Perceptually Optimized Cascaded Long Term Prediction of Polyphonic Signals for Enhanced MPEG-AAC

Tejaswi Nanjundaswamy and Kenneth Rose

Signal Compression Lab
Department of ECE
UCSB

October 21, 2011

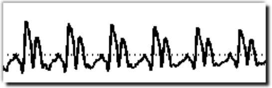
Outline

Outline

1. Introduction to perceptual audio coding and inter-frame prediction
2. Cascaded long term prediction (CLTP)
3. Extending CLTP with perceptual optimization for MPEG AAC
4. Results

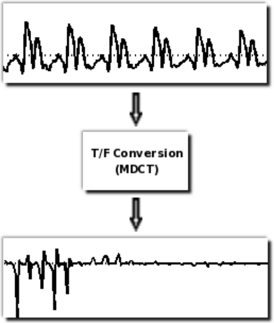
Audio Coding

- Most audio signals contain periodic components



Audio Coding

- Most audio signals contain periodic components
- Transformation is typically used to exploit redundancies within a frame

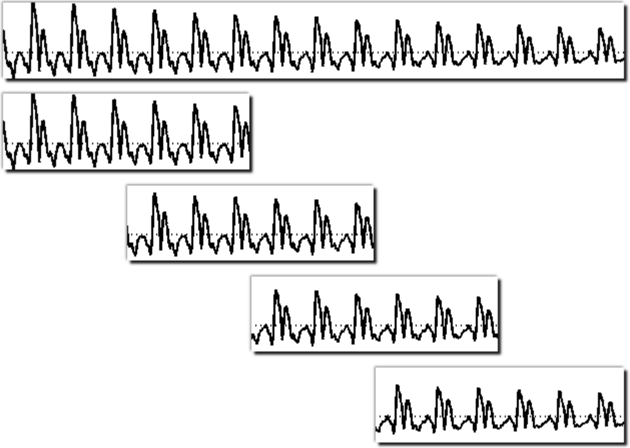


Audio Coding

- But transform coding blocks of data separately results in perceptually undesirable artifacts at the edges

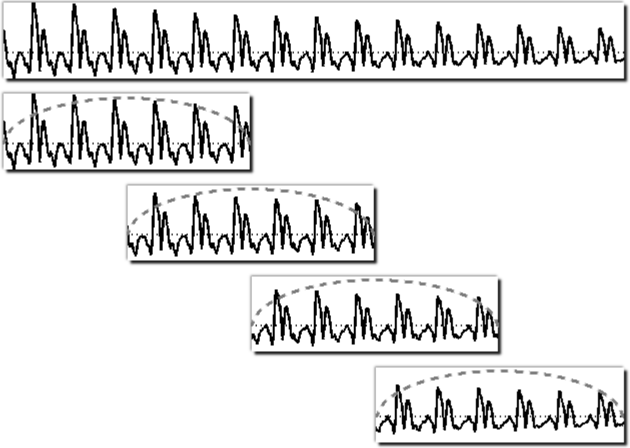
Audio Coding

- But transform coding blocks of data separately results in perceptually undesirable artifacts at the edges
- Solution: windowed overlapping frames



Audio Coding

- But transform coding blocks of data separately results in perceptually undesirable artifacts at the edges
- Solution: windowed overlapping frames



Audio Coding

- Coding in transform domain also facilitates psycho-acoustic redundancy removal
  - Eg: band wise noise masking
- This is captured in the distortion measure, Maximum Noise to Mask Ratio (MNMR)

$$\text{MNMR} = \max_{\forall \text{ bands}} \frac{\text{Quantization noise energy}}{\text{Masking threshold}}$$

- Finally the quantization and coding parameters are selected to minimize this perceptual distortion via the well known two-loop search (TLS) based technique
- Techniques which provide substantially better performance than TLS are known [Aggarwal et al. 2006], but we retain TLS for simplicity and for a fair comparison with reference encoders which use TLS

Audio Coding

- Coding in transform domain also facilitates psycho-acoustic redundancy removal
  - Eg: band wise noise masking
- This is captured in the distortion measure, Maximum Noise to Mask Ratio (MNMR)

$$\text{MNMR} = \max_{\forall \text{ bands}} \frac{\text{Quantization noise energy}}{\text{Masking threshold}}$$

- Finally the quantization and coding parameters are selected to minimize this perceptual distortion via the well known two-loop search (TLS) based technique
- Techniques which provide substantially better performance than TLS are known [Aggarwal et al. 2006], but we retain TLS for simplicity and for a fair comparison with reference encoders which use TLS

Audio Coding

- Coding in transform domain also facilitates psycho-acoustic redundancy removal
  - Eg: band wise noise masking
- This is captured in the distortion measure, Maximum Noise to Mask Ratio (MNMR)

$$\text{MNMR} = \max_{\forall \text{ bands}} \frac{\text{Quantization noise energy}}{\text{Masking threshold}}$$

- Finally the quantization and coding parameters are selected to minimize this perceptual distortion via the well known two-loop search (TLS) based technique
- Techniques which provide substantially better performance than TLS are known [Aggarwal et al. 2006], but we retain TLS for simplicity and for a fair comparison with reference encoders which use TLS

Audio Coding

- Coding in transform domain also facilitates psycho-acoustic redundancy removal
  - Eg: band wise noise masking
- This is captured in the distortion measure, Maximum Noise to Mask Ratio (MNMR)

$$\text{MNMR} = \max_{\forall \text{ bands}} \frac{\text{Quantization noise energy}}{\text{Masking threshold}}$$

- Finally the quantization and coding parameters are selected to minimize this perceptual distortion via the well known two-loop search (TLS) based technique
- Techniques which provide substantially better performance than TLS are known [Aggarwal et al. 2006], but we retain TLS for simplicity and for a fair comparison with reference encoders which use TLS

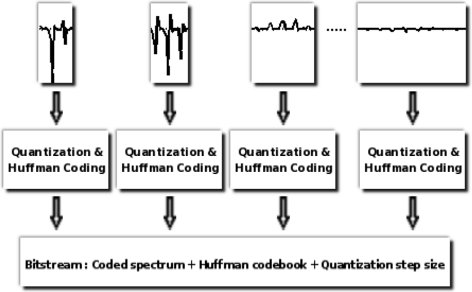
Audio Coding

- Coding in transform domain also facilitates psycho-acoustic redundancy removal
  - Eg: band wise noise masking
- This is captured in the distortion measure, Maximum Noise to Mask Ratio (MNMR)

$$\text{MNMR} = \max_{\forall \text{ bands}} \frac{\text{Quantization noise energy}}{\text{Masking threshold}}$$

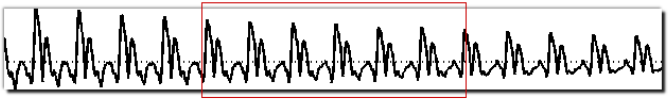
- Finally the quantization and coding parameters are selected to minimize this perceptual distortion via the well known two-loop search (TLS) based technique
- Techniques which provide substantially better performance than TLS are known [Aggarwal et al. 2006], but we retain TLS for simplicity and for a fair comparison with reference encoders which use TLS

Audio Coding



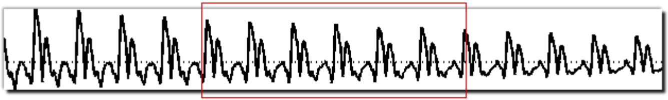
Audio Coding

- But temporal correlation usually extends beyond single frame



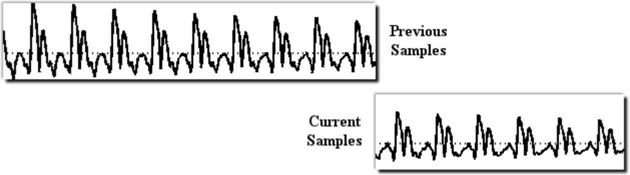
Audio Coding

- But temporal correlation usually extends beyond single frame
- Motivation to introduce the long term prediction (LTP) tool in MPEG AAC to exploit inter-frame redundancies [Ojanperä et al. 1999]



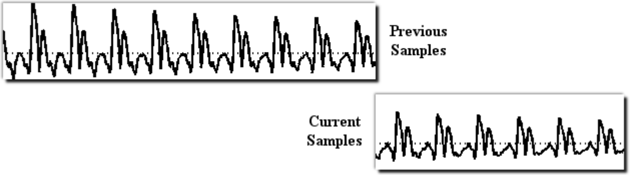
MPEG AAC LTP

- This tool predicts current frame from history



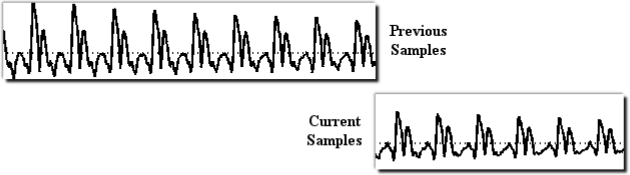
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor



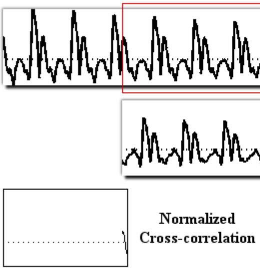
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error



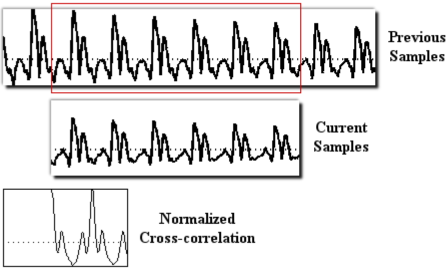
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error
- The resulting optimal lag maximizes the normalized cross-correlation



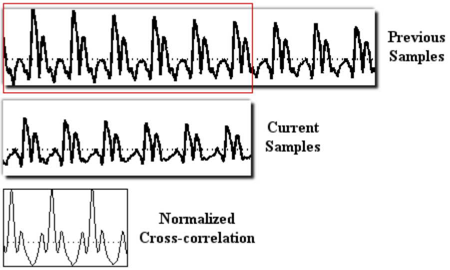
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error
- The resulting optimal lag maximizes the normalized cross-correlation



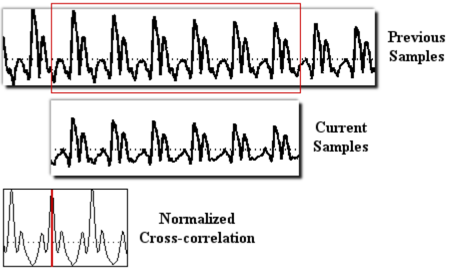
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error
- The resulting optimal lag maximizes the normalized cross-correlation



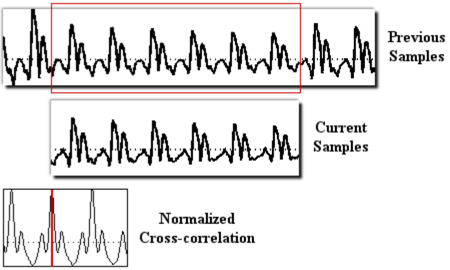
MPEG AAC LTP

- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error
- The resulting optimal lag maximizes the normalized cross-correlation



MPEG AAC LTP

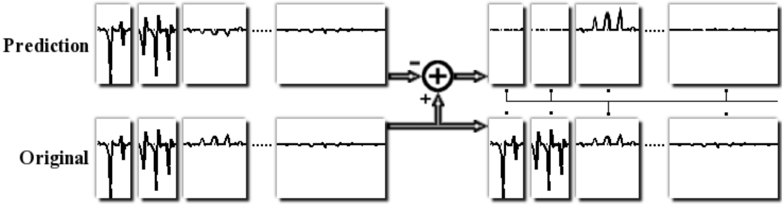
- This tool predicts current frame from history
- With reference position indicated via a lag, and waveforms are matched via gain factor
- The parameters are selected to minimize squared prediction error
- The resulting optimal lag maximizes the normalized cross-correlation
- And the gain matches the energy



MPEG AAC LTP

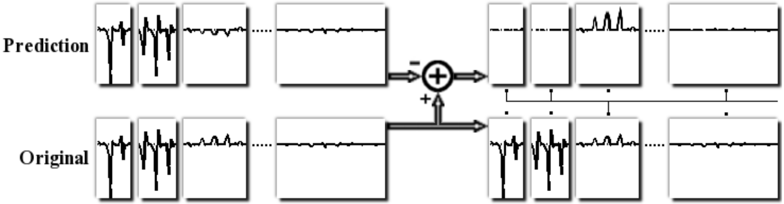
- The tool also provides a per band and per frame LTP activation flag

- The tool also provides a per band and per frame LTP activation flag
  - The per band flag is decided by comparing original with the prediction residue and selecting the lower energy option



MPEG AAC LTP

- The tool also provides a per band and per frame LTP activation flag
  - The per band flag is decided by comparing original with the prediction residue and selecting the lower energy option
  - The per frame flag is set if estimated bit savings due to LTP greater than the side-information rate



Motivation

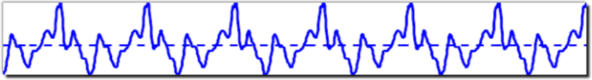
- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)

Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic

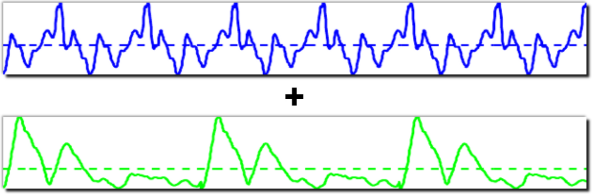
Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic



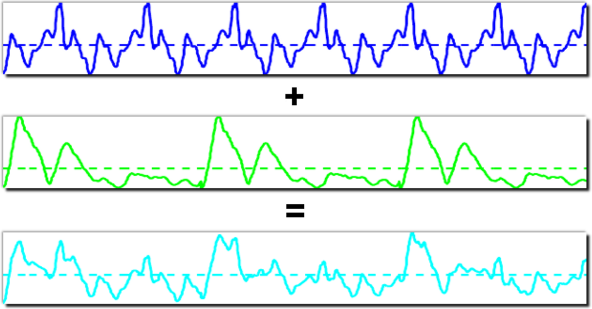
Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic



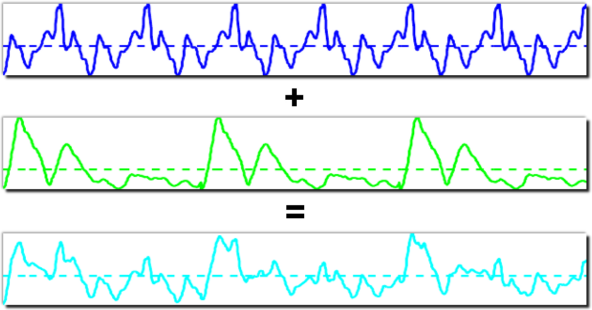
Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic



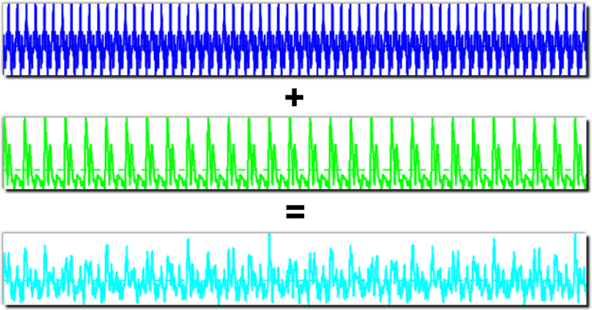
Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic
- In principle such a mixture is itself periodic



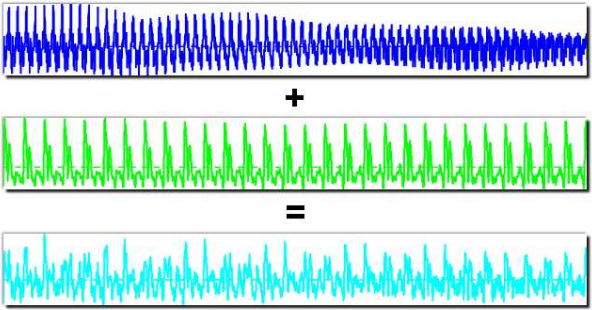
Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic
- In principle such a mixture is itself periodic
- Unfortunately the new period is too long and equal to the least common multiple (LCM) of individual periods



Motivation

- LTP effectively designed to work for monophonic audio signals (i.e., signals with one periodic component)
- But most audio signals are polyphonic
- In principle such a mixture is itself periodic
- Unfortunately the new period is too long and equal to the least common multiple (LCM) of individual periods
- And real audio signals rarely remain stationary for so long



Motivation

- LTP is suboptimal for realistic scenario
- Does this mean the inter frame redundancy is lost when periodic components are mixed?
- Or, is there a better way of exploiting this redundancy?

Motivation

- LTP is suboptimal for realistic scenario
- Does this mean the inter frame redundancy is lost when periodic components are mixed?
- Or, is there a better way of exploiting this redundancy?

Motivation

- LTP is suboptimal for realistic scenario
- Does this mean the inter frame redundancy is lost when periodic components are mixed?
- Or, is there a better way of exploiting this redundancy?

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Possible solutions

- Separate each component, predict individually and add
  - Not feasible for use in compression systems as currently known separation techniques are highly complex, inefficient or non-causal
- Prediction in frequency domain
  - Has been investigated and available in MPEG-2 AAC as a tool
  - Known to be as inefficient as the LTP tool described before
  - This tool's inefficiency usually associated to the fact that data is highly downsampled in the MDCT domain

Outline

Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m - N]$

Periodic signal model

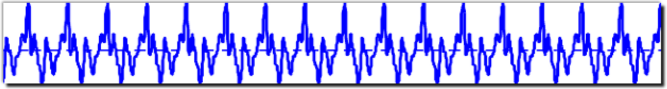
- Simple periodic signal can be characterized as $x[m] = x[m - N]$

Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m - N]$

Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m - N]$



Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m-N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m-N] + \beta x[m-N+1]$, which accounts for non-integral pitch periods and amplitude variation

Periodic signal model

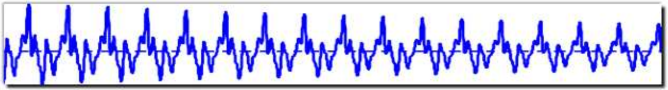
- Simple periodic signal can be characterized as $x[m] = x[m-N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m-N] + \beta x[m-N+1]$, which accounts for non-integral pitch periods and amplitude variation

Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m-N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m-N] + \beta x[m-N+1]$, which accounts for non-integral pitch periods and amplitude variation

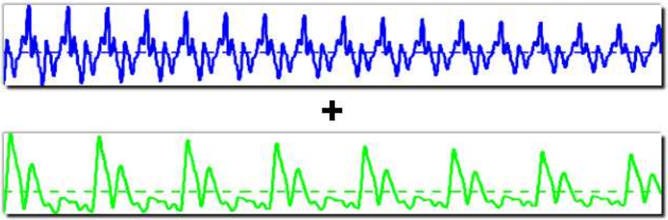
Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m - N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m - N] + \beta x[m - N + 1]$, which accounts for non-integral pitch periods and amplitude variation



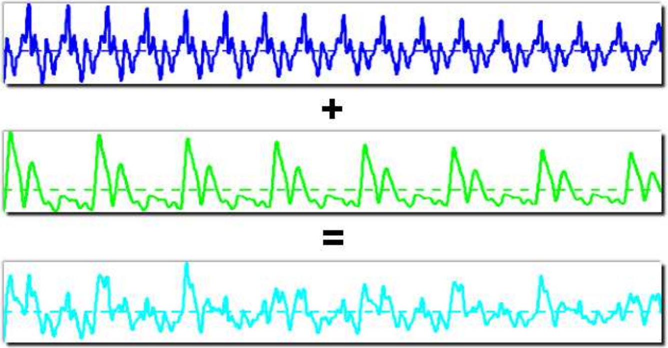
Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m-N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m-N] + \beta x[m-N+1]$, which accounts for non-integral pitch periods and amplitude variation
- Polyphonic audio signal is characterized as a mixture of such periodic signals and noise, i.e., $x[m] = \sum_{i=0}^{P-1} \sum x_i[m] + w[m]$



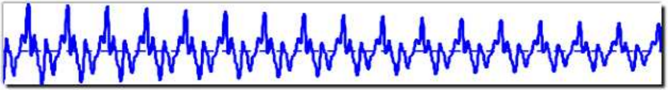
Periodic signal model

- Simple periodic signal can be characterized as $x[m] = x[m-N]$
- More realistic characterization used hereafter for a periodic component is $x[m] = \alpha x[m-N] + \beta x[m-N+1]$, which accounts for non-integral pitch periods and amplitude variation
- Polyphonic audio signal is characterized as a mixture of such periodic signals and noise, i.e., $x[m] = \sum_{i=0}^{P-1} \sum x_i[m] + w[m]$



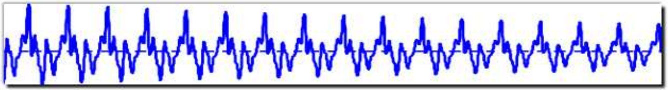
File with single periodic component

- For a audio file with single periodic component



File with single periodic component

- For a audio file with single periodic component
- The LTP filter $H(z) = 1 - \alpha z^{-N} - \beta z^{-N+1}$ predicts perfectly by design



File with single periodic component

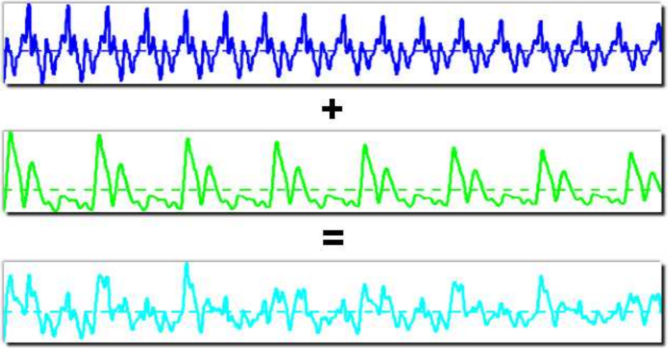
- For a audio file with single periodic component
- The LTP filter $H(z) = 1 - \alpha z^{-N} - \beta z^{-N+1}$ predicts perfectly by design

File with single periodic component

- For a audio file with single periodic component
- The LTP filter $H(z) = 1 - \alpha z^{-N} - \beta z^{-N+1}$ predicts perfectly by design
- Encoding this residue at current frame results in compression gains

File with multiple periodic components

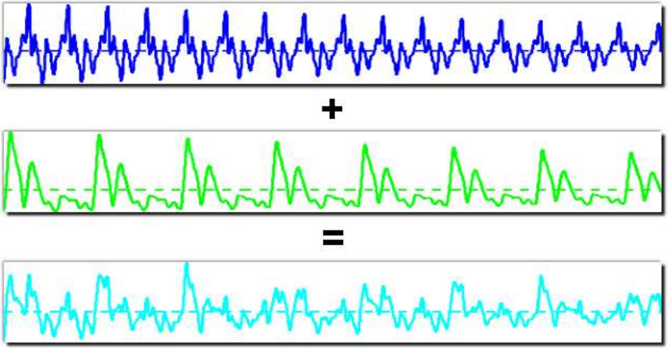
- How to predict a file with multiple periodic components?



- A single LTP filter with period at LCM is ineffective as signal doesn't remain stationary for such long durations

File with multiple periodic components

- How to predict a file with multiple periodic components?



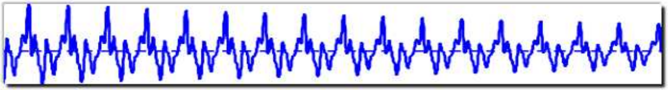
- A single LTP filter with period at LCM is ineffective as signal doesn't remain stationary for such long durations

File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components

File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components
- As per the design, it completely eliminates the first component

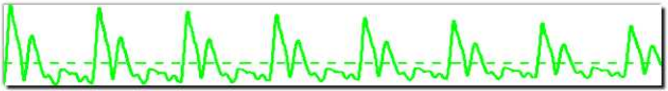


File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components
- As per the design, it completely eliminates the first component

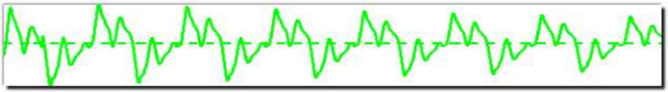
File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components
- As per the design, it completely eliminates the first component
- But it is of no help to the second component



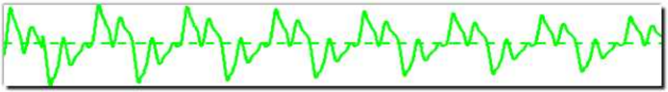
File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components
- As per the design, it completely eliminates the first component
- But it is of no help to the second component



File with multiple periodic components

- Instead let's see the impact of first component's LTP filter on different components
- As per the design, it completely eliminates the first component
- But it is of no help to the second component
- However notice that second component retains its periodicity even after application of this filter



File with multiple periodic components

- Thus filtering with LTP filter designed for second component

File with multiple periodic components

- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well

File with multiple periodic components

- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well

File with multiple periodic components

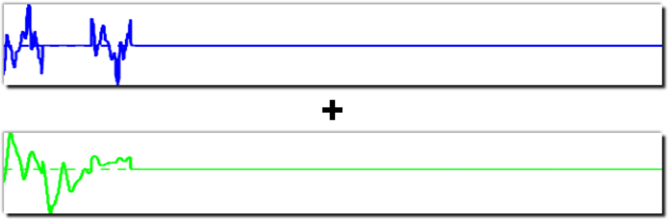
- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well
- Adding this to first component's new residue

File with multiple periodic components

- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well
- Adding this to first component's new residue

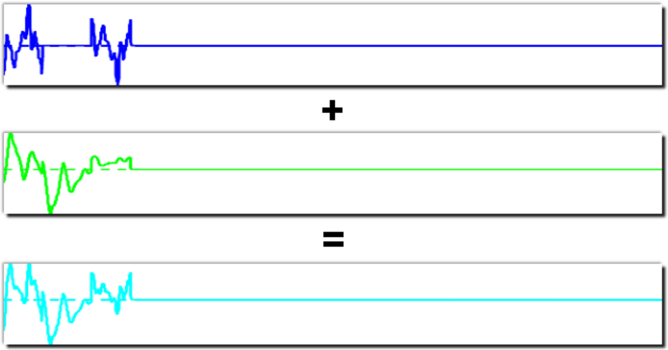
File with multiple periodic components

- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well
- Adding this to first component's new residue
- To get the new mixture residue



File with multiple periodic components

- Thus filtering with LTP filter designed for second component
- Eliminates the second component as well
- Adding this to first component's new residue
- To get the new mixture residue



Cascaded long term prediction

- Thus the cascaded long term prediction filter (CLTP) filter forms the basis of this proposal

$$H_c(z) = \prod_{i=0}^{P-1} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$$

- Note that for this filter to be effective a history of only $\sum_{i=0}^{P-1} N_i$ samples is required

Cascaded long term prediction

- Thus the cascaded long term prediction filter (CLTP) filter forms the basis of this proposal

$$H_c(z) = \prod_{i=0}^{P-1} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$$

- Note that for this filter to be effective a history of only $\sum_{i=0}^{P-1} N_i$ samples is required

Outline

Extending CLTP to MPEG AAC

- How to adapt a period wise predicting CLTP filter to MPEG AAC which operates with overlapping frames
- How to extend CLTP filter so that it can be optimized for the perceptual distortion criteria set in MPEG AAC

Extending CLTP to MPEG AAC

- How to adapt a period wise predicting CLTP filter to MPEG AAC which operates with overlapping frames
- How to extend CLTP filter so that it can be optimized for the perceptual distortion criteria set in MPEG AAC

Extending CLTP to MPEG AAC

- How to adapt a period wise predicting CLTP filter to MPEG AAC which operates with overlapping frames
- How to extend CLTP filter so that it can be optimized for the perceptual distortion criteria set in MPEG AAC

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

- With overlapping frames, some information about first half of the current frame is available from the previous frame
- But this is not useful for prediction within the current frame
- So the entire current frame predicted from fully reconstructed previous samples
- Which means a full block of data needs to be predicted
- The standard LTP does this by finding a match for the entire current frame in history
- But this is inefficient as now samples predicted from at least as far away as the frame length

Predicting with overlapping frames

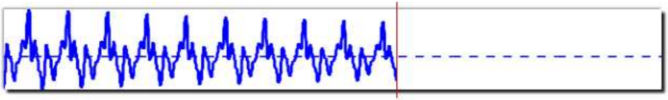
- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero

Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame

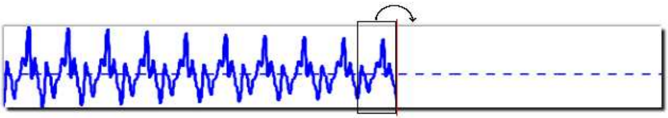
Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



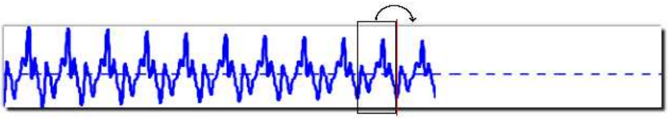
Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



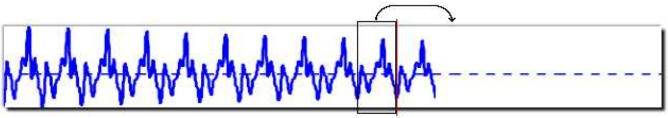
Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



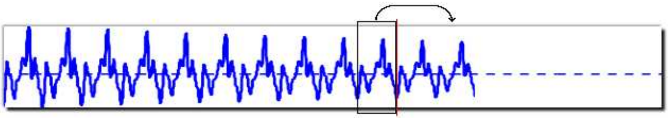
Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



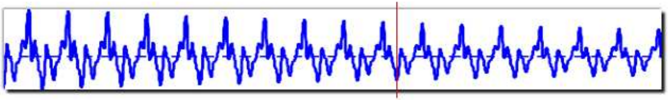
Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



Predicting with overlapping frames

- We instead retain CLTP filter in synthesis form [$1/H_c(z)$], and predict assuming residue to be zero
- This is effectively using most recently reconstructed samples recursively to predict the entire frame



CLTP filter parameter estimation

- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
  - In first stage a large subset estimated backward adaptively from previously reconstructed samples
    - Assumes signal to be locally stationary
    - Reduces side information rate
  - In the next stage, parameters refined to account for perceptual distortion

CLTP filter parameter estimation

- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
  - In first stage a large subset estimated backward adaptively from previously reconstructed samples
    - Assumes signal to be locally stationary
    - Reduces side information rate
  - In the next stage, parameters refined to account for perceptual distortion

CLTP filter parameter estimation

- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
  - In first stage a large subset estimated backward adaptively from previously reconstructed samples
    - Assumes signal to be locally stationary
    - Reduces side information rate
  - In the next stage, parameters refined to account for perceptual distortion

CLTP filter parameter estimation

- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
  - In first stage a large subset estimated backward adaptively from previously reconstructed samples
    - Assumes signal to be locally stationary
    - Reduces side information rate
  - In the next stage, parameters refined to account for perceptual distortion

CLTP filter parameter estimation

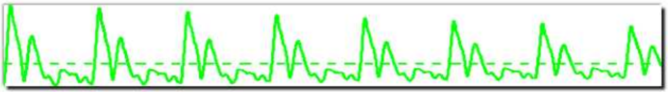
- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
  - In first stage a large subset estimated backward adaptively from previously reconstructed samples
    - Assumes signal to be locally stationary
    - Reduces side information rate
  - In the next stage, parameters refined to account for perceptual distortion

CLTP filter parameter estimation

- Derivation of the CLTP filter demonstrated that it can be practically very effective
- But this critically depends on suitable parameter estimation, which accounts for perceptual distortion criteria
- This achieved in two stages to keep complexity in check
    - In first stage a large subset estimated backward adaptively from previously reconstructed samples
        - Assumes signal to be locally stationary
        - Reduces side information rate
    - In the next stage, parameters refined to account for perceptual distortion

Backward adaptive parameter estimation

- Underlying observation for parameter estimation: Periodicity of a component retained after linear filtering



Backward adaptive parameter estimation

- Underlying observation for parameter estimation: Periodicity of a component retained after linear filtering

Backward adaptive parameter estimation

- Thus, parameters of jth filter of the cascade are estimated in the residue after filtering with all the others $\prod_{\forall i, i \neq j} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$
- Estimating parameters of one filter $(1 - \alpha_j z^{-N_j} - \beta_j z^{-N_j+1})$ simply follows the well known LTP problem
- Each filter in the cascade is estimated this way in a loop until convergence
- As overall prediction error is monotone non-increasing at each step, convergence is guaranteed

Backward adaptive parameter estimation

- Thus, parameters of jth filter of the cascade are estimated in the residue after filtering with all the others $\prod_{\forall i, i \neq j} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$
- Estimating parameters of one filter $(1 - \alpha_j z^{-N_j} - \beta_j z^{-N_j+1})$ simply follows the well known LTP problem
- Each filter in the cascade is estimated this way in a loop until convergence
- As overall prediction error is monotone non-increasing at each step, convergence is guaranteed

Backward adaptive parameter estimation

- Thus, parameters of jth filter of the cascade are estimated in the residue after filtering with all the others $\prod_{\forall i,\, i \neq j} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$
- Estimating parameters of one filter $(1 - \alpha_j z^{-N_j} - \beta_j z^{-N_j+1})$ simply follows the well known LTP problem
- Each filter in the cascade is estimated this way in a loop until convergence
- As overall prediction error is monotone non-increasing at each step, convergence is guaranteed

Backward adaptive parameter estimation

- Thus, parameters of jth filter of the cascade are estimated in the residue after filtering with all the others $\prod_{\forall i, i \neq j} (1 - \alpha_i z^{-N_i} - \beta_i z^{-N_i+1})$
- Estimating parameters of one filter $(1 - \alpha_j z^{-N_j} - \beta_j z^{-N_j+1})$ simply follows the well known LTP problem
- Each filter in the cascade is estimated this way in a loop until convergence
- As overall prediction error is monotone non-increasing at each step, convergence is guaranteed

Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag

Backward adaptive parameter estimation

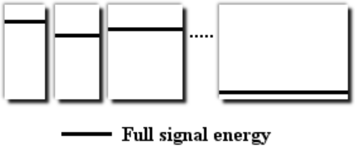
- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account

Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful

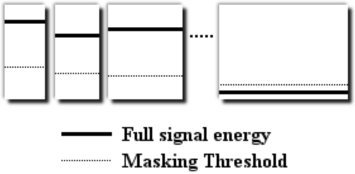
Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful



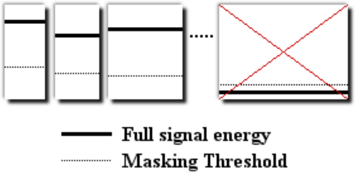
Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful



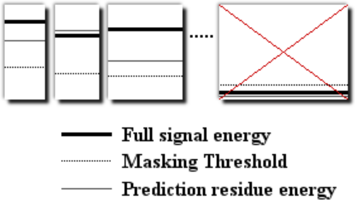
Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful



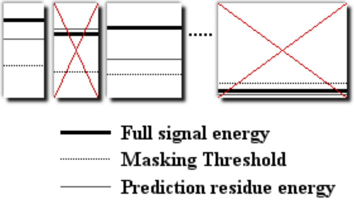
Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful



Backward adaptive parameter estimation

- Given the filter, residue in the previously reconstructed data generated to decide band wise prediction activation flag
  - This flag is similar to the one described in standard LTP
  - But estimated backward adaptively as signal assumed to be locally stationary
  - Also psycho-acoustic masking thresholds are taken into account
- Prediction retained in bands where signal energy is above masking thresholds (as only those will be encoded) and bands where prediction is useful



Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion
- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1}(1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant
- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion
- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1} (1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant
- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion

- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1} (1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant

- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion
- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1} (1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant
- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion
- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1}(1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant
- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- Amongst CLTP parameters, N_i and part of α_i, β_i which capture the non-integral pitch period are dependent only on a component's waveform and not impacted by perceptual distortion
- Thus we break α_i, β_i and introduce gain factors G_i to form an updated CLTP filter

$$H_c(z) = \prod_{i=0}^{P-1}(1 - G_i(\alpha_i z^{-N_i} + \beta_i z^{-N_i+1}))$$

- These gains adapt each periodic component's filter according to the perceptual distortion criteria. For example:
  - Some components might be perceptually more important than others
  - Adapt coefficients to filter only harmonics that are perceptually significant
- The gain factors are quantized to one of the four levels (0.5, 0.75, 1, 1.25) and sent as side information to the decoder

Accounting perceptual distortion

- The estimation of gain factors to minimize perceptual distortion for a given rate is achieved via a two stage process
- In the first stage the squared prediction error is calculated for all combinations of gain factors for different P
- Amongst these top S squared-prediction-error minimizing combinations are retained

Accounting perceptual distortion

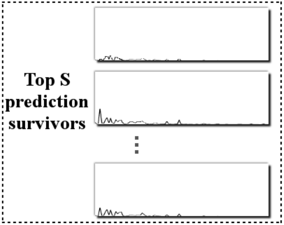
- The estimation of gain factors to minimize perceptual distortion for a given rate is achieved via a two stage process
- In the first stage the squared prediction error is calculated for all combinations of gain factors for different P
- Amongst these top S squared-prediction-error minimizing combinations are retained

Accounting perceptual distortion

- The estimation of gain factors to minimize perceptual distortion for a given rate is achieved via a two stage process
- In the first stage the squared prediction error is calculated for all combinations of gain factors for different P
- Amongst these top S squared-prediction-error minimizing combinations are retained

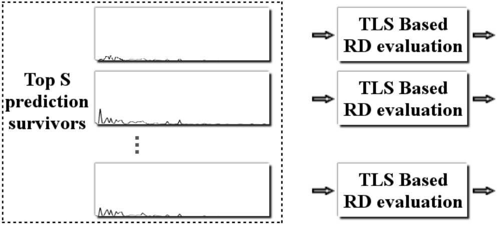
Accounting perceptual distortion

- In the second stage, each of these S survivors rate distortion (RD) evaluated via TLS



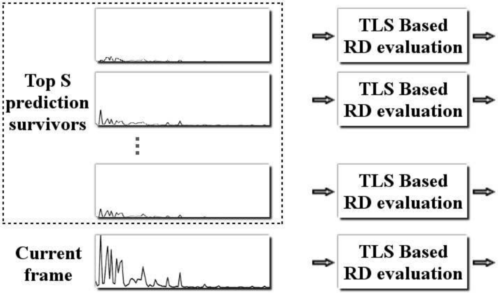
Accounting perceptual distortion

- In the second stage, each of these S survivors rate distortion (RD) evaluated via TLS



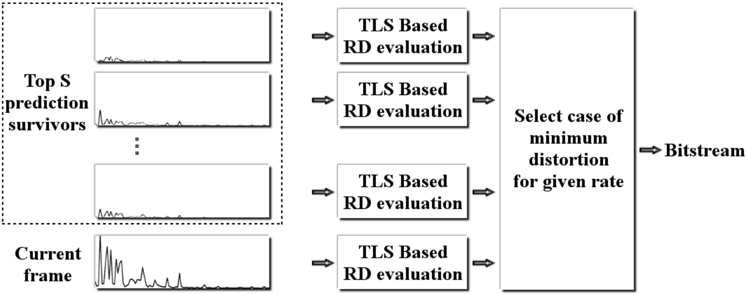
Accounting perceptual distortion

- In the second stage, each of these S survivors rate distortion (RD) evaluated via TLS
- To find per frame flag, the original frame also RD evaluated



Accounting perceptual distortion

- In the second stage, each of these S survivors rate distortion (RD) evaluated via TLS
- To find per frame flag, the original frame also RD evaluated
- Parameters resulting in minimum distortion for a given rate chosen



Outline

1. Introduction to perceptual audio coding and inter-frame prediction
2. Cascaded long term prediction (CLTP)
3. Extending CLTP with perceptual optimization for MPEG AAC
4. Results

Results

- The following three low delay coders compared in our evaluations
  - MPEG reference encoder with no LTP
  - MPEG reference encoder with standard LTP
  - Proposed encoder with CLTP
- Test data set includes real polyphonic audio samples (44.1 / 48 kHz, single channel) from the MPEG standard and EBU SQAM

Results

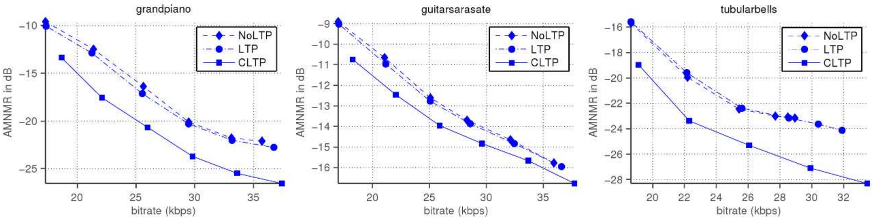
- The following three low delay coders compared in our evaluations
  - MPEG reference encoder with no LTP
  - MPEG reference encoder with standard LTP
  - Proposed encoder with CLTP
- Test data set includes real polyphonic audio samples (44.1 / 48 kHz, single channel) from the MPEG standard and EBU SQAM

Objective evaluation results

- Average MNMR (AMNMR) versus bitrate

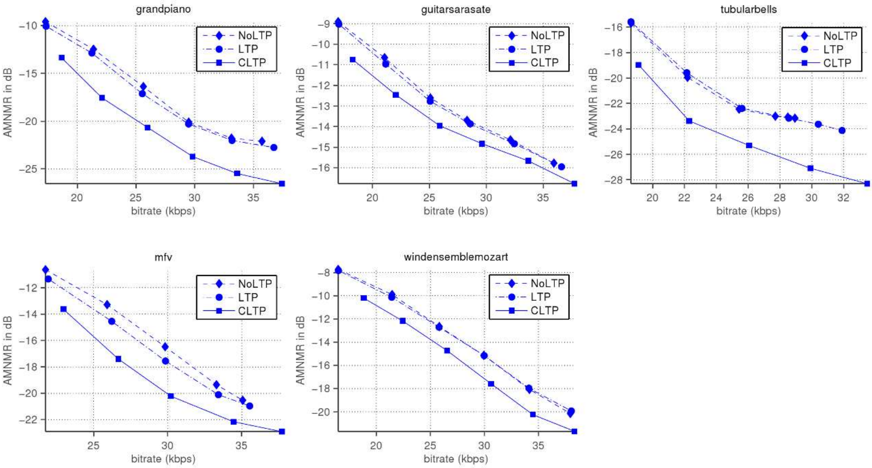
Objective evaluation results

- Average MNMR (AMNMR) versus bitrate



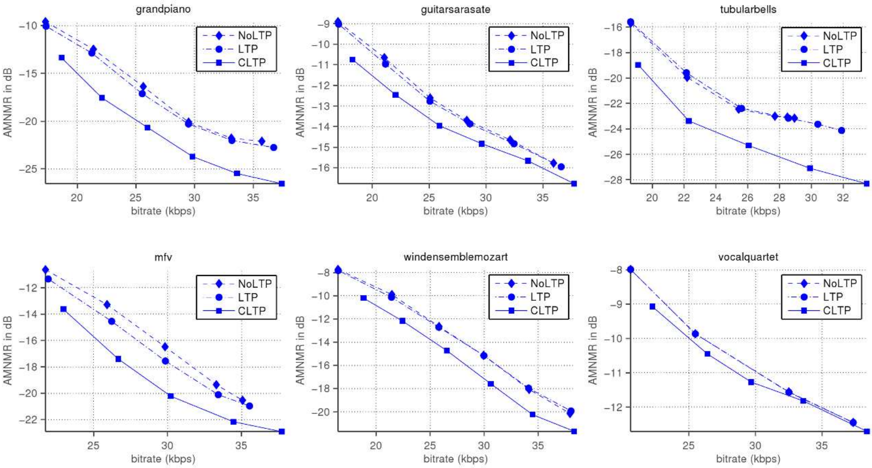
Objective evaluation results

- Average MNMR (AMNMR) versus bitrate



Objective evaluation results

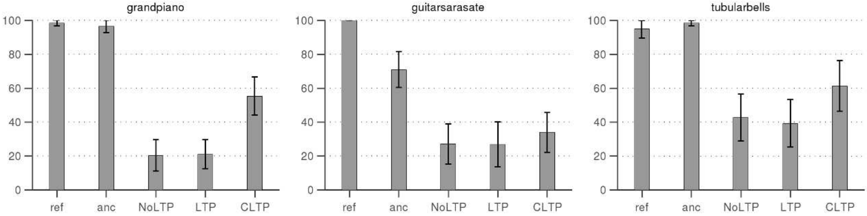
- Average MNMR (AMNMR) versus bitrate



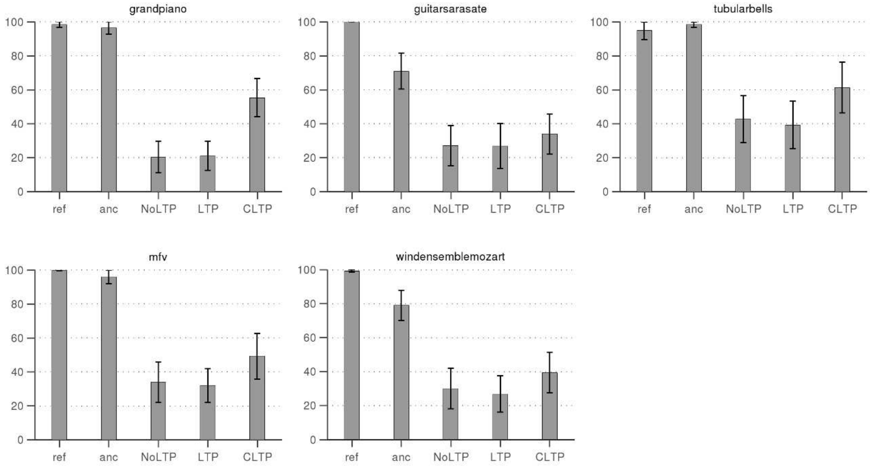
Subjective evaluation

- MUSHRA listening tests for coders operating at 24 kbps
- 15 listeners score on a scale of 0 (bad) to 100 (excellent)
- Plots show average MUSHRA scores and 95% confidence interval

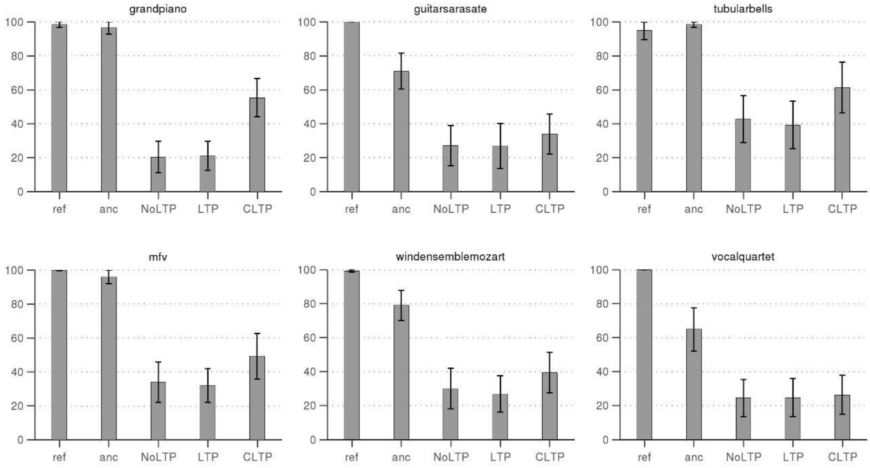
Subjective evaluation results



Subjective evaluation results



Subjective evaluation results



Summary

- Currently used standard LTP sub-optimal for polyphonic signals
- Cascading LTP filters to optimally predict polyphonic signals proposed
- Extending CLTP to MPEG AAC while taking perceptual distortion into account proposed
- Subjective and objective evaluations show substantial improvements
- We conclude that such improved inter-frame redundancy removal could bridge gap between low delay and long block length coders

Summary

- Currently used standard LTP sub-optimal for polyphonic signals
- Cascading LTP filters to optimally predict polyphonic signals proposed
- Extending CLTP to MPEG AAC while taking perceptual distortion into account proposed
- Subjective and objective evaluations show substantial improvements
- We conclude that such improved inter-frame redundancy removal could bridge gap between low delay and long block length coders

Summary

- Currently used standard LTP sub-optimal for polyphonic signals
- Cascading LTP filters to optimally predict polyphonic signals proposed
- Extending CLTP to MPEG AAC while taking perceptual distortion into account proposed
- Subjective and objective evaluations show substantial improvements
- We conclude that such improved inter-frame redundancy removal could bridge gap between low delay and long block length coders

Summary

- Currently used standard LTP sub-optimal for polyphonic signals
- Cascading LTP filters to optimally predict polyphonic signals proposed
- Extending CLTP to MPEG AAC while taking perceptual distortion into account proposed
- Subjective and objective evaluations show substantial improvements
- We conclude that such improved inter-frame redundancy removal could bridge gap between low delay and long block length coders

Summary

- Currently used standard LTP sub-optimal for polyphonic signals
- Cascading LTP filters to optimally predict polyphonic signals proposed
- Extending CLTP to MPEG AAC while taking perceptual distortion into account proposed
- Subjective and objective evaluations show substantial improvements
- We conclude that such improved inter-frame redundancy removal could bridge gap between low delay and long block length coders

Thank you for your attention

Questions?