BLIND DEREVERBERATION OF SINUSOID SIGNALS USING PLL-BASED COMBINED PHASE AND AMPLITUDE ANALYSIS

Ralf Huber ⟨*ralf.huber@student.kit.edu*⟩
Florian Kraft ⟨*florian.kraft@kit.edu*⟩
Alex Waibel ⟨*ahw@cs.cmu.edu*⟩

Karlsruhe Institute of Technology
Department of Computer Science, Institute of Anthropomatics
Adenauerring 2, 76131 Karlsruhe

ABSTRACT

In this paper a new 'blind' single-microphone method for the dereverberation of sinusoid signals is presented. A phase-locked-loop is utilized to precisely track the amplitude, frequency and phase offset of a reverberated recording. This information can then be combined to calculate the amplitude and phase offset of single reverberated wavefronts, which allows to subtract them from the original recording. Experimental results have shown that the direct-to-reverberant ratio of recordings can be improved to an extent equal to a delay-and-sum beamformer with 5 microphones. At the end, extensions are outlined which might make the method suitable for dereverberation of real speech.

***Index Terms*—** Dereverberation, Phase-Locked-Loop, PLL, Echo, Robust Speech Recognition

1. INTRODUCTION

A common front-end for automatic speech recognition uses Mel-scale cepstral coefficients, which are based on Fourier transforms, to obtain a spectral representation of sound. The drawback of discrete Fourier transforms is that their output is subject to a trade-off between frequency resolution and time-domain resolution. If better frequency resolution is needed, it can only be achieved by decreasing time-domain resolution and vice versa.

Phase-locked-loops (PLLs) are another method of finding and tracking the frequency which is present in a signal and they have been used for example in FM demodulators or GPS receivers for many years [1]. Publications by Pelle, et al. have shown that PLLs can also be used in the context of automatic speech recognition, for instance for pitch-tracking [2] or as ASR frontends [3] [4].

In this paper the PLL's good frequency tracking quality and fast tracking speed are used as a foundation for a new dereverberation method for single sinusoids of constant frequency. Extensions to the system are described in section 5, which should allow to use it for real speech signals in the future.

Existing dereverberation algorithms include for example beamforming [5], multi-pass inverse filtering [6] or multi-step linear prediction (MSLP) [7]. The disadvantage of beamformers is that they require multiple microphones, inverse filters need to be 'trained' on a sufficient amount of data before they can be used for dereverberation and the MSLP-method presented in [7] is only effective for late reflections.

The goal for this work was to achieve single-microphone 'blind' dereverberation, which means that no knowledge about the room or the location of speaker and microphone should be needed. Additionally, no training should be required for it to work. The solution to the problem could be found by using both the amplitude- and phase-information present in a recording, in contrast to systems which calculate a power spectrum and thus effectively discard the phase information.

The rest of this document is organized as follows: The basic dereverberation principle is derived in the next chapter (2), whereas the PLL and the rest of the algorithm are described more detailed in section 3 and experimental results are presented in section 4.

2. INTERFERENCE OF SOUND WAVES

After an utterance is made, the direct sound arrives at the microphone first, followed by the reflected wavefronts. When there is no direct path from the speaker to the microphone, the sound that has travelled the shortest distance will be referred to as the "direct" sound from now on.

The frequencies of the direct sound and the reflected wavefronts are equal because they originate from the same source. However, due to the different distance that each of the wavefronts has travelled, their amplitudes and phase offsets differ when they arrive at the microphone.

When wavefront W_n arrives, it interferes with S_{n-1}, which is the sum of all the wavefronts that have already arrived (W_0 being the direct sound), as expressed by the following equation:

$$S_n(t) = \overbrace{W_0(t) + \underbrace{W_1(t) + \cdots + W_{n-1}(t)}_{S_{n-1}(t)} + W_n(t)}^{R_n(t)} \quad (1)$$

A linear combination of sinusoids with the same frequency is again a sinusoid with that frequency, but different amplitude and phase offset. Let a_{W_0}, a_{R_n}, ϕ_{W_0} and ϕ_{R_n} denote the amplitudes and phase offsets of the direct sound wave W_0 and the sum of reflections R_n, respectively, then equation 1 can also be written in complex exponential form:

$$S_n(t) = a_{W_0} \cdot \mathrm{e}^{i(\omega t + \phi_{W_0})} + a_{R_n} \cdot \mathrm{e}^{i(\omega t + \phi_{R_n})} \quad (2)$$

Equation 2 can be split in real and imaginary parts, i.e. cos and sin,

This research is partially funded by the German Research Foundation (DFG) under Sonderforschungsbereich SFB 588 "Humanoid Robots - Learning and Cooperating Multimodal Robots", http://www.sfb588.uni-karlsruhe.de

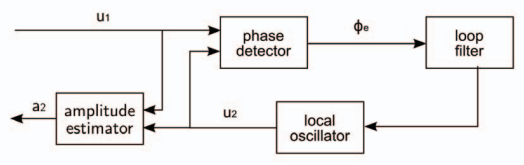

Fig. 1. block diagram of a PLL

respectively:

$$a_{S_n}\cos(\omega t+\phi_{S_n}) = a_{W_0}\cos(\omega t+\phi_{W_0}) + a_{R_n}\cos(\omega t+\phi_{R_n}) \tag{3}$$

$$a_{S_n}\sin(\omega t+\phi_{S_n}) = a_{W_0}\sin(\omega t+\phi_{W_0}) + a_{R_n}\sin(\omega t+\phi_{R_n}) \tag{4}$$

The microphone can of course only pick up the resulting wave S_n. However it is possible to measure the signal parameters a_{S_n}, ω, and ϕ_{S_n} after the arrival of each wavefront with a PLL, so equation 3 can be solved for a_{R_n}:

$$a_{R_n} = \frac{a_{S_n}\cos(\omega t+\phi_{S_n}) - a_{W_0}\cos(\omega t+\phi_{W_0})}{\cos(\omega t+\phi_{R_n})} \tag{5}$$

This, in turn, can be used in equation 4 to solve it for ϕ_{R_n}:

$$\phi_{R_n} = \operatorname{atan}\left(\frac{a_{S_n}\sin(\omega t+\phi_{S_n}) - a_{W_0}\sin(\omega t+\phi_{W_0})}{a_{S_n}\cos(\omega t+\phi_{S_n}) - a_{W_0}\cos(\omega t+\phi_{W_0})}\right) - \omega t \tag{6}$$

The described method allows to calculate the amplitude and phase offset of the sum of reflected wavefronts. This information can then be used to synthesize the sum of reflections and subtract it from the signal which was picked up by the microphone in order to achieve dereverberation.

3. SYSTEM OVERVIEW

3.1. Notation

The formulas in the following section are mostly expressed using continuous variables denoted with the time-variable in parenthesis, e.g. $S_n(t)$. However, the actual system was implemented as a digital system in MATLAB/SIMULINK. In this paper, only those formulas that rely on the usage of discrete samples, for instance the calculation of a median, will be expressed using a sample index in brackets, e.g. $a_2[k]$. All other equations will be expressed using continuous functions, irrespective of the actual implementation.

3.2. Phase-locked loop

It is necessary to track the microphone signals' frequency, amplitude and phase offset very precisely for the suggested method. Short-time fourier transforms are not precise enough because in order to achieve a good resolution in the frequency domain, it would be necessary to calculate the FFT of many samples, resulting in bad temporal resolution and vice versa.

A phase-locked loop (PLL), however, features sufficient tracking accuracy. PLLs are control circuits which match the output of a local oscillator ($u_2(t) = \sin(2\pi f_2(t)t + \phi_2(t))$) to a sinusoidal target ($u_1(t) = a_1(t)\cdot\sin(2\pi f_1(t)t + \phi_1(t))$). Figure 1 depicts the basic structure of a PLL as it was implemented in SIMULINK.

ϕ_1 and ϕ_2 are first compared in the phase detector. A multiplier would be the most simple form of a phase detector, but an additional gain control would be needed so it can work for different values of a_1 [8, p. 19]. To avoid this problem, the Hilbert-Transform phase detector [1, p. 270] has been used in this work. Mathematically, its output $\phi_e(t)$ exactly equals $\phi_1(t) - \phi_2(t)$ if $f_1(t) = f_2(t)$, no matter the amplitude of the signals it compares. The drawback of the Hilbert-Transform phase detector is that it needs Hilbert-filtered copies of its inputs, i.e. u_1 and u_2 with all frequencies shifted by -90° each. For the SIMULINK implementation the Hilbert-transform was approximated using a 1200-tap finite impulse response (FIR) filter.

The phase error ϕ_e is then filtered in the loop filter, which was realized using a proportional-integral (PI) controller, and after that, the output of the loop filter is fed into a voltage-controlled oscillator (VCO). The VCO increases or decreases f_2 according to its input until ϕ_2 and f_2 finally match ϕ_1 and f_1, respectively. The PLL is said to be "locked" when $f_1(t) = f_2(t)$ and $\phi_1(t) = \phi_2(t)$.

In a "classic" PLL the amplitudes of both signals are not matched, so u_1 and u_2 are only coherent, but not equal. For the proposed method it is however necessary to estimate the amplitude a_1 of the target signal. Karimi-Ghartemani and Iravani proposed a so-called "Extended PLL" in [9] which features a simultaneous frequency, phase and amplitude locking. The amplitude estimator in figure 1 is an implementation of their circuit, albeit only the part which is responsible for the amplitude estimation has been used.

3.3. Dereverberation algorithm

The following steps have been performed to achieve dereverberation:

1. At first the recorded signal is processed using the PLL. After the first run, $f_2(t)$ and $a_2(t)$ are known, which are estimates of $f_1(t)$ and $a_1(t)$, respectively.
2. $a_2(t)$ is then filtered using a FIR differentiator filter and the derivative is searched for its maximum and minimum. The indices of these two samples correspond to the increasing amplitude at the beginning and the decreasing amplitude at the end of the direct sound within a recording. i_S denotes the sample index at the start of the direct sound and i_E refers to the sample at the end of the direct sound.
3. The first PLL run can only be used to get estimates of $f_1(t)$ and $a_1(t)$. An estimate of $\phi_1(t)$ can't be obtained because the PLL constantly tries to eliminate the phase error ϕ_e. A solution to this problem is a second PLL run where the VCO frequency is set to $f_1(t)$. Since $f_1(t)$ is unknown in a real environment (but it is constant during our experiments), it has to be approximated by $\overline{f_2}$, the mean of $f_2(t)$ from the first PLL run:

$$\overline{f_2} = \frac{1}{i_\mathrm{E} - i_\mathrm{S} + 1}\sum_{k=i_\mathrm{S}}^{i_\mathrm{E}} f_2[k] \tag{7}$$

   To keep the oscillator frequency constant, the "wire" between the loop filter and the VCO is imaginarily cut in the SIMULINK model for the second PLL run. In this modified model the VCO generates $u_2(t) = \sin(2\pi\overline{f_2}t)$, i.e. $\phi_2(t) = 0$ for all t. Because of that, $\phi_e(t) = \phi_1(t) - \phi_2(t) = \phi_1(t) - 0 = \phi_1(t)$, so the output of the phase detector can be used as an estimate of $\phi_1(t)$.
4. The beginning of the direct sound has been detected in step 2 and for a small amount of samples immediately after i_S, the direct sound wavefront W_0 is the only one present at the microphone. The recording setup used in our experiments ensured that the first reflected wavefront (from the floor) arrived 2-3ms after the direct sound. The amplitude a_{W_0} of the direct sound can therefore be

safely identified by calculating the median of $a_2(t)$ of the first 49 samples ($\widehat{=}$ 1.021ms @ 48kHz sampling rate) immediately following i_S. ϕ_{W_0} can be derived similarly from $\phi_2(t)$.

5. In step 5 equations 5 and 6 are used to calculate $a_R(t)$ and $\phi_R(t)$ (i.e. the amplitude and phase offset of the sum of reflected wavefronts) for all samples after i_S.

6. When wavefront W_n arrives at the microphone at time t_n, it results in a change of $a_R(t)$ and $\phi_R(t)$, as described in section 2. Because of this, arriving wavefronts can be detected by searching for extrema of the derivatives $\frac{\mathrm{d}a_R(t)}{\mathrm{d}t}$ and $\frac{\mathrm{d}\phi_R(t)}{\mathrm{d}t}$. Depending on the phase and amplitude relation of R_{n-1} and W_n, it might happen that there is a discontinuity of either only a_R or only ϕ_R or both at time t_n.

   A FIR differentiator filter has been used to calculate both of the derivatives, which then have to be normalized and smoothed before peaks can successfully be detected. The peaks can simply be found by iterating over all samples and finding those where both neighbors have smaller values. To avoid misdetection of small local maxima which arise for example from noise, detected peaks must have a minimum distance of 0.5ms. Figure 2 depicts the output of the detection of step 6 of the algorithm. The minimum

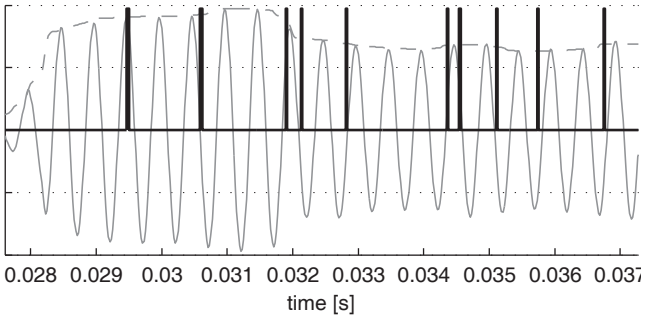

Fig. 2. 2kHz sine wave; recording distance 1.2m; grey solid line: PLL output waveform; grey dashed line: PLL amplitude estimate; black line: detected discontinuities of reverb amplitude or phase

   timespan between successive detections leads to the fact that multiple wavefronts which arrive less than 0.5ms apart cannot be detected separately. As a result, only one of them can possibly be deleted. It should be expected that this detection scheme leads to bad dereverberation performance during late reflections, because wavefronts which arrive that quickly after each other occur mainly during the late reflections. Each individual late reflection wavefront, however, has only a comparably low amplitude, because it was reflected from multiple surfaces and it had to travel a far distance in the air while loosing most of its energy. The early reflections, which contain comparatively more energy and which therefore cause stronger variations on the recorded signal, are further apart than 0.5ms and so they can be detected correctly. As the experiments will show later the proposed dereverberation method also works sufficiently during late reflections.

7. After the timestamps for all arriving reflected wavefronts have been detected, the algorithm iterates over them. For each timespan defined by two consecutive timestamps the mean reverb amplitude and phase offset is calculated, i.e. the mean of the output of step 5. After that, the appropriate reverb for that timespan is created artificially and subtracted from the recording, thus removing the reverb.

4. EXPERIMENTAL RESULTS

To test the dereverberation method, individual sine waves at varying frequencies from 150Hz to 4kHz have been played over a loudspeaker for 1s each and recorded simultaneously (including 0.5s of reverb after the end of each playback). Recordings were carried out at distances of 0.6m, 1.2m and 1.8m in an L-shaped living room (approx. $24\mathrm{m}^2$). The microphone remained in one place for all distances, while the speaker was moved accordingly. Due to a lack of a measurement microphone, reverberation times could only be estimated from the recordings by analyzing power decay after the end of the direct sound. The estimated T_{60} time is around 0.5 seconds.

The results are depicted in figure 3 and they are given as an increase of the direct-to-reverberant ratio (DRR). In similar fashion to the signal-to-noise ratio (SNR), the DRR measures the ratio of the power of the direct sound of a recording relative to its reverberant parts. The maximum possible DRR improvement theoretically achievable by a delay-and-sum beamformer (DSB) with 5 microphones was calculated in [5], which is why these figures are given as a comparison.

It can be seen that the average performance of the new method for each distance is as good as or slightly better than a 5-microphone DSB. It must be said, however, that the experiments with DSB's in [5] have shown that the theoretical maximum can (almost) be reached using real speech as an input, not only with sine waves.

Outliers can be explained by the fact that the PLL doesn't always detect the signal's amplitude, frequency and phase offset exactly. The problem is particularly severe if the frequency isn't detected correctly: If the detected frequency is slightly different from the real frequency, this will result in an error that increases constantly over time, as it was the case for the 1kHz recording at a distance of 0.6m. It is expected that errors due to incorrect frequency detection can be reduced by the system extensions described in section 5. It will never be possible to eliminate parameter misdetection fully, but improved loop filter parameters (or even completely different loop filter designs) can result in improved PLL tracking.

5. EXTENSIONS OF THE SYSTEM

5.1. Multiple frequencies

The dereverberation method presented so far is only designed to work on signals which contain a single fixed frequency. However, voiced phonemes of real speech consist of many harmonic frequencies. A bandpass filterbank could be used to split the signal into bands with a single harmonic frequency per band, which could then be tracked by one PLL each.

5.2. Time-varying amplitude

So far, all changes of amplitude within the recording were identified as arriving wavefronts. Naturally, the amplitude of a harmonic frequency within a segment of speech also changes when the speaker switches from one phoneme to another. Assume that a change of amplitude (or phase) is detected. The proposed method (combined with extension 5.1) allows to calculate the amplitude and phase offset.

If the cause of the change was indeed a reflected wavefront, then the relationships between the calculated signal parameters would have to be the same as those of the direct sound. For example, if two harmonic frequencies f and $2f$ have a certain phase difference in the direct sound, then they should also have the same phase difference in a reflected wavefront, because the path a wavefront travels

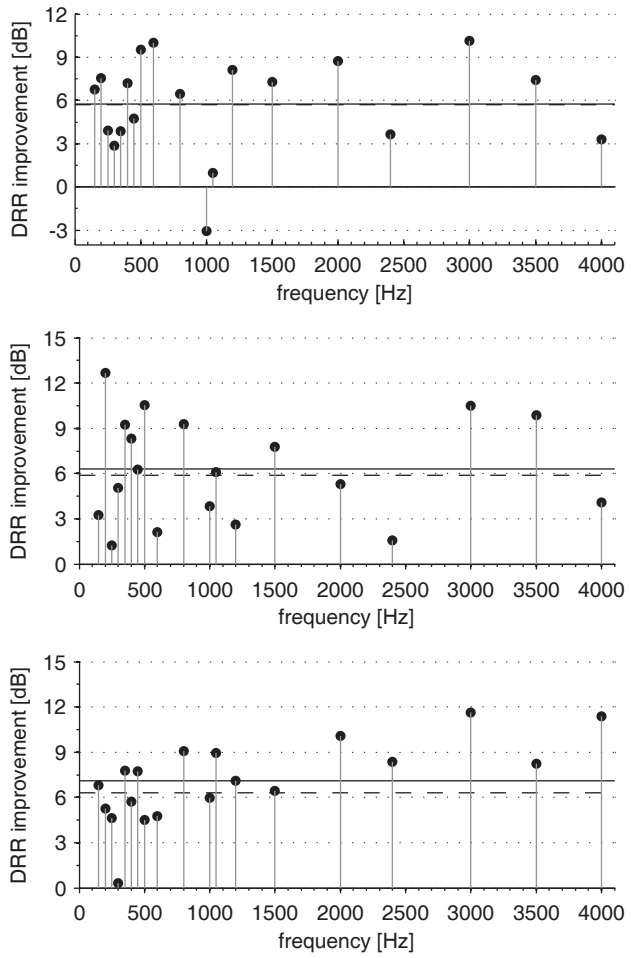

Fig. 3. stems: DRR improvement for rec. dist. of 0.6m (top), 1.2m (middle) and 1.8m (bottom); solid line: mean DRR improvement; dashed line: DRR improvement of a DSB using 5 microphones [5]

is the same for all frequencies. Likewise, the ratio of the amplitudes of f and $2f$ should be the same in the direct sound and in a single reflected wave.

5.3. Time-varying frequency

Each single harmonic frequency of the direct sound is variable in real speech, i.e. $\omega_{W_0} = \omega_{W_0}(t)$. However, the variable frequency $\omega_{W_0}(t)$ can be split into a constant part $\omega_{W_0}(0)$ (the frequency at the beginning) and a time-varying part $\Delta_\omega(t)$. The varying part can then be combined with the phase offset ϕ_{W_0} to form a time-varying phase offset $\phi_{W_0}(t) = \omega_{W_0}(t) \cdot t + \phi_{W_0}$. If the PLL tracks this, the phase deviations which are caused by a change in frequency can be separated from the actual phase deviations which occur due to arriving reflections.

6. CONCLUSIONS AND FUTURE WORK

The dereverberation algorithm described in this paper detects discontinuities of recording amplitude and phase, which are supposedly caused by reflected wavefronts. A reverb estimate is then calculated, which, when subtracted from the recording, restores the amplitude and phase offset of the direct sound.

The quality of the dereverberation achieved using a single microphone turned out to be equal to, and for some frequencies even better than, the results of a 5-microphone delay-and-sum beamforming array. At a recording distance of 1.8m the mean improvement of the direct-to-reverberant ratio was 7.1dB.

Further experiments with artificially reverberated signals have already shown that the estimated times for the arrival of reflected wavefronts do not always match the actual times given by the room impulse response. This, however, does not decrease the dereverberation performance, because a misdetected arrival time only results in an error until the next correctly detected arrival time. Nevertheless, further preliminary experiments suggest that misdetection can be reduced by using different amplitude estimators.

So far, it is difficult to make any statements regarding the performance of the method for real speech signals. Further experiments have already shown that it is possible to use a bandpass filterbank in order to split voiced speech signals into frequency bands which can then be tracked by individual PLLs. Future work includes an investigation on whether the tracking accuracy on these bandpass filtered signals is good enough and whether the proposed extensions of section 5 can be used as intended.

7. REFERENCES

[1] R. Best, *Phase-locked loops: theory, design and applications*, McGraw-Hill, New York, 2nd edition, 1993.

[2] P. Pelle, "A robust pitch extraction system based on phase locked loops," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toulouse, 2006.

[3] C. Estienne, P. Pelle, and J. Piantanida, "A front-end for speech recognition systems using phase-locked-loops," in *Proceedings of Workshop in Information Processing and Control (RPIC)*, Santa Fe, 2001.

[4] P. Pelle, C. Estienne, and H. Franco, "Robust speech representation of voiced sounds based on synchrony determination with plls," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[5] N. Gaubitch and P. Naylor, "Analysis of the dereverberation performance of microphone arrays," in *Proceedings of the International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Eindhoven, 2005.

[6] T. Nakatani, K. Kinoshita, and M. Miyoshi, "Harmonicity based blind dereverberation for single-channel speech signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 80 –95, 2007.

[7] K. Kinoshita, T. Nakatani, and M. Miyoshi, "Spectral subtraction steered by multi-step forward linear prediction for single channel speech dereverberation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toulouse, 2006.

[8] D. R. Stephens, *Phase locked loops for wireless communications: digital, analog and optical implementations*, Kluwer Academic Publishers, Boston, 2nd edition, 2002.

[9] M. Karimi-Ghartemani and M. R. Iravani, "A new phase-locked loop (pll) system," in *Proceedings of the IEEE Midwest Symposium on Circuits and Systems (MWSCAS)*, Dayton, Ohio, 2001.