Computers and Electrical Engineering 000 (2017) 1–18

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A compression-based backward approach for the forward sparse modeling with application to speech coding☆

A.N. Omara[a,*], A.A. Hefnawy[a], Abdelhalim Zekry[b]

[a] Computers and Systems Department, Electronics Research Institute, Giza, Egypt
[b] Communications and Electronics Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt

ARTICLE INFO

Article history:
Received 12 June 2016
Revised 1 May 2017
Accepted 6 May 2017
Available online xxx

Keywords:
Sparse representation
Forward greedy pursuit
Backward elimination
Backward replacement
Speech coding

ABSTRACT

This paper concentrates on the development of an efficient and robust backward solution for the forward sparse greedy algorithms and applies this solution in the field of speech compression. All existing backward solutions are based on constraining more and more weights to zero while re-optimizing the remaining nonzero weights to compensate. Our approach is termed Backward Replacement (BRe) algorithm and its idea is to replace the $k-$sparse weights vector with a $k-$sparse symmetric matrix. The key result of this paper showed that, the replacement approach has demonstrated successfully the superiority over existing backward elimination algorithms in both enhancing the compression capabilities of the forward greedy algorithms, and reducing the time complexity.

1. Introduction

Despite the evolution of the hybrid speech codecs such as Adaptive Multi-Rate codec (AMR [1], AMR-wideband codec (AMR-WB) [2] and Enhanced Voice Services codec (EVS) [3]), the waveform-based speech codecs such as Pulse Code Modulation (PCM) [4] and adaptive PCM (ADPCM) [5] are still used in applications, for example, they can be used to send audio on fiber-optic long-distance lines as well as to record it on electronic devices such as CDs, USBs and others. Also, there are recent enhancements to the ADPCM such as NUT-ADPCM [6] that yields improvements in both signal-to-noise ratio and average bit rate. In this work, we consider the usage of the sparse modeling as a waveform-based speech encoder that tries to reproduce the speech waveform without any consideration to the statistics of the signal.

Sparse approximation problems request a good approximation of an input signal $\mathbf{x} \in \mathbb{R}^M$ as a linear combination of elementary signals "atoms" belonging to a dictionary $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$, yet they stipulate that the approximation may involve only a few of the elementary signals. This class of problems arises throughout applied mathematics, statistics, and electrical engineering. One of the main tasks of the sparse approximation in the digital signal processing is the signal compression which is the main objective of this work. As it is already known, for compression purposes, the data encoder consists of two main stages that work separately to minimize the number of bits allocated to each input sample. The first stage is called the lossy compression that ends with the quantization process, and the second stage is called the lossless compression and ends with the entropy encoding. The sparse modeling is a lossy compression that precedes the quantizer and works as a transform encoder which transforms the signal from its time domain to the atoms domain.

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. Zhihong Man.

* Corresponding author.
E-mail address: ahmed_omara@eri.sci.eg (A.N. Omara).

http://dx.doi.org/10.1016/j.compeleceng.2017.05.005

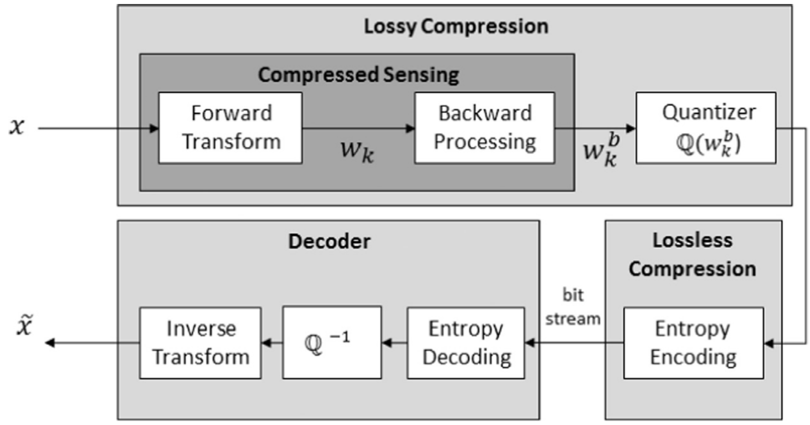

Fig. 1. Compressed sensing-based coding system.

As illustrated in Fig. 1, the sparse decomposition process may include two procedures, namely forward modeling and backward processing. The main procedure is the forward modeling, and there are several Forward Greedy (FG) algorithms such as Matching Pursuit (MP)[7], Orthogonal MP (OMP)[8], Optimized OMP (OOMP)[9], Stagewise OMP (StOMP)[10], MULIT-STAGE OMP [11], Compressive Sampling MP (CoSaMP)[12]; etc. By looking at the difference between some FG algorithms, we find that, they compete each other along two main fronts residual error minimization and complexity reduction. For OMP and OOMP, the residual error is minimized per forward iteration such that the resulting error vector is orthogonal to the selected subspace of atoms, and this orthogonalization process increases the complexity of the FG algorithm [13]. Unlike OMP and OOMP, the MP is fast but its residual error is orthogonalized only on the last selected atom which means a weak convergence [13]. Unlike both OMP and MP, the StOMP and MULIT-STAGE OMP are more fast algorithms because they select more than one atom per forward iteration and it is also resulting in a weak convergence.

All FG algorithms try to get a good approximation using only k atoms such that $k < M$ by solving the following objective function.

$$min \ \|\mathbf{x} - \mathbf{\Phi}\mathbf{w}\|_2^2 \qquad s.t. \qquad \|\mathbf{w}\|_0 = k \tag{1}$$

The solution of (1) is a vector of k nonzero weights or $\mathbf{w}_k$ and the obtained approximation is $\mathbf{\Phi}\mathbf{w}_k$. The second procedure in the compressed sensing stage (see Fig. 1) is called the backward processing. This task handles the vector $\mathbf{w}_k$ to increase the sparsity level by solving the following objective function.

$$min \ \|\mathbf{w}_k\|_0 \qquad s.t. \qquad \|\mathbf{x} - \mathbf{\Phi}\mathbf{w}_k\|_2^2 \le \alpha \|\vec{\mathbf{e}}_{k-1}\|_2^2 \tag{2}$$

Where α is a real number less than or equal 1 that guarantees that the backward processing doesn't eliminate the overall gain obtained by the last forward step, and $\vec{\mathbf{e}}_{k-1}$ is the residual vector at the forward step of order $k-1$. All the existing backward processing techniques such as Backward OOMP (BOOMP) [14], Forward-Backward greedy algorithm (FoBa) [15], Forward-Backward Pursuit (FBP) [16] and Backward Greedy Algorithm (BGA) [17] are based on the fact that the selected atoms by any FG algorithm can be reduced by eliminating some of them and the elimination error can be compensated if and only if the atoms are not orthogonal to each other. Both FoBa and FBP have built-in OMP-based FG algorithm such that after each forward iteration there is also a backward iteration. Unlike FoBa and FBP, both BOOMP and BGA are pure backward processing that works after finishing the forward modeling. After the backward processing, b atoms are eliminated from $\mathbf{w}_k$ to be $\mathbf{w}_k^b$, hence, the output should be quantized by a quantizer and the obtained vector becomes $\mathbb{Q}(\mathbf{w}_k^b)$, where $\mathbb{Q}(.)$ denotes the quantizer operator.

This work focused on the backward processing and its impact on the signal compression. From the point of view of sparsity representation, the efficiency of the Backward Elimination (BE) algorithms is directly proportional to the atoms' correlations. So, they fail in case of the orthogonality and quasi-orthogonality. Also, their efficiency is inversely proportional to the recovery performance of the FG algorithm, and this backs to the fact that the FG algorithm tries to select the smallest independent set of atoms during the forward iterations. On the side of signal compression, it is expected that the existing backward processing would not introduce any significant impact due to the strong dependency on the atoms' correlations. The main contribution of this paper is introducing a new backward technique so-called the Backward Replacement (BRe) that takes into consideration the impact of backward processing on the signal compression. The new technique doesn't exploit the correlations to eliminate the weights, but it exploits the converged weights to replace the sparse vector $\mathbf{w}_k$ with a sparse symmetric matrix which could be encoded efficiently. The rest of this paper is organized as follows: Section 2 lists some notational symbols. Section 3 discusses briefly the impact of greedy pursuit algorithms on the signal compression. Our approach is discussed in details in Section 4 including error, complexity and rate analysis . Section 5 presents the experimental results to demonstrate the effectiveness of the proposed approach. Finally, conclusions are summarized briefly in Section 6.