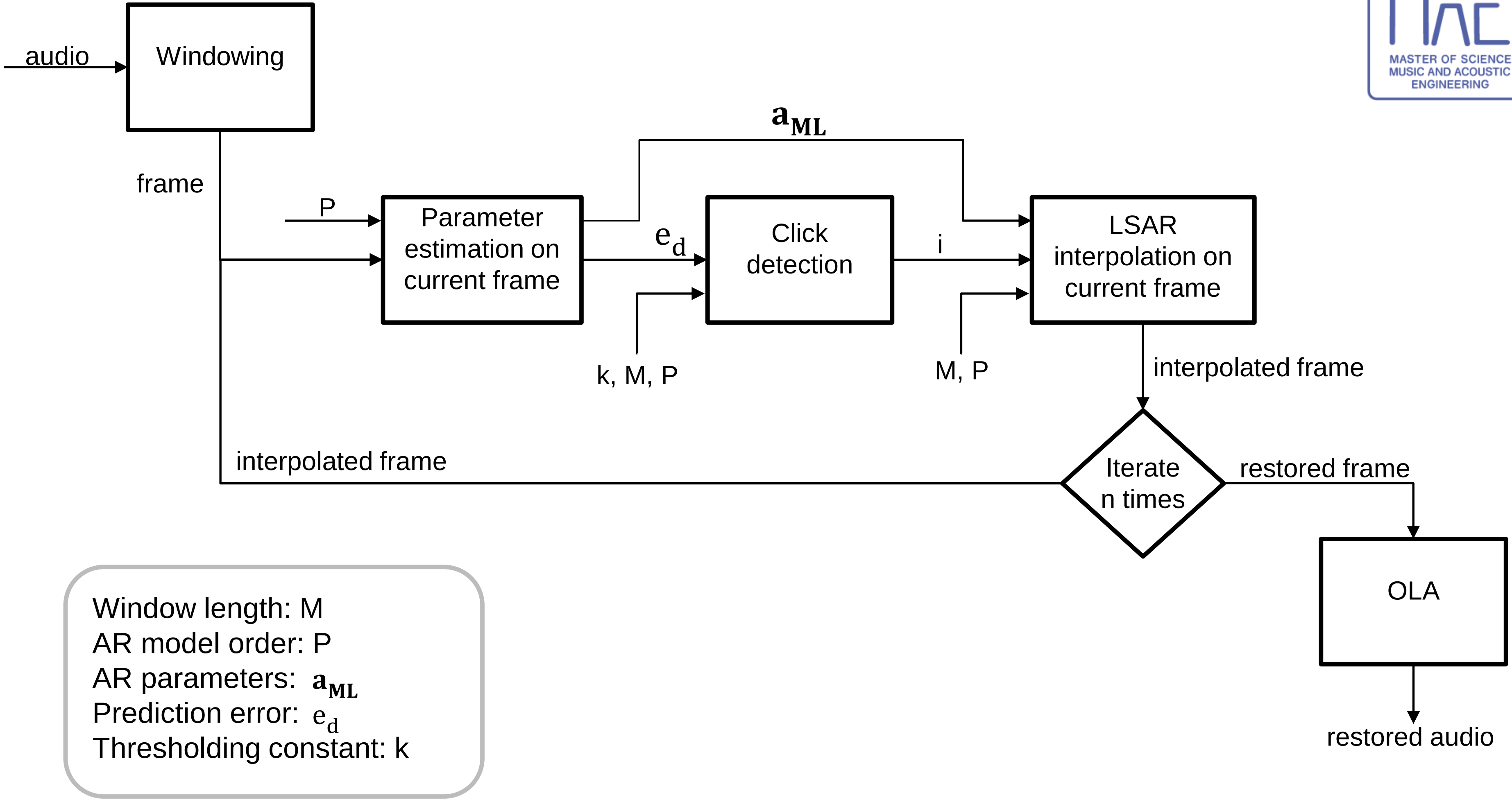
LSAR Click removal

The algorithm is composed of 5 steps:

1. Windowing
2. Parameter estimation
3. Click detection
4. LSAR interpolation
5. Overlap and Add

Algorithm’s characteristics:

- Input is drawn from a locally stationary AR process → **windowing**
  E.g., Hanning window of length M = 2048 and 50% overlap
- Steps from 2 to 4 can be **iterated** to improve performance


audio
Windowing
frame
P
Parameter estimation on current frame
$\mathbf{a}_{ML}$
e_d
Click detection
k, M, P
i
LSAR interpolation on current frame
M, P
interpolated frame
Iterate n times
interpolated frame
restored frame
OLA
restored audio
Window length: M
AR model order: P
AR parameters: $\mathbf{a}_{ML}$
Prediction error: e_d
Thresholding constant: k

Parameter estimation

Using the Maximum Likelihood (ML) estimator

- Build the matrix $\mathbf{G_x}$ (see General linear model)
- Estimate the parameters $\mathbf{a_{ML}}$ by computing the pseudoinverse of $\mathbf{G_x}$

```
a_ml = pinv(G_x)*x_hat';
```

- Compute the prediction error by filtering the corrupted signal with the prediction error filter

```
H_a = [1; -a_ml];
e_d = filter(H_a, 1, x);
```

Click detection

MASTER OF SCIENCE MUSIC AND ACOUSTIC ENGINEERING

The prediction error is the sum of the innovation process of the AR model (which passes unfiltered) and the filtered clicks:

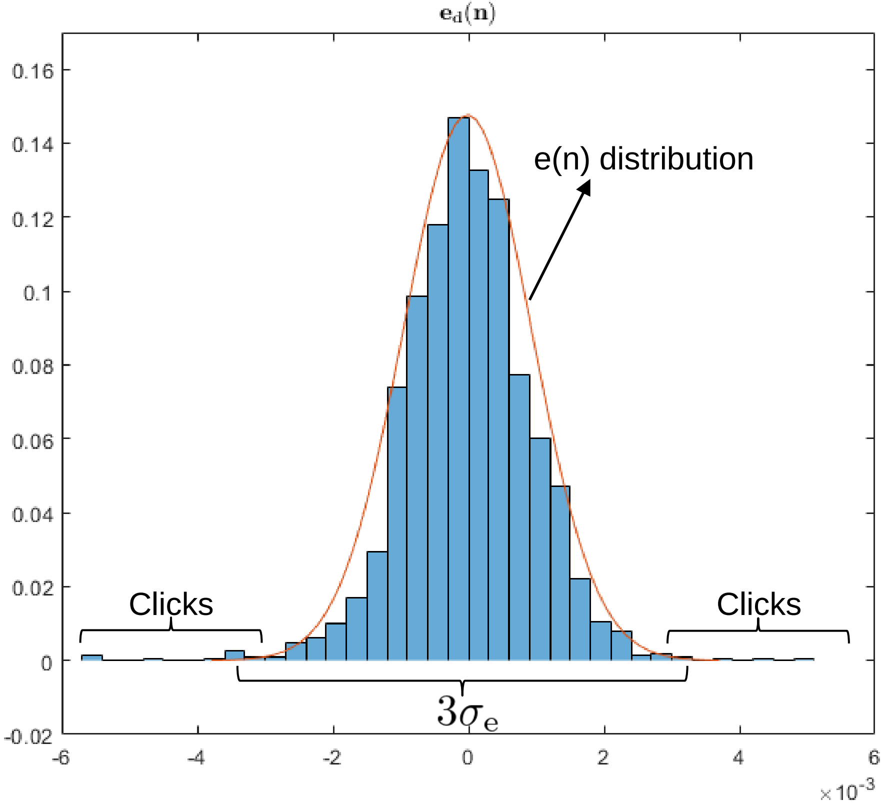
$$e_d(n) = e(n) + i(n)v(n) * h(n)$$

How to discriminate between the two?

- e(n) takes small values (Gaussian assumptions)
- clicks cause large error since the filter fails to predict

If we compute the standard deviation of e(n), we can tell when a sample of the prediction error is drawn from e(n) or not (i.e. it is a click!)

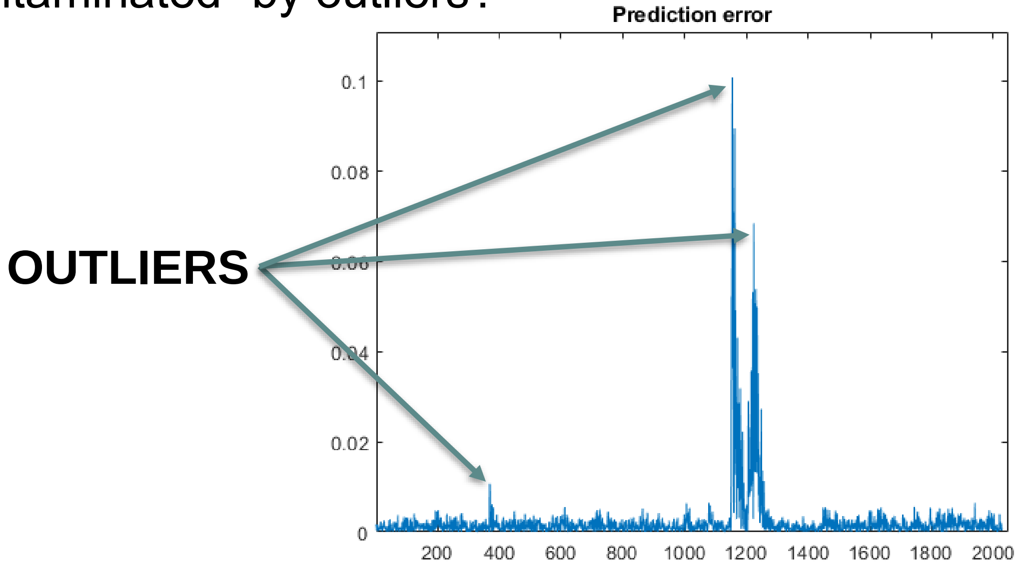
Standard deviation



Histogram of $e_d(n)$

Click detection

Problem: how to compute the standard deviation of a Guassian process when it is “contaminated” by outliers?



Solution: use the **Median absolute deviation** (MAD), which is a robust measure of the variability of a process

Median absolute deviation

MASTER OF SCIENCE MUSIC AND ACOUSTIC ENGINEERING

Given a sequence of samples $X = \{ X_1, X_2, \ldots, X_n \}$, the median absolute deviation is defined as

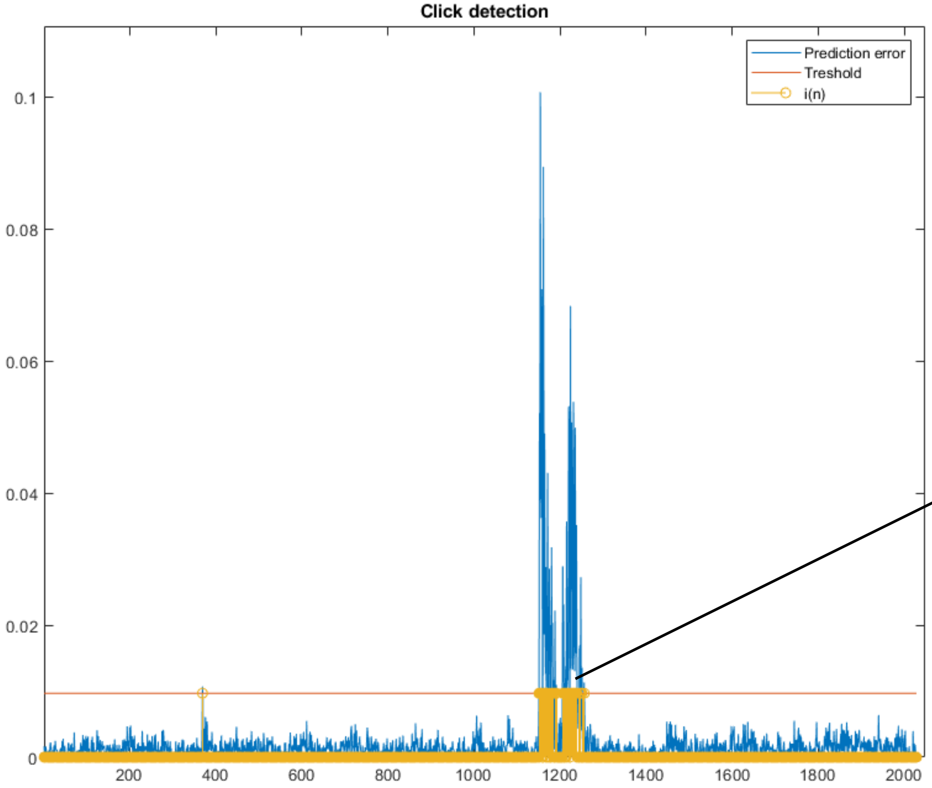
$$\mathrm{MAD} = \mathrm{median}(|\mathrm{X_i} - \mathrm{median(X)}|)$$

And it can been shown that for a Gaussian distribution (as in our case for e(n)) the following relation for the standard deviation holds:

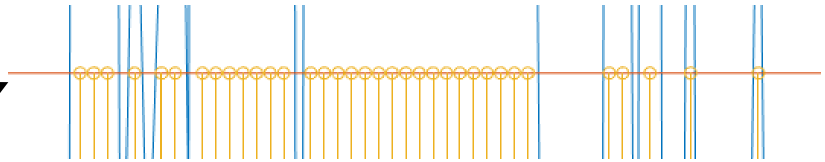
$$\sigma \approx 1.4826 \times \mathrm{MAD}$$

Thresholding

$$\texttt{if } |e(n)| > k\sigma_e \texttt{ then } i(n) = 1\ ,\ \ \texttt{else } i(n) = 0$$



→ The constant k generally can takes values from 3 to 8



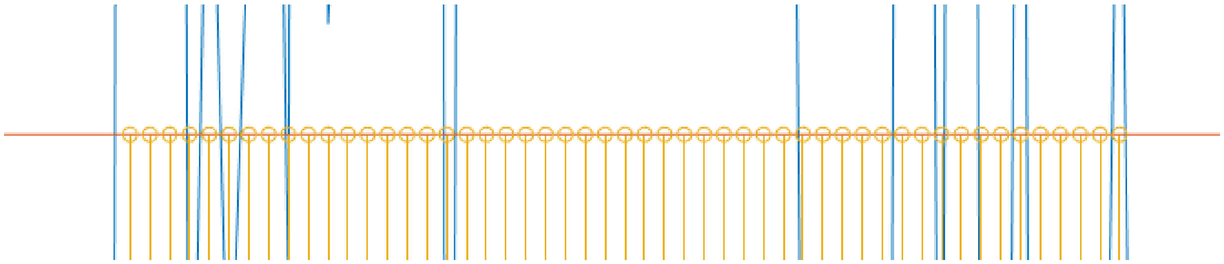
Grouping detections

Problem:

- to properly work, the LSAR interpolator needs at least P samples on the left and on the right
- Clicks trigger a full impulse response of the prediction error filter, thus an irregular behaviour afterward a click’s peak in the prediction error

$\Rightarrow$ detections are too close for the interpolator to work

Solution: group the detections in clusters by filling the gaps



LSAR interpolation

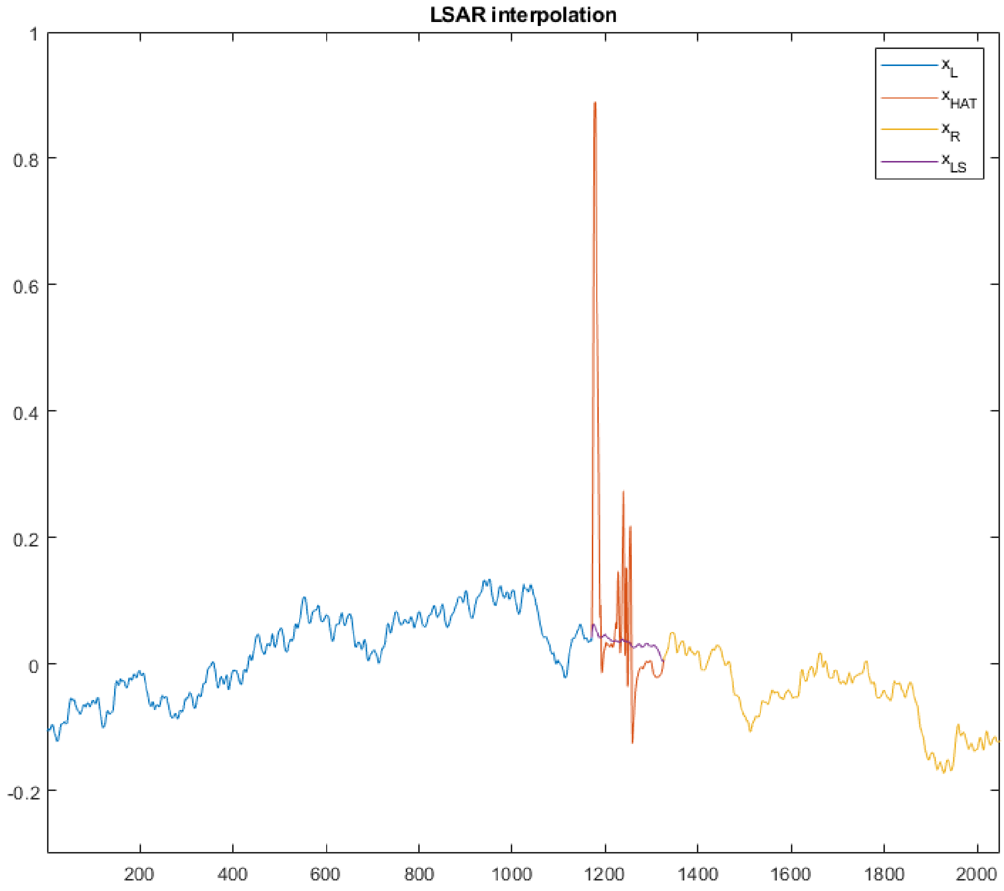
For each detected click:

- Build the relocation matrices **U** and **K**
- Build the matrix **A** (see linear model)
- Compute $\hat{\mathrm{A}} = \mathrm{AU}$ and $\bar{\mathrm{A}} = \mathrm{AK}$
- Interpolate by computing the pseudoinverse of $\hat{\mathrm{A}}$:

```
x_ls = -pinv(A_hat)*A_sgn*x_sgn;
```

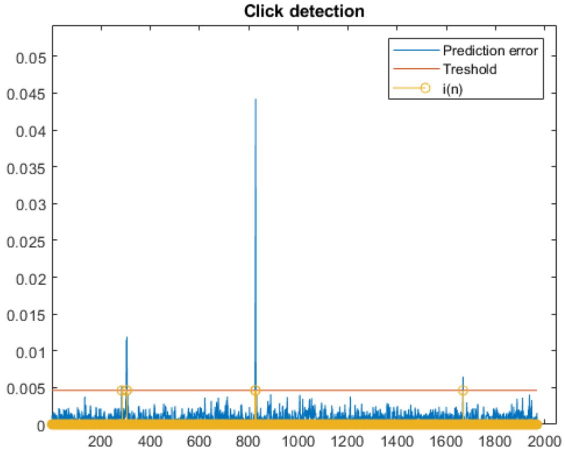
LSAR interpolation

P = 80

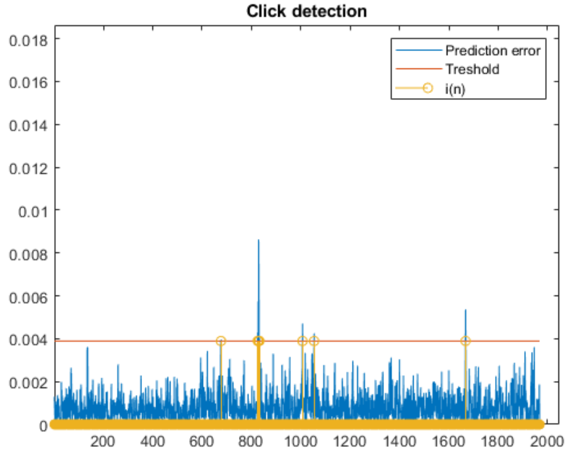


The importance of iteration

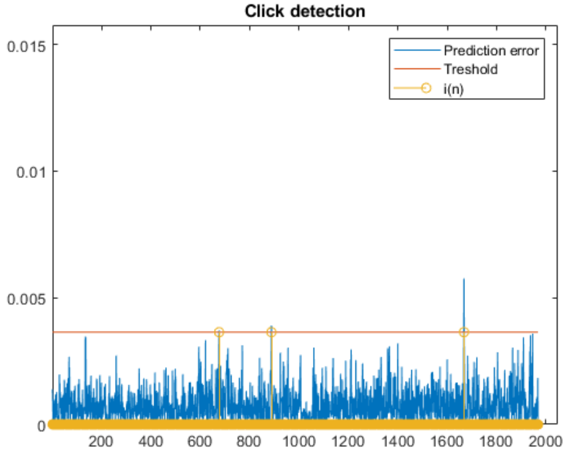
Clicks can be still classified as clicks even after interpolation



First iteration

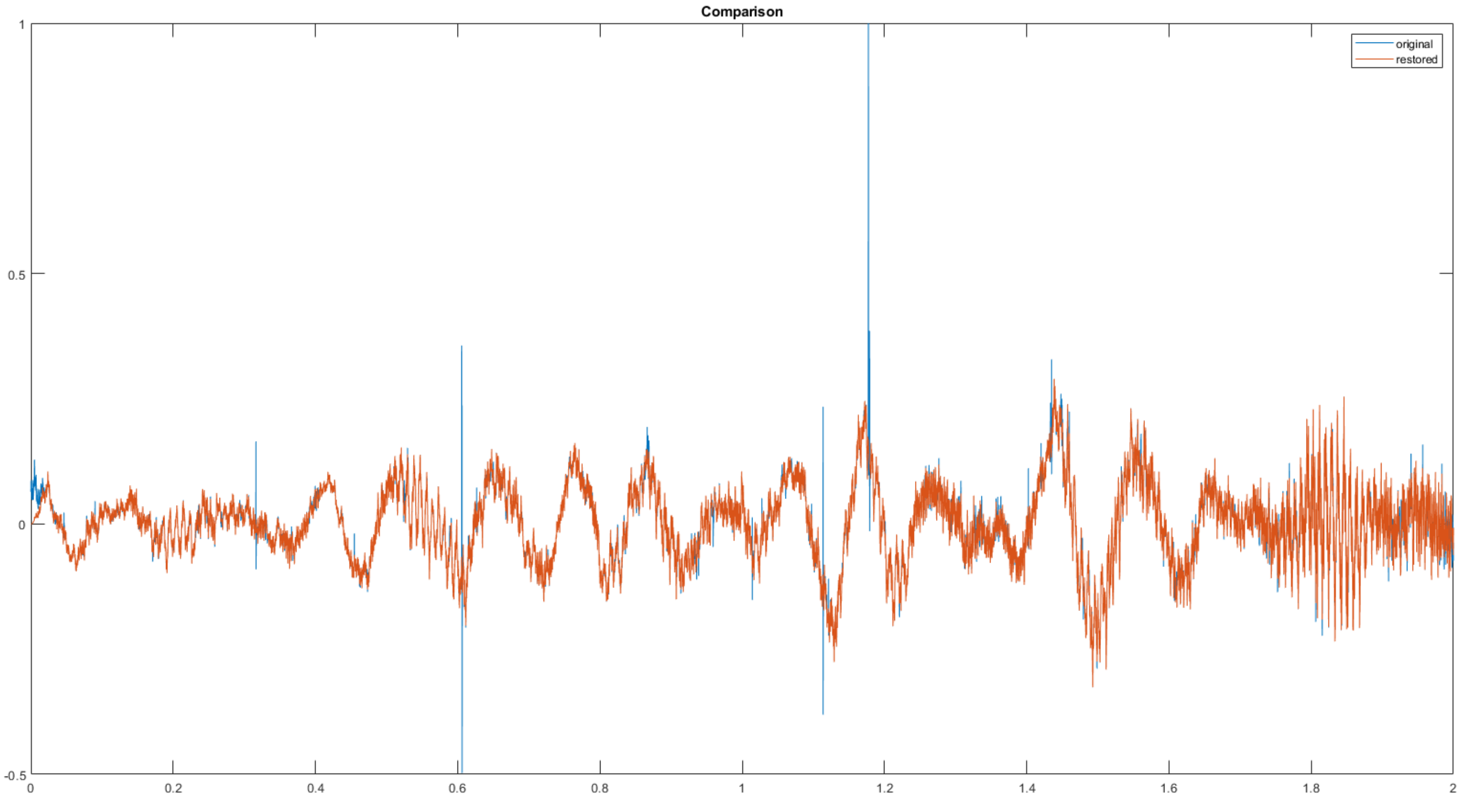


Second iteration



Third iteration

Results



[Digital Audio Restoration - A Statistical Model-Based Approach]. Mussorgsky - Night on Bald Mountain
M = 2048 with 50% overlap, k = 3.5, P = 20, 5 iterations

Further improvements

MASTER OF SCIENCE
MUSIC AND ACOUSTIC
ENGINEERING

- This implementation uses the function *pinv()* for computing the pseudoinverse of a *m* by *n* matrix. This function relies on the Singular Value Decomposition (SVD) which has complexity O(max(m, n)*min(m, n)^2), not **exploiting the Toepliz structure** of the system (Levinson-Durbin algorithm is O(m^2))
- Find a way to avoid grouping detections by **rewriting relocation matrices** that take into account gaps of length less than P samples
- Is there a way to **dynamically choose the constant k** by optimizing a metric?
- Is there a way to **dynamically choose the number of iterations** by optimizing a metric?