband filter impulse response, while preserving the ability to directly feed the beamformer output into the recognizer.

3.1. ML beamforming

The evaluation of Eq. (5), central to all *maximum likelihood* (ML) beamformers, requires knowledge of the correct transcript h_c.

In *semi-supervised ML beamforming* [1] we use a calibration utterance with known transcription h_c to optimize the beamformer weights. The calibration utterance is then read by the user and recorded with the microphone array. We initialize a beamformer to delay and sum and use it to produce features y. From the features y and the transcription h_c we then calculate error function and gradient according to Section 3 or 4. These we use to optimize the beamformer weights, employing a gradient descent procedure. Once our weights have converged, we use this calibrated beamformer to decode all subsequent utterances.

The *unsupervised ML beamformer* [1] requires no calibration utterance, rather it employs the EM algorithm. Delay and sum is used to obtain a preliminary set of features. The recognizer determines a Viterbi state sequence and transcript for the given features (E-step). From this state sequence one calculates likelihoods and gradients as in Section 3 or 4, that are used to minimize the negative log-likelihood under these models (M-step). This procedure is iterated until the weights have converged.

4. CDML beamforming

To improve recognition performance, we close the gap between optimization criterion and actual optimization target (minimal WER) further by following Seltzer's approach [1], but replacing the single gaussian log-spectral domain auxiliary models with single gaussian cepstral domain auxiliary models. Since each cepstral coefficient depends on all frequency bands, duplication of weights and independent optimization as in [1] is no longer feasible. Hence, in *cepstral domain maximum likelihood* (CDML) beamforming convergence may become more of an issue, but the optimization target ought to be matched more closely.

Assume a frequency domain snapshot sequence u and a matching transcript h_c are given. As error function for CDML beamforming we then use the negative log-likelihood of the beamformer output $v = v(w, u)$ under the speech recognizer's model associated with the maximum likelihood state sequence s_c for the transcript h_c. We have $s_c(t) = (\mu(t), \Sigma^{-1}(t))$ where $\mu(t)$ is the mean vector for the state $s_c(t)$ at frame t and $\Sigma^{-1}(t)$ the inverse covariance matrix. We still get (8) as error performance surface, however the output of the recognizer frontend $y(t) = (y_i(t))$ now is (compare Eq. (9)):

$$y_i(t) = \sum_m D_{i,m} \log_{10}(\sum_f M_{m,f}|v_f(t)|^2) \tag{14}$$

$D_{i,m}$ is the DCT matrix, indexed by DCT band i and Mel band m. We may now calculate the derivative $\frac{\partial}{\partial w^*}\varepsilon$ of the error function ε by the weights w. Let w_f denote the weights acting on frequency bin f:

$$\frac{\partial}{\partial w^*}\varepsilon = 2\sum_{t=0}^{|s_c|}(y(t) - \mu_{s_c(t)})^H \Sigma^{-1}_{s_c(t)} \frac{\partial}{\partial w^*} y(t) \tag{15}$$

$$\frac{\partial}{\partial w_f^*} y_i = \sum_m \frac{D_{i,m}}{\sum_g M_{m,g}|v_g|^2} M_{m,f} \frac{\partial}{\partial w_f^*}|v_f|^2 \tag{16}$$

Subsequently we give gradient equations for a number of beamformer setups, that can be used in the algorithms of section 3.1.

4.1. Unconstrained Filter and Sum

A simple narrowband filter and sum beamformer is given by Eq. (10). To provide for a longer finite duration impulse response, as motivated in Section 3, we now describe a unconstrained filter and sum beamformer with N taps per frequency band f and channel c:

$$v_f(t) = \sum_{n=0}^{N-1} w_{f,n}^H u_f(t-n) \tag{17}$$

For $N = 1$ this still describes the simple filter and sum beamformer from Eq. (10). Eq. (17) yields the following gradient expression:

$$\frac{\partial}{\partial w_{f,n}^*}|v_f(t)|^2 = u_f(t-n)v_f(t)^H \tag{18}$$

where we define $\frac{\partial}{\partial w_{f,n}^*}|v_f|^2$ to be a column vector of partial derivatives each position corresponding to one channel. Together with Eq. (15) and (16) this describes the gradient.

4.2. GSC-CDML Beamformer

The GSC can be used to enforce a set of linear contraints while optimizing the target function (8). This could be useful to incorporate external information, such as visually obtained speaker location information. If we replace the freely adjustable beamformer with a GSC in the system described above, we can prescribe a look direction and adapt the remaining degrees of freedom as before.

As describen in section 2.2, the GSC makes use of a quiescent weight vector w_q and an active weight vector $w_a = w_{a,n,f,\tilde{c}}$. Once more we provide for a longer finite duration impulse response by replacing a simple weight by a tap-delay line. The GSC Beamformer is then given as:

$$v_f(t) = w_{q,f}^H u_f(t) - \sum_{n=0}^{N-1} w_{a,n,f}^H B_f^H u_f(t-n) \tag{19}$$

The gradient equations with respect to the active weight vector are:

$$\frac{\partial}{\partial w_{a,n,f}^*}|v_f(t)|^2 = -B_f^H u_f(t-n) v_f(t)^H \tag{20}$$

Subtituting into Eq. (15) and (16) we obtain the full gradient expression.

Note that the GSC enforces a distortionless contraint on signals arriving from the look direction. This may lead to degraded performance, if the speech signal is distorted in some fashion or if there are multipath components arriving from near the look direction. The GSC is also sensitive to erroneous steering vectors, whereas the unconstrained adaptive algorithm finds its direction automatically.

4.3. Modified GSC-CDML Beamformer

We use multiple taps in order to compensate for effects a beamformer with short impulse response can not address, in particular reverberation. To do so, we need to cancel part of the signal with time delayed samples. It may therefore be sensible not to subject the time delayed frequency samples to the linear constraints (i.e the blocking matrix), since it suppresses all signal components incident from the look direction, preventing them from cancelling the signal. Hence we construct the following modified tap delay line GSC:

$$v_f(t) = (w_{q,f}^H - w_{a,f}^H B_f^H) u_f(t) + \sum_{n=1}^{N-1} w_{e,n,f}^H u_f(t-n) \tag{21}$$

For the active weight vector w_a we find:

$$\frac{\partial}{\partial w_{a,f}^*}|v_f(t)|^2 = -B_f^H u_f(t) v_f(t)^H \tag{22}$$

For the extended weight vector w_e we obtain:

$$\frac{\partial}{\partial w_{e,n,f}^*}|v_f(t)|^2 = u_f(t-n) v_f(t)^H \tag{23}$$

Again we subtitute into Eq. (15) and (16) to obtain the full gradient expression.

5. Experiments and results

The experiments in this section were conducted with the Janus Recognition Toolkit (JRTk) developed jointly at the Universität Karlsruhe (TH), Germany, and the Carnegie Mellon University Pittsburgh, USA. The recognizer and the auxiliary models used for beamforming were trained on the English Spontaneous Scheduling Task (ESST) corpus, which is comprised of approximately 35 hours of dialog contributed by 242 speakers. The data was collected recording the planning of an overseas business trip with Sennheiser head-mounted close-talking microphones.

The ESST test set, on which all results quoted here were obtained, consists of 1,825 utterances by 16 unique speakers, 210 minutes total, 22,889 words. For our beamforming experiments we used an 8 element linear array with 41 mm inter-element spacing. The data was replayed over a stationary speaker positioned 2 meters from one end of the array, measured perpendicularly to the array. Recording took place in our reasonably quiet, but by no means soundproofed or damped seminar room. All data was sampled at a rate of 16kHz.

	Interference	Spec.	Ceps.	WER
Single Channel	none			61.37%
	music 12.72dB SNR			65.94%
	talking 9.03dB SNR			64.36%
Delay & Sum	none	59.47	33.81	51.34%
	music 12.72dB SNR	56.33	35.20	59.95%
	talking 9.03dB SNR	58.35	34.83	59.39%
LIMA BEAM	none	53.31	40.16	58.76%
	music 12.72dB SNR	55.26	40.20	63.62%
	talking 9.03dB SNR	54.53	40.26	62.24%
CDML	none	60.87	33.46	51.00%
	music 12.72dB SNR	59.15	33.67	58.05%
	talking 9.03dB SNR	60.04	33.97	57.10%
GSC-CDML	none	58.26	33.07	54.65%
	music 12.72dB SNR	55.54	33.62	58.93%
	talking 9.03dB SNR	56.81	33.30	58.04%

Table 1: *WERs and per frame negative log-likelihoods under cepstral and log-spectral auxiliary models*

For the experiments with interference, the interfering source was recorded separately but in the same room with the array in the same position, but the interference source positioned 3m (measured parallel to the array) from where the speech source was placed. We used speech from another talker and music from a chamber orchestra as interfering signals. For the experiments, the interference was added to the "interference free" signal before processing.

Features were calculated every 10ms, using a 20ms sliding Hamming window. The windowed data was padded with zeros and passed through a 512 point FFT, acting as filter bank. The frequency samples were then passed to the array processor or, for the single microphone experiments, transformed into 13 cepstral coefficients in the usual fashion. Recognition was performed with static and dynamic features, without power features, using cepstral domain mixture models and cepstral mean subtraction. Also listed are the average per frame negative log-likelihoods (i.e. lower is better) of the beamformer outputs, both under the single Gaussian log-spectral domain auxiliary models we used in our implementation of Seltzer's LIMA BEAM [1] and under the single Gaussian cepstral domain auxiliary models used in our CDML beamformers.

Comparing the recognition results on a single channel of the beamformer (see Table 1) to a 31.94% WER on clean close talking data we see the heavy degradation switching from close talk to distant microphone. Furthermore Table 1 shows the baseline delay and sum results and the results for the CDML Beamformer and Seltzer's LIMA BEAM operating in a semi-supervised fashion. All experiments were conducted without multiple frequency domain taps. One utterance per speaker of at least 10s length was used for calibration. The remaining utterances for the speaker were then processed using the weights estimated during calibration. Optimization was performed using the gradient expressions derived above and employing a Pollak-Ribiere conjugate gradient minimizer [6].

We find that improvements and degredations in cepstral domain likelihood correllate well with improvements and degredations in WER. Actually there is only a single deviation from this pattern, for the no interference case on the GSC-CDML beamformer. There the optimization appears to get caught in a local optimum, leading to anomalous behavior. We were unfortunately unable to establish a reliable connection between log-spectral domain likelihoods and WER in our experiments. In particular we failed to get any impovements from Seltzer's LIMA BEAM on our system and data, although the log-spectral domain likelihood numbers clearly indicate that the optimization is converging.

6. Discussion and further work

We find that CDML beamforming yields decent improvements (up to 2.29% absolute) in presence of interference. Since the GSC-CDML beamformer, which does not act as a postfilter, is only .94% absolute worse, we believe that at most .94% absolute of the 2.29% are due to the postfilter effect of the CDML algorithm.

Note that the central problem in distant microphone speech recognition is usually reverberation. The reverberation lies in the same frequency range as the target signal and is generally prone to "confuse" the recognizer. Unfortunately typical room impulse responses are about 150ms long, about 10 times as long as the filters employed in our single tap algorithms. Seltzer [1] was able to demonstrate considerable improvements on reverberated data using a tap-delay line approach in the frequency domain. We reformulated that approach for the cepstral domain in section 4. Further experiments on this approach appear promising.

Seltzer [1] quotes improvements using single tap beamformers in the log-spectral domain. We were unfortunately unable to reproduce these, probably due to the different data and recognizer.

We employed an FFT filter bank in our experiments, because it is simple, fast and already part of the recognizer frontend. However, there is a comparatively large overlap between the FFT bins [4]. Therefore the assumption of independence of different frequency bins may be violated with observable impact on performance. We expect that employing a perfect reconstruction filterbank with better separation properties could boost performace.

Finally it would be desireable to have a beamformer with tracking capability. The semi-supervised beamformers are calibrated only once and do not support tracking. The unsupervised approach allows for crude adaptation, by optimizing for each new utterance separately, using the weights obtained for the last utterance as initialization. This is however only viable if environmental conditions change slowly. To cope with more rapidly changes, one has to use an adaptation algorithm like LMS, RMS or a Kalman filter. We conducted some preliminary experiments with LMS beamformers, but convergence appeared to be too slow to yield sufficient improvements. Experiments described in [7] using RLS and Kalman Filters gave more promising results, but further work is still necessary.

This work was funded by the European Union as part of the CHIL Project, grant number IST-506909.

7. References

[1] Seltzer, M. L., "Microphone Array Processing for Robust Speech Recognition", PhD Thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2003.

[2] Yu, H., Tomokiyo, T., Wang, Z. and Waibel, A., "New Developments in Automatic Meeting Transcription", ICSLP, Beijing, China, October 2000.

[3] van Trees, H. L., Optimum Array Processing (Detection, Estimation, and Modulation Theory, Part IV), John Wiley & Sons Inc., New York, NY, USA, 2002.

[4] Shynk, J. J., "Frequency-Domain and Multirate Adaptive Filtering", IEEE Signal Processing Magazine; Jan. 1992.

[5] Haykin, S., Adaptive Filter Theory, Prentice Hall, Upper Saddle River, NJ, USA, 2002.

[6] Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P., Numerical Recipes in C: The Art of Scientific Computing, 2nd ed., Cambridge University Press, Cambridge, 1992.

[7] McDonough, J., Raub, D., Wölfel, M. and Waibel, A., "Towards Adaptive Hidden Markov Model Beamformers", submitted to *IEEE Trans. on Speech and Audio Processing*; 2004.