CLOSED-FORM MMSE ESTIMATOR FOR DENOISING SIGNALS UNDER SPARSE REPRESENTATION MODELLING

Matan Protter, Irad Yavneh, and Michael Elad

The Computer Science Department – The Technion, Haifa 32000, Israel

ABSTRACT

This paper deals with the signal denoising problem, assuming a prior based on a sparse representation with respect to a unitary dictionary. It is well known that the Maximum A-posteriori Probability (MAP) estimator in such a case has a closed-form solution based on shrinkage. The focus in this paper is on the better performing and less familiar Minimum-Mean-Squared-Error (MMSE) estimator. We show that this estimator also leads also to a simple closed-form formula, in the form of a plain recursive expression for evaluating the contribution of every atom in the solution. We demonstrate this formula, and compare it to the MAP and the Random-OMP method devised for approximating the MMSE result.

Index Terms— Sparse representations, MAP, MMSE, Unitary dictionary

1. INTRODUCTION

One of the most fundamental and extensively studied problem in signal processing in the removal of additive noise, known as denoising. In this task, it is assumed that the measured signal $\mathbf{y} \in \mathbb{R}^n$ is the result of a clean signal $\mathbf{x} \in \mathbb{R}^n$ being contaminated by noise, $\mathbf{y} = \mathbf{x} + \mathbf{v}$. As in many other works, we limit the discussion to zero-mean i.i.d Gaussian noise, with each entry being drawn according to the Normal distribution $\mathcal{N}(0, \sigma^2)$, with known variance σ.

In order to be able to differentiate the signal from the noise, is is important to characterize the signal family as well. A very successful model is one that leans on the signal's sparsity with respect to some transform. In such a model, the signal is assumed to be created as a linear combination of a few basic signal building blocks, known as *atoms*. Formally put, $\mathbf{x}$ can be represented as $\mathbf{x} = \mathbf{D}\alpha$, where $\mathbf{D} \in \mathbb{R}^{n \times m}$ is a known dictionary (set of atoms) and α being a *sparse* vector of coefficients. Sparsity here implies that α contains a small number ($\ll n$) of non-zero coefficients. In general, the dictionary may be redundant, containing more atoms than the dimension of the signal ($m \geq n$).

This research was supported by the Israel Science Foundation grant No. 599/08 and by the United States Israel Binational Science Foundation grant No. 2004199

How can this model be used for recovering $\mathbf{x}$ from the measurement $\mathbf{y}$? A commonly used method is seeking a signal $\hat{x}$ that is both sparse (i.e. has a sparse representation) and close enough to the measured signal. This task can be written as seeking the representation $\hat{\alpha}$ by

$$\hat{\alpha} = \arg\min_{\alpha} \ \|\alpha\|_0 + \lambda \|\mathbf{y} - \mathbf{D}\alpha\|_2^2. \tag{1}$$

This energy function contains two forces, the first promoting sparsity of the signal (where $\|\alpha\|_0$ counts the number of non-zeros in α) and the second requires proximity to the measurement. This minimization task can be shown to emerge as the Maximum A-posteriori Probability (MAP) estimator.

Solving the minimization task is in general NP-hard [1], and therefore approximate solvers are required. A common such solver, which we will be focusing on here, is a greedy algorithm known as the Orthogonal Matching Pursuit (OMP) [2]. In this algorithm, one atom is selected at each step, such that the norm of the residual (that portion of the signal not yet represented) is best decreased.

In this paper we focus on the special case where the dictionary $\mathbf{D}$ is unitary $\mathbf{D}^T\mathbf{D} = \mathbf{I}$ (it also means that $m = n$). In such a case, the problem formed in Equation (1) need not be approximated, as there is a closed-form non-iterative solution for it [3]. Furthermore, the OMP is known to be exact in such a case. Needless to say, these facts make the MAP estimator a very appealing approach for the unitary case.

While MAP estimation promotes seeking the single sparsest representation, recent work shows that a better result is possible using the Minimum-Mean-Squared-Error (MMSE) estimator [4, 5, 6]. The MMSE estimator requires a weighted average of all the possible sparse representations that may explain the signal, with weights related to the probability of these solutions. Just as the MAP in the general setting, this estimation is impossible to compute, and thus approximation is proposed. For example, the work reported in [6] suggests a random version of the OMP for getting several representations, and plain averaging of them for getting the final result.

The question we focus on in this paper is the following: When dealing with a unitary dictionary, could the MMSE estimator also enjoy a simple and closed form solution? We show that this is indeed the case, presenting a recursive formula for computing this estimation. We develop this formula, and then

demonstrate its superior performance, when compared to the random-OMP approximate estimator proposed in [6], and the MAP solution.

The structure of the paper is as follows: In the next section we formulate the denoising problem, and review the prior work on the MAP and the MMSE estimators for the denoising task described above. Section 3 then re-develops the MMSE estimation for the unitary case, getting a closed-form recursive formula. Section 4 is dedicated to empirical study of the various estimators discussed, and in Section 5 we conclude this paper.

2. PRIOR WORK

In order to deploy the MAP and MMSE estimators for the denoising task, we need to start by better defining the signal creation process. The information provided in this section follows the work in [6].

We assume that $\mathbf{x} = \mathbf{D}\alpha$ is generated by first choosing the support of α (locations of non-zero coefficients), denoted by S, using the probability function $p(S)$. Following [6] we shall restrict our treatment for now to the case where all the supports $|S| = k$ are equally probable, all the rest having zero probability. We denote the permissable supports by Ω_k. Once S is chosen, the representation's non-zeros are formed as a set of $|S|$ random iid entries drawn from the Normal distribution $\mathcal{N}(0, \sigma_x^2)$.

We define the operator $\mathbf{P}_S$ as the matrix that multiplies a sparse vector $\alpha \in \mathbb{R}^m$ with support S, and extracts only the non-zeros by their order to a vector of length $|S|$. This is a matrix of size $|S| \times m$, and we denote $\mathbf{D}_S = \mathbf{D}\mathbf{P}_S^T$ as the sub-matrix that contains only the columns referring to the support S from $\mathbf{D}$.

For this signal model, if the support S is known, the best estimator for $\mathbf{x}$ (termed *oracle*) is given by

$$\hat{\mathbf{x}}_{oracle} = c^2\mathbf{D}_S(\mathbf{D}_S^T\mathbf{D}_S)^{-1}\mathbf{D}_S^T\mathbf{y} = c^2\mathbf{y}_S. \tag{2}$$

This is a simple projection of the measurement $\mathbf{y}$ onto the sub-dictionary of $\mathbf{D}$ built of the columns of the support. The coefficient c^2 stands for $c^2 = \sigma_x^2/(\sigma_x^2+\sigma^2)$, performing shrinkage.

As the support is random, the MMSE estimate is given by an expectation over all possibilities,

$$\hat{\mathbf{x}}_{MMSE} = c^2 \sum_{S\in\Omega_k} p(S|\mathbf{y})\mathbf{y}_S. \tag{3}$$

This is a weighted average of many such oracles, each standing for a possible support, and each weighted by the probability of this support to explain $\mathbf{y}$. The term $p(S|\mathbf{y})$ is given by

$$p(S|\mathbf{y}) \propto \exp\left\{\frac{c^2\|\mathbf{y}_S\|^2}{2\sigma^2}\right\}, \tag{4}$$

up to a normalization factor. This expression suggests that the higher the energy remaining in the projection of $\mathbf{y}$ onto the k-dimensional subspace of S, the more probable this support is.

The MAP estimator chooses the support S that maximizes the above probability, $p(S|\mathbf{y})$, and computing the *oracle* estimate for this support. Both this estimate and the MMSE require a sweep through all supports in Ω_k, which is an impossible task in general, due to the exponentially growing size of this set as a function of the number of atoms m. Thus, OMP is used to approximate the MAP by solving an exact MAP estimator for $k = 1$ (one atom), peeling the found portion of the signal, and repeating the process. Similarly, the MMSE is approximated by the Random-OMP by repeating the OMP several times, with a random choice of the next atom, based on $p(S|\mathbf{y})$ for $k = 1$. This stands as an approximate Gibbs sampler of this distribution, and thus plain averaging of the found representations leads to a good approximation of the MMSE.

In the unitary case, any subset of columns from $\mathbf{D}$ are orthogonal, and thus the best support S that maximizes $\|\mathbf{y}_S\|^2$ is simply found by computing $\mathbf{D}^T\mathbf{y}$, sorting the result by (absolute) size, and choosing the first k entries. Thus, MAP for this case can be computed exactly. Furthermore, OMP in such a case is also exact, as the sequential detection of the largest inner product leads to the same result. Naturally, we should wonder whether the unitary case installs such simple and closed-form solution that bypasses the need for the Random-OMP. This is the topic of the next Section.

3. UNITARY DICTIONARY

3.1. Closed-Form MMSE Formula

The MMSE estimate, as described in Equation (3), can be read differently. Every possible support in the summation provides a candidate sparse representation vector $\hat{\alpha}_S = \mathbf{P}_S^T(\mathbf{D}_S^T\mathbf{D}_S)^{-1}\mathbf{D}_S^T\mathbf{y}$, each having k non-zeros in locations defined by the support S. Thus, the MMSE estimator is given by

$$\hat{\mathbf{x}}_{MMSE} = c^2\mathbf{D}\sum_{S\in\Omega_k} p(S|\mathbf{y})\hat{\alpha}_S. \tag{5}$$

This expression suggests that there is one effective representation that governs the estimated outcome, given as (we simply remove the multiplication by $\mathbf{D}$):

$$\hat{\alpha}_{MMSE} = c^2 \sum_{S\in\Omega_k} p(S|\mathbf{y})\hat{\alpha}_S. \tag{6}$$

This implies that every atom contributes a pre-specified portion of the overall MMSE estimator. We shall construct a formula for this contribution, thus turning this estimator into a practical algorithm.

Returning to Equation (4), we observe that in the unitary case $\mathbf{y}_S = \mathbf{D}_S\mathbf{D}_S^T\mathbf{y}$ (since $\mathbf{D}_S^T\mathbf{D}_S = \mathbf{I}$). Denoting the i-th entry in $\mathbf{D}^T\mathbf{y}$ as α_i, we get that

$$\mathbf{y}_S = \sum_{i\in S}\alpha_i\mathbf{d}_i \;\;\Rightarrow\;\; \|\mathbf{y}_S\|^2 = \sum_{i\in S}\alpha_i^2. \tag{7}$$

where $\mathbf{d}_i$ is the i-th column from $\mathbf{D}$. Plugging this in Equation (4) leads to

$$p(S|\mathbf{y}) \propto \exp\left\{\frac{c^2\|\mathbf{y}_S\|^2}{2\sigma^2}\right\} = \prod_{i\in S}\exp\left\{\frac{c^2\alpha_i^2}{2\sigma^2}\right\}. \tag{8}$$

We shall denote hereafter $q_i = \exp(c^2\alpha_i^2/2\sigma^2)$, and thus $p(S|\mathbf{y}) \propto \prod_{i\in S} q_i$. Using this notation, the MMSE estimator can be re-written as

$$\hat{\mathbf{x}}_{MMSE} = c^2\sum_{S\in\Omega_k} p(S|\mathbf{y})\mathbf{y}_S = \sum_{S\in\Omega_k}\prod_{i\in S} q_i \sum_{i\in S}\alpha_i\mathbf{d}_i. \tag{9}$$

Computing this formula in a straight-forward manner is exponential ($\approx m^k$), as every group of $k = |S|$ atoms has to be considered and summed. Rearranging the order of summations and multiplications in Equation (9) yields an equivalent expression for the MMSE estimator,

$$\hat{\mathbf{x}}_{MMSE} = \sum_{i=1}^{m}\left(\left(\sum_{Ss.t.i\in S}\prod_{j\in S} q_j\right)\alpha_i\mathbf{d}_i\right). \tag{10}$$

For simplicity, we denote $Q_i^k = \sum_{S_i^k}\prod_{j\in S_i^k} q_j$ where we used S_i^k to denote a group of size k containing q_i. Using this notation, the MMSE estimator can be written as $\hat{\mathbf{x}}_{MMSE} = \mathbf{D}diag(\mathbf{Q})\alpha$. While the direct computation of $\mathbf{Q}$ is exponential in k, a recursive formula for computing $\mathbf{Q}$ is within reach. Obviously, $\mathbf{Q}^1 = \mathbf{q}$. $\mathbf{Q}^2$ can be obtained from $\mathbf{Q}^1$ by

$$\begin{aligned} Q_i^2 &= \sum_{j\neq i} q_iq_j = q_i\cdot\sum_{j\neq i} q_j = q_i\cdot\sum_{j\neq i} Q_j^1 \qquad (11)\\ &= q_i\cdot\left(\sum_{j=1}^{m} Q_j^1 - Q_i^1\right). \end{aligned}$$

Similarly, any $\mathbf{Q}^k$ can be computed from its predecessor $\mathbf{Q}^{k-1}$ using

$$\begin{aligned} Q_i^k &= \sum_{S_i^k}\prod_{j\in S^k} q_j = q_i\cdot\sum_{S^{k-1}s.t.i\notin S^{k-1}}\prod_{j\in S^{k-1}} q_j\\ &= q_i\cdot\left(\sum_{S^{k-1}}\prod_{j\in S^{k-1}} q_j - \sum_{S_i^{k-1}}\prod_{j\in S^{k-1}} q_j\right). \qquad (12) \end{aligned}$$

The second term in the brackets is exactly the definition of Q_i^{k-1}. The first part can also be simplified by

$$\begin{aligned} \sum_{S^{k-1}}\prod_{j\in S^{k-1}} q_j &= \frac{1}{k-1}\sum_{l=1}^{m}\sum_{S_l^{k-1}}\prod_{j\in S^{k-1}} q_j \qquad (13)\\ &= \frac{1}{k-1}\sum_{l=1}^{m} Q_l^{k-1}. \end{aligned}$$

with the division by $(k-1)$ due to each group of $(k-1)$ appearing $(k-1)$ times, once for the summation for each of its members. Plugging (13) back into (12) we get the final recursive formula for Q_i^k :

$$Q_i^k = q_i\cdot\left(\frac{1}{k-1}\sum_{l=1}^{m} Q_l^{k-1} - Q_i^{k-1}\right). \tag{14}$$

3.2. Numerical Instability

The recursive formula obtained above suffers from numerical instability due to errors growing vary rapidly in the recursive iterations. In order to illustrate this, we consider a toy example, where a similar recursive formula is given as

$$x_{k+1} = (1-x_k)\frac{k^2}{k+1}, \quad k = 2,3,\ldots, \tag{15}$$

We show how numerical errors develop in this simpler formula. To our aid comes the knowledge of an exact solution for each element in the series $x_k = 1 - \frac{1}{k}, \quad k = 2,3,\ldots$ (this is easy to verify). Now, suppose that at the k^{th} step our calculation is effected by a small numerical error ϵ_k and thus, $\tilde{x}_k = x_k + \epsilon_k$. Even if the calculation at this stage is done exactly, the error at the next step becomes

$$\begin{aligned} \epsilon_{k+1} &= \tilde{x}_{k+1} - x_{k+1} = (1-\tilde{x}_k)\frac{k^2}{k+1} - x_{k+1} \qquad (16)\\ &= (1-x_k-\epsilon_k)\frac{k^2}{k+1} - x_{k+1} = -\epsilon_k\frac{k^2}{k+1}, \end{aligned}$$

which means that the error grows by $\frac{k^2}{k+1}$ at the $(k+1)^{th}$ iteration, implying an overall growth proportional to $(k-1)!$. This blow of the error is very similar to the one we obtain in the MMSE recursive formula, and it may cause havoc in applications (especially for large k).

Stability can be gained by enforcing some known constraints on the recursive formula outcome, so as to force the errors to remain small and controlled. The following constraints are straightforward to verify (while normalizing the series such that $\sum_i Q_i^k = k$):

1. Limited domain: $1 \geq Q_i^k \geq 0$
2. Monotonicity in k: $Q_i^k \geq Q_i^{k-1}$
3. Preservation of order: if $q_i > q_j$ than $Q_i^k > Q_j^k$.

Enforcing these constraints at each iteration of the recursive formula is a relatively cheap method of keeping the numerical errors in control. Furthermore, if at stage k during the calculation of the formula it is determined that one (or more) probability $Q_i^k$is to be assigned a value of 1, it is then possible to fix its value for all the following iterations, since it will (effectively) be in any group of k or more buckets. To improve the numerical accuracy for the rest of the entries, it it then possible to recalculate Q^k for these entries, by starting over while ignoring the fixed ones, and running for only $(k-1)$ iterations.

3.3. Treating More General Cardinalities

In the above development we assumed that $|S| = k$ and fixed. However, this is usually not the case, and the more general case should consider a probability function $p(|S|)$ (assumed to be descending for obvious reasons). How does that fit into the above description?

For the MAP estimator, this is easily integrated. The selected cardinality is one that fulfills

$$\hat{k} = \arg\max_{k} \max_{|S|=k} P(S|\mathbf{y}) \tag{17}$$

or in other words, for each cardinality the probability of the best support is multiplied by the probability of the cardinality, and the cardinality that yields the maximal product is then chosen.

For the MMSE, we need to consider all cardinalities with their appropriate probabilities. Going back to Equation (9), this translates into

$$\hat{\mathbf{x}}_{MMSE} = c^2 \sum_{k} p(k) \sum_{S\in\Omega_k} p(S|\mathbf{y})\mathbf{y}_S \tag{18}$$

Can the close-form formula developed above still be used? The answer is yes, with only a slight modification. In the developed formula, there is a normalization step missing, that did not affect the computation as long as we had only one cardinality. This normalization requires division by the number of groups of size k that can be formed, which is given by $\frac{n!}{k!\cdot(n-k)!}$. This is required to "level the playing field" between different cardinalities, as otherwise if $k_1 < k_2 < \frac{n}{2}$, there are more combinations for $|S| = k_2$ than for $|S| = k_1$, and this will create a bias towards $|S| = k_2$. That makes the overall recursive formula

$$Q_i^k = \frac{p(k)}{p(k-1)} \cdot \frac{(k-1)!\,(n-k+1)!}{k!\,(n-k)!} \cdot q_i \cdot \left(\frac{1}{k-1} \sum_{m} Q_m^{k-1} - Q_i^{k-1} \right), \tag{19}$$

where $\frac{p(k)}{p(k-1)}$ introduces the probability of each cardinality, and $\frac{(k-1)!(n-k+1)!}{k!(n-k)!}$ performs the normalization due to the different number of groups available at each cardinality. Implementing this formula requires slightly more care in overcoming numerical instabilities, as the normalization suggested in the previous section need to be tracked and then undone in order for the relative weight of each cardinality to be considered appropriately (when only one cardinality was considered, this normalization was irrelevant). However, these modifications hardly effect the number of overall calculations compared to computing the MMSE for the maximal cardinality only.

4. EXPERIMENTAL RESULTS

We now proceed to demonstrate the superior performance of the exact MMSE estimator over its approximation, the Random-OMP, which in itself is superior to the MAP estimator. For this end, we harness synthetic experiments, in which we can control all the parameters of the signals and the noise.

We use $n = 64$ as the working dimension, and DCT as the unitary dictionary. First the support size k of the signal's representation is drawn at random from $p(|S|)$. For a given k, the support itself is drawn at random with uniform probability over all $\binom{n}{k}$ possibilities. The coefficients (α_i) are drawn independently for each atom in the support from a Normal distribution $\mathcal{N}(0, \sigma_x^2)$. The resulting sparse vector of coefficients is multiplied by the dictionary to obtain the *ground-truth* signal. Each entry is independently contaminated by white Gaussian noise $(\mathcal{N}(0, \sigma^2))$ to create the *input* signal (note that due to $\mathbf{D}$ being unitary, this is equivalent to contaminating the coefficients themselves with WGN with the same parameters).

The obtained noisy signal is denoised by several methods: (i) MAP using OMP, (ii) Random-OMP that approximates the MMSE [6]; (iii) An exact and exhaustive MMSE using Equation (3); (iv) The recursive MMSE formula; and (v) An oracle that knows the exact support (we use the numerical estimator, and a closed-form formula as well -see [6]). This process is repeated for 1000 signals, and the mean L_2 error is averaged over all signals to obtain an estimate of the expected quality of each estimator.

In order to test the performance of these estimators under different noise conditions, several such tests are run, with $\sigma_x = 1$ kept constant in all tests, while the noise level σ being changed between experiments (in the range $0.1 - 2$). This is sufficient, since the important parameter is the ratio σ_x/σ, and not their individual absolute values.

Figure 1 shows the denoising achieved by each method, when $P(|S|) = 1$ for $|S| = 3$ and 0 otherwise. This Figure also contains the results for the exhaustive MMSE, which is the straight forward (exponential) computation of the MMSE estimator. Figure 2 shows the same graph for $P(|S|) = 1$ for $|S| = 7$, with the exhaustive MMSE omitted due to the enormous number of calculations required. In Figure 3, we ran a very similar experiment, only this time the cardinality of the signals is drawn according to the probability of $P(|S| = k) = 0.8^k$ and normalized to sum to 1.

To demonstrate the gap between the different methods, we show in Figure 4 the effective representation achieved by each method (for $P(|S| = 3) = 1$. The MAP estimator selects the wrong atoms, due to the relatively strong noise ($\sigma_N = 0.6$). It is important to note that the Random-OMP and MMSE generally do not result in a sparse representation, but they are still better than the MAP even though the original signal is sparse.

5. SUMMARY

In this work we discuss the problem of denoising a signal known to have a sparse representation, considering the MAP and the MMSE estimators. We focus on unitary dictionaries,

for which we show that a closed-form, exact, and simple recursive formula exists for the MMSE estimator. This replaced the need for an approximation, such as the Random-OMP algorithm. We show experimentally that this exact MMSE formula out-performs the approximate Random-OMP. We also discuss several numerical issues raised when implementing this formula in practice. In a follow-up work, we intend to implement this estimator on images, with necessary modifications to the model generation.

6. REFERENCES

[1] B.K. Natarajan, Sparse approximate solutions to linear systems, *SIAM Journal on Computing*, 24:227–234, 1995.

[2] S. Mallat and Z. Zhang, Matching pursuits with time-frequency dictionaries, *IEEE Trans. on Signal Processing*, 41(12):3397–3415, 1993.

[3] M. Elad, Why simple shrinkage is still relevant for redundant representations?, *IEEE Trans. on Information Theory*, 52(12):5559–5569, 2006.

[4] E.G. Larsson, and Y. Selen, Linear gregression with a sparse parameter vector, *IEEE Trans. on Signal Proc.*, 55(2):451–460.

[5] P. Schnitter, L.C. Potter, and J. Ziniel, Fast Bayesian matching pursuit, Proc. Workshop on Information Theory and Applications (ITA), (La Jolla, CA), Jan. 2008.

[6] M. Elad and I. Yavneh, A weighted average of sparse representations is better than the sparsest one alone, submitted to *IEEE Trans. on Information Theory*.

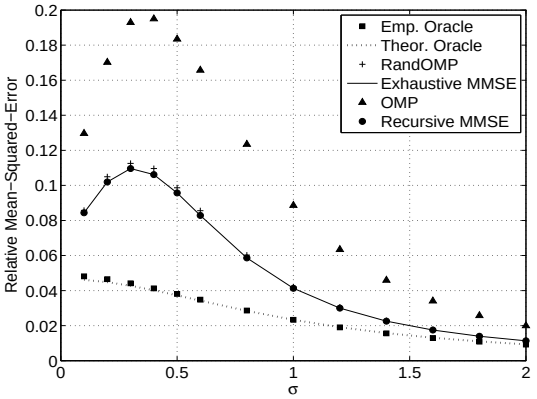

Fig. 1. Denoising achieved (averaged on 1000 signals) by several methods, for different noise strengthes for $|S| = 3$.

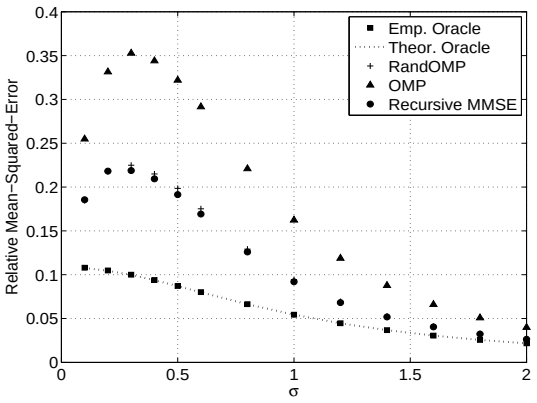

Fig. 2. Denoising achieved (averaged on 1000 signals) by several methods, for different noise strengthes for $|S| = 7$.

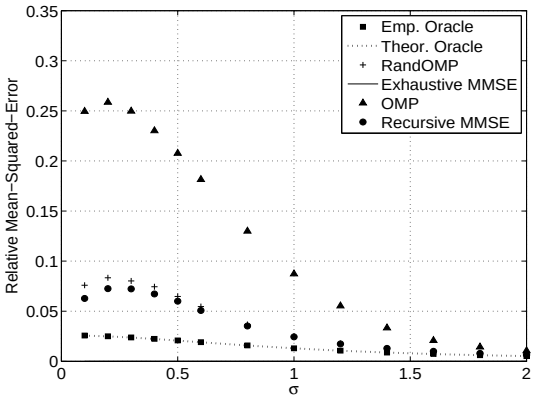

Fig. 3. Denoising achieved (averaged on 1000 signals) by several methods, for different noise strengthes for $P(|S| = k) = 0.8^k$ for $k = 1 - 4$.

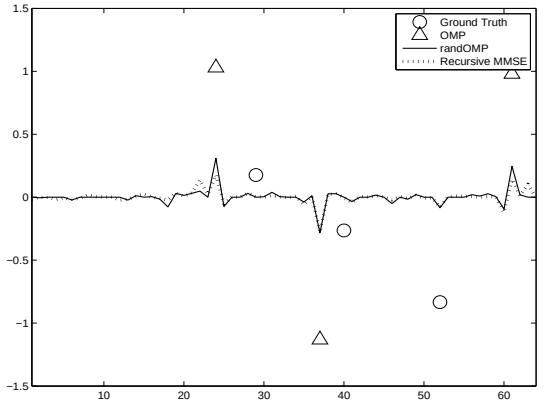

Fig. 4. The effective representation achieved by different methods for one example signal, with noise $\sigma_N = 0.6$.