The Mirror Transform

Fabrizio Guerrini, Alessandro Gnutti, *Member, IEEE*, and Riccardo Leonardi, *Fellow, IEEE*

***Abstract*—This paper explains how to describe any finite-energy signal through a unique representation consisting of an ordered set of positions and a sparse set of signals. This is obtained by designing an iterative decomposition through a series of mirror operations around those positions. The purpose is to find at any step of the decomposition the location that provides for the maximum decoupling between the even and odd components of the signal with respect to it. By limiting such even and odd components on three separate information bearing support, the algorithm can be iterated at infinity determining a sequence of positions. The per location information determines the optimal energy decoupling strategy at each stage providing remarkable sparsity in the representation. The resulting transformation leads to a 1-to-1 mapping. The approach is easily extended to finite-energy sequences, and in particular for sequences of finite length N, at most N iterations of the decomposition are required. Thanks to the sparsity of the resulting representation, experimental simulations demonstrate superior approximation capabilities of this proposed non-linear Mirror Transform with potential application in many domains such as approximation and coding. Its implementation has been made publicly available.**

***Index Terms*—Transforms, non-linear approximation, signal decomposition, sparsity.**

I. Introduction

FOR analysis or communication tasks such as classification, machine learning, detection, estimation, and coding, signal decomposition is fundamental for the representation of information [1], [2], [3]. In 1822, Joseph Fourier first established that a periodic wave can be represented as a linear combination of harmonic components of its fundamental frequency [4]. In addition, the representation completeness converges at infinity, in the mean square error sense, to the representation of discontinuous waveforms. About a hundred years later in 1909, Alfred Haar showed the converse result that a continuous waveform in $\mathcal{L}^2(\mathbb{R})$ can be represented as an infinite series of discontinuous functions [5]. More generally, any separable Hilbert space of infinite dimensionality $\mathcal{H}$ can be represented using some infinite set of functions forming one of its bases. With the concept of frames, the decomposition may be overcomplete and the representation becomes non-unique. Consequently, Fourier decomposition and/or multiresolution representations [6] are at the foundation of waveform description and enable new forms of signal classification, data recovery, denoising, etc.

Lately, more emphasis has been devoted to the study of alternative linear expansions for representing any vector $\underline{x} \in \mathcal{H}$:

$$\underline{x} = \sum_{i \in \mathcal{I}} \alpha_i \underline{\varphi}_i \,, \ \forall \underline{x} \in \mathcal{H}, \tag{1}$$

Fabrizio Guerrini and Alessandro Gnutti are with the Department of Information Engineering, CNIT – University of Brescia, Italy.

Riccardo Leonardi has a joint appointment with the Department of Information Engineering, CNIT – University of Brescia, Italy, and the Department of Electronics Engineering, University of Rome Tor Vergata, Italy.

learning different dictionaries that would generate sparse representations while considering the typical statistical distributions of real phenomena leading to particular classes of signals (such as natural images, individual ratings, population health, etc.). This has given us the possibility to construct effective alternative sparse expansions of waveforms [6], [7], with further advancements for compressive sensing, denoising, super-resolution, data recovery, and many more applications.

When the size of such dictionaries is larger than the signal dimensionality, there is an infinite number of solutions to decompose $\underline{x}$. The problem to find the sparsest solution to Eq. (1), that is the solution with the fewest number of atoms $\underline{\varphi}_i$, is generally non convex and NP-hard. Thus, only approximate solutions can be found for the general case. Among these, greedy algorithms and relaxing the non convex sparsity cost by its tight convex surrogate have been proposed (see, *e.g.*, the recently proposed atomic norm minimization [8]).

In a broader perspective, there have been attempts to model signal families as lying on manifolds. These approaches try to follow signal geometrical features [9]. This can be framed in the general context of dimensionality reduction, where signals laying in high dimensional spaces are approximated as a set of lower dimensional subspaces. Instead of a representation through a series of projections on a (possibly overcomplete) set or through a sparse expansion on an accurately designed dictionary, dimensionality reduction does not usually achieve completeness, since the signal can only be approximated with no guarantee on the representation error convergence. In addition, there is no general solution to the inverse problem of generating a signal given a manifold, so such problems usually need a regularization approach (see, *e.g.*, [10]).

Another attempt of sharing an approximate signal representation paradigm is based on the use of contractive transformations. Iterative Function Systems (IFS) try to describe a signal through a set of base signals and a pool of contractive transformations, so that their iteration converges to an approximate fixed point. Where IFS are quite attractive from a signal generation perspective, when the set of transformations is estimated for signal analysis purposes, the solution to the inverse problem is difficult, and there exist only bounds on the approximation error to the original signal [11].

Another approach worth mentioning in the family of data-adaptive iterative methods is the Empirical Mode Decomposition (EMD) [12], which is a multi-resolution technique which decomposes a signal into physically meaningful components, known as Intrinsic Mode Functions (IMFs), which are signals at the same time scale as the original signal. While this peculiarity makes such components easier to analyze from an empirical perspective, such decomposition lacks sparsity properties, since the representation is given by a certain number of signals with the same size of the original one, thus leading

to an increase of the overall temporal information bearing support. Furthermore, the IMFs do not form an orthogonal set, thus there is no simple way to assess the introduced distortion when considering only a subset of the IMFs for reconstruction.

In this paper, we want to preserve the idea of a converging representation where any signal $\underline{x}$ may be exactly recovered. Differently with respect to most traditional approaches, this representation will not come from a projection on the space (or subspaces) spanned by predefined set(s) of vectors. Instead, each signal is described through an iterative decomposition into a possibly infinite and unique set of orthogonal components, that can be recombined through a chain of summations and mirroring operations. In contrast with IFS or dimensionality reduction methods, the proposed approach is generative, which means that the signal is exactly represented, and an iterative transformation applied at each step is not found by solving an inverse problem, but rather it is constructively generated from intrinsic characteristics of the considered signal.

To achieve energy compaction a mirroring location is selected at each stage of the proposed iterative transform so as to provide maximal energy concentration into one of two components of an additive decomposition. By iterating the decomposition, an excellent approximation of the original signal can be extracted by using just a few components, thus becoming intrinsically sparse. By truncating the decomposition at some level, or by recombining only the constituent components with the highest energy, the reconstructed signal will be very close to the original one. In this paper, it is shown that a much more sparse representation can be generated if compared to any linear expansion, thus leading to a better approximation (in the $\mathcal{L}^2$ sense) of the original waveform.

The peculiar nature of the proposed transform also allows the display of a number of useful and distinctive properties besides sparsity, which could turn out beneficial to address many signal processing problems. For example, since the iterative decomposition process at the core of the transform is signal dependent, the transform structure itself represents an alternative representation that is tied to the generating signal through a one-way function.

The rest of the paper is organized as follows. Sec. II reviews some background while the basic even-odd decomposition is generalized. Then, the Mirror Transform (MT) is formally introduced in Sec. III, as the even-odd decomposition is generalized to achieve maximum energy decoupling and the process is iterated to form a decomposition tree. A few transform properties that can be derived from this novel representation paradigm are given in Sec. IV. The extension of the MT to the discrete-time domain is discussed in Sec. V. Then, Sec. VI provides more insights on the peculiar characteristics of the MT through a series of experiments, mainly allowing to verify its sparsity and one-wayness properties in the case of discrete-time sequences. Conclusions are finally drawn in Sec. VII. In addition, an implementation of the proposed transform and the relevant code to run experiments are publicly available [13].

II. Background

The objective of this section is to provide some background on the processing steps that are used throughout the rest of the paper. We start by briefly recalling the well-known even-odd decomposition for continuous-time signals (Sec. II-A). It is widely recognised that a signal exhibiting either an odd or even symmetry around its origin optimally decouples the energy between the two components. In order to exploit this property more generally, the decomposition process is redefined in Sec. II-B, so as to identify an arbitrary mirroring position.

A. Even-odd decomposition

The even-odd (or parity) decomposition of a finite-energy signal, denoted as $x(t) \in \mathcal{L}^2(\mathbb{R})$, states that $x(t)$ can be expressed as the sum of its even and odd parts, respectively $x_e(t)$ and $x_o(t)$, given by:

$$x_e(t) = \frac{x(t) + x(-t)}{2}; \quad x_o(t) = \frac{x(t) - x(-t)}{2}, \tag{2}$$

and $x(t) = x_e(t) + x_o(t)$. For the even signal $x_e(t) = x_e(-t)$; for the odd signal $x_o(t) = -x_o(-t)$. Since $\mathcal{L}^2(\mathbb{R})$ is a Hilbert space, with inner product $< x(t), y(t) >= \int_{\mathbb{R}} x(t) y^*(t) dt$, such decomposition is possible $\forall\, x(t)$. Therefore, since the inner product $< x_e(t), x_o(t) >$ is 0, $x(t)$ can be expressed as the sum of 2 orthogonal vectors. Defining the energy E as the squared Euclidean norm of the signal $x(t)$, and since $x_e(t) \perp x_o(t)$ with respective energy E_e and E_o, it is easy to verify that:

$$\begin{aligned} E &= \int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |x_e(t) + x_o(t)|^2 dt = \\ &= \int_{-\infty}^{\infty} |x_e(t)|^2 dt + \int_{-\infty}^{\infty} |x_o(t)|^2 dt = E_e + E_o. \end{aligned} \tag{3}$$

Let us assume without loss of generality that the original signal has finite temporal support, say $[-T, T]$ (which falls under the general formulation if one admits to zero-pad the signal). An example of an even-odd decomposition of a real signal is shown in Fig. 1 for $T = 1$. The pair of signals $x_e(t)$ and $x_o(t)$ obtained through the above decomposition step constitutes an alternative representation of $x(t)$, since the decomposition is unique and both signals are needed to reconstruct $x(t)$. The intuitive importance of such representation stems from the fact that whenever $x(t)$ has an approximate even/odd character (even when it is not perfectly symmetric), most of its energy will be carried by one of the constituent components. As such, the even-odd decomposition carries the possible symmetric nature of $x(t)$ around $t = 0$. As a matter of fact, when performed on a limited support, it can be used to track local reflective symmetries, as shown in [14].

Following Eq. (2), both signals $x_e(t)$ and $x_o(t)$ have the same support $[-T, T]$ of the original signal. However, even if two signals are needed to represent $x(t)$, the new representation does not double the information necessary to reconstruct the original signal. In fact, the parity property of the even and odd parts implies that only either their causal part (respectively $x_e^{(c)}(t) = x_e(t) \cdot \mathbb{1}(t \geq 0)$ and $x_o^{(c)}(t) = x_o(t) \cdot \mathbb{1}(t \geq 0)$, both with support $[0, T]$) or anticausal part (respectively $x_e^{(ac)}(t) = x_e(t) - x_e^{(c)}(t)$ and $x_o^{(ac)}(t) = x_o(t) - x_o^{(c)}(t)$, both with support $[-T, 0]$) is informative and thus sufficient to describe the entire signal. In other words, given only, *e.g.*, the causal

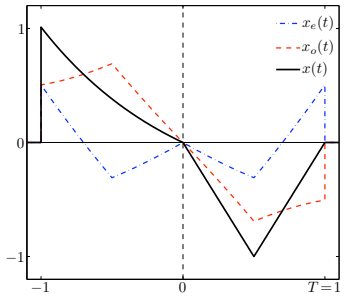


Figure 1: The original signal $x(t)$ (black) is decomposed into $x_e(t)$ (blue) and $x_o(t)$ (red), according to Eq. (2).

part of both signals, the anticausal part can be readily obtained by mirroring the causal one:

$$x_e(t) = x_e^{(c)}(t) + x_e^{(c)}(-t); \quad x_o(t) = x_o^{(c)}(t) - x_o^{(c)}(-t). \quad (4)$$

Note that, to keep Eq. (4) simple, we ignore the fact that the isolated value in $t=0$ needs to be handled separately, which is not an actual concern for well-behaved signals.

The original signal $x(t)$ is then reconstructed summing $x_e(t)$ and $x_o(t)$ as in Eq. (2). By retaining just the causal part of the even and odd parts, their energy is halved, respectively $E_e/2$ and $E_o/2$, and their sum gives $E/2$.

B. Generalized even-odd decomposition

It can be noted that there are some cases where the parity decomposition as defined in Eq. (2) has not the intended effect of separating the underlying symmetric signal, even if the original signal has simple parity characteristics, only because its center of symmetry is not in the time origin. Consider for example a rectangular impulse $x(t) = \mathbb{1}(4 < t < 5)$. Though the indicator function $\mathbb{1}(\cdot)$ applied on an interval, like the given $x(t)$, is a perfectly even signal with respect to its midpoint, if $x(t)$ is decomposed along the lines of Eq. (2), its support would be $[-5, 5]$, and therefore the decomposition would lead to the signals $x_e(t) = (\mathbb{1}(4 < t < 5) + \mathbb{1}(-5 < t < -4))/2$ and $x_o(t) = (\mathbb{1}(4 < t < 5) - \mathbb{1}(-5 < t < -4))/2$. Since $x_e(t)$ and $x_o(t)$ have the same energy, there is no indication of parity by inspecting their energies.

The reason for this result is that the even-odd decomposition only considers the parity with respect to the decomposition support midpoint (namely, the time origin for centered supports). The even-odd decomposition with respect to an arbitrary flipping point $t=t_f$ can be chosen instead, provided that we slightly extend Eq. (2), which becomes:

$$x_e(t;t_f) = \frac{x(t)+x(2t_f-t)}{2}; \; x_o(t;t_f) = \frac{x(t)-x(2t_f-t)}{2}, \quad (5)$$

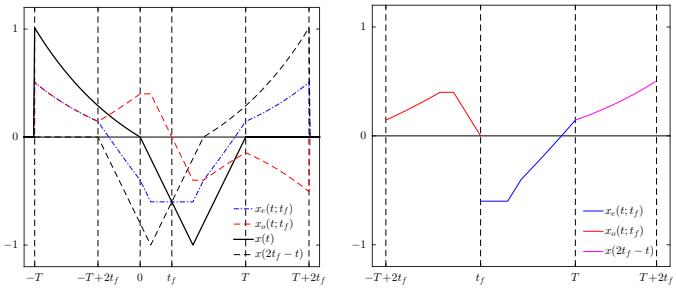
with $x(t) = x_e(t;t_f) + x_o(t;t_f)$. Eq. (5) turns into Eq. (2) when $t_f = 0$. In general, t_f is not the signal support midpoint.

Let us consider again that the original signal $x(t)$ has finite support $[-T, T]$. After the even-odd decomposition with respect to t_f with $|t_f| < T$, the support of both $x_e(t;t_f)$ and $x_o(t;t_f)$ is extended to $2T + 2|t_f|$. This evidence is shown in Fig. 2a, with $T=1$. In the figure, we have fixed an arbitrarily chosen $t_f = 0.3$. If t_f sits to the right of the signal support (*i.e.*, $t_f > 0$), when the signal $x(t)$ is mirrored around it, obtaining $x(2t_f - t)$, it can be observed that the latter signal now has support $[-T+2t_f, T+2t_f]$ (in the figure $[-0.4, 1.6]$). Thus, when $x(2t_f - t)$ is added or subtracted to $x(t)$ as in Eq. (5) the resulting signals have support which is (at most) the union of $[-T+2t_f, T+2t_f]$ and $[-T, T]$. If the symmetry point is to the right of the support center as in this example, the resulting support is $[-T, T + 2t_f]$, whereas it would be $[-T + 2t_f, T]$ if $t_f < 0$. Its extent is therefore $2T + 2|t_f|$.

Nevertheless, the information-bearing support remains $2T$ since the added support on one side simply mirrors the other end of the signal confined in such informative support. In particular, in both the even and odd parts, the $2|t_f|$ long tails correspond to the last or first part of the original signal, where one of them has the sign reversed for the odd part, depending respectively on whether $t_f < 0$ or $t_f > 0$. Looking again at Fig. 2a, the leftmost parts of $x_e(t;t_f)$ and $x_o(t;t_f)$, *i.e.*, in the interval $[-T, -T+2t_f]$, solely correspond to $x(t)/2$, while for the rightmost part, *i.e.*, in the interval $[T, T + 2t_f]$, the even signal is a mirrored copy of the former (which corresponds also to $x(2t_f - t)/2$), while the odd signal is its sign-reversed mirrored copy. Clearly, just one of these is actually enough to reconstruct $x(t)$. Moreover, each signal in the intervals $[-T + 2t_f, t_f]$ and $[t_f, T]$ enjoys the usual mirror symmetry, ensuring that only one of the supports is informative.

In conclusion, the representation still uses the same support $2T$ as the original signal, when only the informative parts are kept. Therefore, when the decomposition is implemented by Eq. (5), and the informative supports are separated during the decomposition along the lines discussed above, three signals can be considered. First, removing the tail from the even and odd signals $x_e(t)$ and $x_o(t)$ output of Eq. (5) (henceforth dropping the t_f parameter for convenience) produces a truncated even signal $\overline{x}_e(t)$ (the overline here indicates truncation by removing the tail), the (truncated) odd signal $\overline{x}_o(t)$ and what is referred to as the tail part, $x_t(t)$. The tail part, which is chosen as the even signal in the $[-T, -T + 2t_f]$ and $[T, T + 2t_f]$ intervals in Fig. 2a, is just a double copy of that $2|t_f|$ long part of the original signal which is not involved in the decomposition, that is the rightmost one if $t_f < 0$ or the leftmost if $t_f > 0$, and thus leads to a cumulative support extent of $4|t_f|$. For convenience we chose the causal part to retain the even signals and the anticausal part for the odd ones, to ensure that the informative even signal $\overline{x}_e^{(c)}(t)$ and informative odd signal $\overline{x}_o^{(ac)}(t)$ each have support $T - |t_f|$. Thus, the informative (causal) support for the even part is the $[t_f, T + 2t_f]$ interval for $t_f < 0$ or the $[t_f, T]$ interval for $t_f > 0$, and the converse is true for the odd part. The tail $x_t^{(c)}(t)$ is $x(t)/2$ in the $[T + 2t_f, T]$ interval if $t_f < 0$ or it is $x(2t_f - t)/2$ in the $[T, T + 2t_f]$ interval if $t_f > 0$, *i.e.*, a $2|t_f|$ support. The informative signals total support is thus $2T$. Fig. 2b depicts the result of such a ternary decomposition.

Eq. (3) for the computation of the energy distribution still



(a) The original, real signal $x(t)$ (black) is decomposed around the mirroring point $t_f = 0.3$ into the even and odd signals, $x_e(t)$ (blue) and $x_o(t)$ (red), according to Eq. (5). The signal $x(2t_f - t)$ is also depicted (dashed black) for convenience.

(b) The results of the decomposition: the causal part of the (truncated) even signal is $\overline{x}_e^{(c)}(t)$, the anticausal part of the (truncated) odd signal is $\overline{x}_o^{(ac)}(t)$, and the causal part of the tail signal is $x_t^{(c)}(t)$.

Figure 2: Ternary even-odd decomposition process of a signal.

applies. In particular, for the signal in Fig. 2a, it becomes:

$$\begin{aligned} E &= \int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-T}^{T+2t_f} |x_e(t) + x_o(t)|^2 dt = \\ &= \int_{-T+2t_f}^{T} |\overline{x}_e(t) + \overline{x}_o(t)|^2 dt + 2\int_{-T}^{-T+2t_f} |x_t(t)|^2 dt = \\ &= \int_{-T+2t_f}^{T} |\overline{x}_e(t)|^2 dt + \int_{-T+2t_f}^{T} |\overline{x}_o(t)|^2 dt + \ldots = \\ &= E_e + E_o = E_{\overline{e}} + E_{\overline{o}} + E_t, \end{aligned} \tag{6}$$

since $\overline{x}_e(t)$ and $\overline{x}_o(t)$ are still orthogonal, and $x_t(t)$ is on a distinct support. From the definition we have given above, it is clear that (the same) part of the energy of the even/odd signals moves from all truncated signal energies to the tail signal energy. This means that $E_t \leq \min\{E_e, E_o\} \leq E/2$. In addition, $E_t < \max\{E_{\overline{e}}, E_{\overline{o}}\}$, implying that most of the energy is carried by one of the truncated signals.

III. The Mirror Transform (MT)

To optimize the parity decomposition just discussed on the informative part of the produced signals, it is advisable to suitably choose t_f to allow for the best tracking of symmetries that might be present in the original signal: this is discussed in Sec. III-A. When iterated, a tree may be constructed by repeating the even-odd decomposition process for every component. The tree thus becomes *ternary*, since three signals remain as presented in Sec. II, that is, the decomposition must be repeated for each even, odd, and tail component. This tree represents what we refer to as the *Mirror Transform (MT)* of the signal, that we formally define in Sec. III-B.

A. Finding the optimal symmetry point

In Sec. II-B we have shown how to perform the even-odd decomposition around an arbitrary mirroring point t_f which is not the support midpoint, and divide the resulting signals according to their informative support. Our objective is now to determine whether an optimal symmetry point t_0 can be identified. If a signal is symmetric (or antisymmetric) with respect to a certain point t_0, performing the even-odd decomposition using $t_f = t_0$ as mirroring point would output the even and odd signals, $x_e(t)$ and $x_o(t)$, whose energies are in great disproportion. Therefore, it is natural to search for all t_f for which there is a maximum decoupling of the energies E_e and E_o associated to the even and odd parts, as expressed from Eq. (6) when t_f varies (see also [15]).

Since the energies E_e and E_o depend on t_f, to sum up to E both possess the same extreme points. The optimal symmetry point t_0 corresponds to the maximum of either E_o or E_e (*i.e.*, respectively the minimum of E_e or E_o). The search for t_0 is meaningless when applied to zero-energy signals.

To find t_0, let us therefore concentrate on the extrema of the energy of the even part. For a complex $x(t)$, we have:

$$\begin{aligned} E_e(t_f) &= \int_{-\infty}^{+\infty} |x_e(t;t_f)|^2 dt = \int_{-\infty}^{+\infty} \left| \frac{x(t) + x(2t_f - t)}{2} \right|^2 dt = \\ &= \frac{1}{4}\int_{-\infty}^{+\infty} \left[|x(t)|^2 + |x(2t_f - t)|^2 + 2\,\mathrm{Re}\{x(t)x^*(2t_f - t)\} \right] dt, \end{aligned} \tag{7}$$

where we can safely extend the integral on the whole real axis without affecting the result for finite-support signals. The first two terms in the last integral give E as a result, since reversing the time axis and shifting the origin do not influence the energy value, thus they are both independent from t_f. Hence:

$$E_e(t_f) = \frac{1}{2}E + \frac{1}{2}\int_{-\infty}^{+\infty} \mathrm{Re}\{x(t)x^*(2t_f - t)\} dt. \tag{8}$$

For energy signals, the linear convolution being defined as:

$$(x * y)(t) = \int_{-\infty}^{+\infty} x(\tau) y(t - \tau) d\tau, \tag{9}$$

we can simply write:

$$E_e(t_f) = \frac{1}{2}E + \frac{1}{2}\,\mathrm{Re}\{(x * x^*)\}(2t_f). \tag{10}$$

The energy of the even part is a function of t_f dictated by the convolution of the original signal with its complex conjugate – a "conjugate self-convolution" (which is equivalent to the cross-correlation between the signal and its mirrored version).

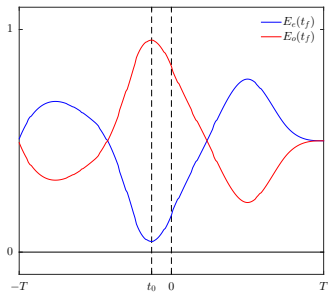
E_e admits at least one maximum, since a maximum exists for the convolution. This can be readily seen since it must be limited for finite energy signals, as implied by the Cauchy-Schwarz inequality, and on the other hand that the energy of the even part cannot be greater than that of the original signal. Furthermore, since the auto-convolution is an integral function it must be continuous, so E_e admits at least one extreme point.

Thus, the derivative of $E_e(t_f)$ with respect to t_f exists and it can be studied to find the position of its extreme points:

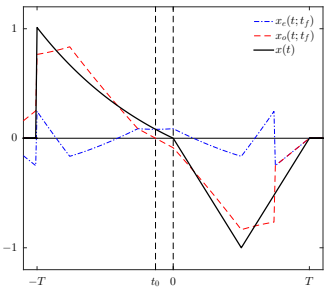
$$\begin{aligned} \frac{dE_e}{dt_f} &= \frac{d}{dt_f}\frac{1}{2}\,\mathrm{Re}\{(x * x^*)\}(2t_f) = \\ &= \frac{d}{dt_f}\,\mathrm{Re}\left\{ \int_{-\infty}^{+\infty} x(t)x^*(2t_f - t) dt \right\}. \end{aligned} \tag{11}$$

Under mild assumptions on the continuity of the signal $x(t)$ and its derivative, we can exchange the order of derivation and integration, and therefore we obtain:

$$\frac{dE_e}{dt_f} = \mathrm{Re}\{x * (dx^*/dt)\}(2t_f) = -\frac{dE_o}{dt_f}. \tag{12}$$



(a) The normalized energies of the even and odd parts of $x(t)$ of Fig. 1, function of the mirroring point t_f.



(b) The result of the optimal parity decomposition of $x(t)$, using the optimal point t_0 as the mirroring point.

Figure 3: Revisited previous example using a variable mirroring point, showing how the energies of the even and odd parts vary as the point moves. The optimal decomposition point t_0 leads to a maximal energy of its associated odd part, which in this example is larger than the maximal energy with respect to any other decomposition location for the even part.

So, for continuous-time signals, candidate extreme points of the auto-convolution can be determined by convolving the signal with the derivative of its complex conjugate, finding its zero-crossing points, and dividing by 2 the found locations. A local minimum for E_e corresponds to a complementary local maximum for E_o and *vice-versa*. To find the global minimum between E_e and E_o we must consider both the global maximum for E_e as well as its global minimum (*i.e.*, associated to the global maximum of E_o). t_0 then corresponds to the location that leads to the largest value between them.

Fig. 3 shows the evolution of $E_e(t_f)$ and $E_o(t_f)$ (normalized by E) for the (real) signal shown in Fig. 1. In this case, $t_0 \approx -0.13$ yields the maximum value for E_o, which is larger than the maximum value of E_e. Accordingly this one is selected as the optimal symmetry point t_0. The prevalent odd nature of $x(t)$, even around $t=0$, is somewhat evident, but the optimization of t_f proves how considering a negative offset for the mirroring point guarantees further energy decoupling: in fact, E_o increases from around 85% of E for $t_f = 0$ to more than 95% for $t_f = t_0$. Observe also how the two local maxima of E_e are able to capture the two milder even symmetries.

For arbitrary signals with no analytical expression there is no closed-form formula to identify the optimal locations. We need to resort to numerical computation. For each local maximum, we need to find which one leads to the absolute maximum of the auto-convolution (recall that the optimal symmetry location is found by dividing by 2 the location of the auto-convolution). Note that the maximum may not be unique. For a simple example, consider $x(t) = \mathbb{1}(-1 < t < 1) + \mathbb{1}(9 < t < 10) - \mathbb{1}(10 < t < 11)$ whose auto-convolution $(x * x)(t')$ has two maxima in $t' = 0$ and $t' = 9$ and two minima in $t' = 11$ and $t' = 20$ with the same absolute value. Of course, the two (respectively even and odd) symmetries of $x(t)$ centered at $t = 0$ and $t = 10$ are equivalent energy-wise. If such ties occur (rarely enough for real-world signals), we define by convention the optimal location: for example, we could arbitrarily choose the leftmost maximum, or the one nearest to the center of the signal support (in this toy example the latter solution would lead either to $t' = 9$ or $t' = 11$ whereas the former one would select $t' = 0$, but either strategy leads to the same optimal energy decoupling).

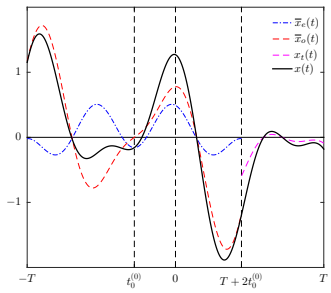
B. The Mirror Transform: optimal ternary decomposition tree

In order to construct a full energy-compacting decomposition of the original signal, we are going to iterate the optimal energy decoupling process on the relevant information bearing even and odd components, obtained through the previously described optimal decoupling step. The objective is thus to build a decomposition tree by iterating the optimal decomposition, so that an increasing portion of the signal energy is carried by an ever smaller temporal support at each step.

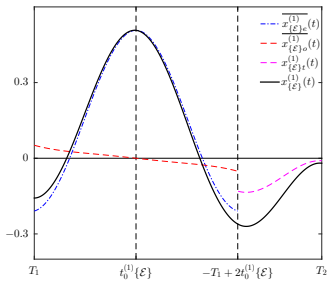
First, let us assume as usual that the original signal $x(t)$ has a $[-T, T]$ finite support. To recap what happens during a single decomposition step, the process previously explained allows us to find $t_0 \in [-T, T]$ that optimally decouples the energy of the even and odd part, which have total support $2T + 2|t_0|$. Once the optimal symmetry point t_0 has been found according to the procedure detailed in Sec. III-A, three total information bearing signals which all together cover the same original support may be defined. The causal part of the even signal is composed by a $T - |t_0|$ interval extent which is the result of the actual "even" computation (half the sum of the original signal and the mirrored version with respect to t_0) followed by a $2|t_0|$ interval extent signal which is simply the tail of the original signal divided by 2. This also applies to the anticausal part of the odd signal (except that in this case the tail precedes it), thus the tail can be considered once. Therefore, a single energy decoupling stage leads to a ternary decomposition, if the tail part is treated as a separate signal. The rationale behind this consists in the fact that the tail part has undergone no modifications and thus can be meaningfully handled separately. In addition, it has at most half the energy of the least significant part of the decomposition whether it is the even or the odd part. The total informative support remains of $2T$ extent, the same of the original signal, divided into three intervals of extents $T - |t_0|$ for the generated even and odd components, and $2|t_0|$ for the tail signal (see Fig. 2b).

This process defines the first level of the transform: the root of the tree $x(t)$ is decomposed into three first-level children nodes, determined by the even causal part $\overline{x}_e^{(c)}(t)$, the odd anticausal part $\overline{x}_o^{(ac)}(t)$, and the tail part $x_t(t)$. The first-level nodes can then be further decomposed into their even, odd, and tail parts. Thus, iterating the decomposition, keeping the causal part for even components, the anticausal part for odd components, and the tail, nine new children nodes are obtained in this second level. In principle, the decomposition can go on *ad infinitum*. The number of nodes on a given level l increases exponentially with the level number, and the support of the single nodes also tends to shrink accordingly. The new representation in each level l of the tree, formed by 3^l components, as explained earlier, does not increase the support, therefore the total support extent remains $2T$.

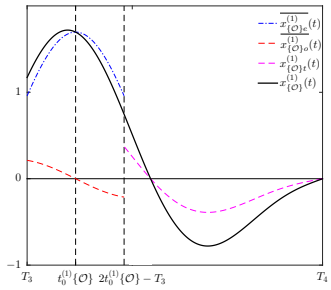
We will now introduce the notation which will be considered throughout the rest of the paper. At the root of the tree, $x(t)$ is decomposed into $\overline{x}_e^{(c)}(t)$, $\overline{x}_o^{(ac)}(t)$ and $x_t(t)$. We



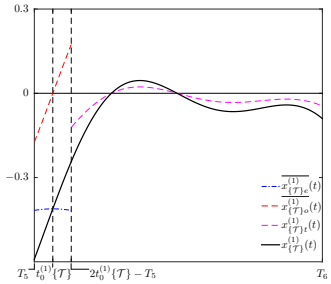
(a) A continuous-time signal $x(t)$ with support $[-T, T]$ is optimally decomposed around $t_0^{(0)}$.



(b) The first-level node $x^{(1)}_{\{\mathcal{E}\}}(t)$ will decompose further around its own optimal symmetry point, $t_0^{(1)}\{\mathcal{E}\}$.



(c) The first-level node $x^{(1)}_{\{\mathcal{O}\}}(t)$ will decompose further around its own optimal symmetry point, $t_0^{(1)}\{\mathcal{O}\}$.



(d) The first-level node $x^{(1)}_{\{\mathcal{T}\}}(t)$ will decompose further around its own optimal symmetry point, $t_0^{(1)}\{\mathcal{T}\}$.

Figure 4: First-level ternary decomposition of a sample signal $x(t)$. Temporal extents T_i are reported without further explanation of their inter-relationships for simplicity. The even node $x^{(1)}_{\{\mathcal{E}\}}(t)$ of Fig. 4b is the causal part of the blue signal of Fig. 4a in the $[t_0^{(0)}, T + 2t_0^{(0)}]$ interval; the odd node $x^{(1)}_{\{\mathcal{O}\}}(t)$ of Fig. 4c is the anticausal part of the red signal of Fig. 4a in the $[-T, t_0^{(0)}]$ interval; the tail node $x^{(1)}_{\{\mathcal{T}\}}(t)$ of Fig. 4d is the magenta signal of Fig. 4a in the $[T + 2t_0^{(0)}, T]$ interval.

refer to these latter signals as $x^{(1)}_{\{\mathcal{E}\}}(t)$, $x^{(1)}_{\{\mathcal{O}\}}(t)$ and $x^{(1)}_{\{\mathcal{T}\}}(t)$, respectively. They constitute the first-level nodes of the ternary decomposition tree. In general, the signal component found at a node in the l level of the decomposition tree can be identified as $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}(t)$, where each $\mathcal{S}_i$ represents a label taking either the value $\mathcal{O}$, $\mathcal{E}$, or $\mathcal{T}$ to relate to an odd, even, or tail portion of the decomposition. By convention, when a node identifies the even component its causal part is preserved, whereas the anticausal part is retained for odd components. The sequence $\{\mathcal{S}_1, \ldots, \mathcal{S}_l\}$ uniquely identifies the position of the node in the ternary tree. We keep the redundant (l) apex to immediately determine the tree depth for that node and avoid counting the number of $\{\mathcal{S}_i\}$ labels. Since the symbols $\{\mathcal{S}_i\}$ are written in increasing level-order, when expressing in natural language what a particular node corresponds to, these symbols should be read in reverse order. So for example $x^{(3)}_{\{\mathcal{O},\mathcal{O},\mathcal{E}\}}(t)$ is the signal found at the third-level node corresponding to the even part of the odd part of the odd part of the original signal, with respect to each preceding level optimal symmetry location.

As we did with the tree nodes, we introduce a suitable notation for the optimal symmetry points for each node. Let $t_0^{(0)}$ be the optimal symmetry point of the root signal $x(t)$. For each successive level l, we define $t_0^{(l)}\{\mathcal{S}_1, \ldots, \mathcal{S}_l\}$ as the optimal symmetry point of the corresponding decomposition tree node, adopting the same convention as the one used to identify the node signals. As such, $t_0^{(2)}\{\mathcal{E}, \mathcal{T}\}$ is the optimal symmetry point of the second-level node associated with the tail signal of the causal even component of the original signal. To compute it, after a first decomposition step with $t_0^{(0)}$, the causal even node $x^{(1)}_{\{\mathcal{E}\}}(t)$ is extracted. The next optimal symmetry point $t_0^{(1)}\{\mathcal{E}\}$ is computed on it and the decomposition is performed again. Finally, $t_0^{(2)}\{\mathcal{E}, \mathcal{T}\}$ corresponds to the optimal symmetry location of the tail signal, $x^{(2)}_{\{\mathcal{E},\mathcal{T}\}}(t)$. Fig. 4 provides an example of a 2-level optimal decomposition for a sample signal (level 1 and level 2), defining nine signal components, three associated to each of the three level-1 decomposition components.

It must be noted that the decomposition tree is actually only approximately ternary. It is indeed possible that one or two children nodes are zero-energy signals. This can happen when: a) the tail may not be present, in the case that t_0 for the parent node falls exactly at the middle of its support, or b) the parent node is perfectly symmetric or antisymmetric (leading to a null odd or null even child node). When this occurs, there is no need to extend the decomposition. In this particular context, for tree coding purposes, an *ad hoc* symbol may be used instead of t_0 to indicate that the decomposition is terminated for that node. This fact has no implication at this stage, but it will be reconsidered later on in Sec. VI.

To stop the decomposition process different criteria may be adopted. For example, the decomposition tree could be interrupted at a certain level S. In this case, the Mirror Transform contains at most 3^S components (since it may contain zero-energy nodes), namely $x^{(S)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_S\}}(t)$, for all labels $\mathcal{S}_i,\ i = 1, \ldots, S$ combinations.

An alternative criterion to stop the decomposition of a given node if its energy falls below a certain fraction of the original signal energy. Another alternative is disregarding nodes corresponding to low enough energy components having too limited temporal support. Generally speaking when the decomposition process is stopped, the associated node always determines a *leaf* node. The Mirror Transform is represented by the concatenation of the signals at the leaves, at their designated level, scanned depth-first according to a predefined label order, for example, even, odd, and tail. The overall leaf signal concatenation, which exhibits the same total support as $x(t)$, is referred to as $\mathbb{F}$. For example, stopping the decomposition tree at the second level yields $\mathbb{F} = \{x^{(2)}_{\{\mathcal{E},\mathcal{E}\}}(t), x^{(2)}_{\{\mathcal{E},\mathcal{O}\}}(t), x^{(2)}_{\{\mathcal{E},\mathcal{T}\}}(t), x^{(2)}_{\{\mathcal{O},\mathcal{E}\}}(t), x^{(2)}_{\{\mathcal{O},\mathcal{O}\}}(t), x^{(2)}_{\{\mathcal{O},\mathcal{T}\}}(t), x^{(2)}_{\{\mathcal{T},\mathcal{E}\}}(t), x^{(2)}_{\{\mathcal{T},\mathcal{O}\}}(t), x^{(2)}_{\{\mathcal{T},\mathcal{T}\}}(t)\}$.

The signals in $\mathbb{F}$ are not enough to represent $x(t)$ because the mirroring points are also needed to invert the transformation. In fact, given the leaf signals in $\mathbb{F}$, the reconstruction process iterates the inversion of a single decomposition step. In detail, the reconstruction of a parent node at a given level consists in first mirroring their even and odd (causal/anticausal) components around their t_0 (retaining and changing the sign respectively), summing them up, and then concatenating the result to the tail component. Depending on

the value of t_0, the tail signal should be appended before or after, but this is just an implementation detail. As an example, a level 3 decomposition would reconstruct the level 2 signal $x^{(2)}_{\{\mathcal{O},\mathcal{E}\}}(t)$ from $x^{(3)}_{\{\mathcal{O},\mathcal{E},\mathcal{E}\}}(t)$, $x^{(3)}_{\{\mathcal{O},\mathcal{E},\mathcal{O}\}}(t)$, and $x^{(3)}_{\{\mathcal{O},\mathcal{E},\mathcal{T}\}}(t)$. Once all level 2 parent nodes are reconstructed, the process is repeated until the root is reached. If $t_0^{(l)}$ represent the optimal temporal locations extracted during the maximal energy decoupling process, we call them the tree information $\mathbb{G}$. In the previous example of Fig. 4, for a tree stopped at level 2, $\mathbb{G} = \{t_0^{(0)}, t_0^{(1)}\{\mathcal{E}\}, t_0^{(1)}\{\mathcal{O}\}, t_0^{(1)}\{\mathcal{T}\}\}$. Thus, the MT of $x(t)$ is fully described by the $\{\mathbb{F}, \mathbb{G}\}$ pair.

IV. Mirror Transform properties

Here we further expound the significance of the Mirror Transform by enumerating some of its properties. They are recapped in Table I. Two additional empirical properties, sparsity and one-wayness, are presented later in Sec. VI.

A. Basic signal transformations

First, we analyze what happens to the MT when some basic signal manipulations are applied to the root signal $x(t)$.

Since the optimal even/odd decomposition that is at the core of the transform tracks the mirroring point where the even and odd component energies are maximally decoupled, it clearly derives that the MT of the shifted signal $x(t - t_1),\ \forall t_1$, is essentially the same: $\mathbb{F}$ is unchanged, and the shifting term t_1 is applied to each mirroring instant of $\mathbb{G}$, obtaining $\mathbb{G}_{t_1}$.

Next, multiplying $x(t)$ with a real constant K has no effect on $\mathbb{G}$, since the energy distribution of $Kx(t)$ among its even and odd parts remains unchanged. Of course, the total energy is now K^2E, so the same scaling factor K affects each node of the decomposition tree, so the leaves in $\mathbb{F}$ become $K\mathbb{F}$.

Finally, the effect of time-scaling is still quite simple. The signal $x(at), \forall a \neq 0$, represents a time axis expansion or contraction, plus time-reversal if $a < 0$, of $x(t)$. The shape of the signal, however, still has the same parity properties, and therefore the auto-convolution only experiences the same time-scaling by a. Thus, the optimal mirroring point at the root of the tree is $t_0^{(0)}/a$. Then, this fact cascades along the decomposition tree, obtaining $\mathbb{G}/a$: in essence, the distance between the mirroring point of the children with respect to that of the parent is scaled by a compared to the original tree. For example, if for a given $x(t)$ we have $t_0^{(0)} = 1$ and $t_0^{(1)}\{\mathcal{E}\} = 2$, for $x(2t)$ we would have $t_0^{(0)} = 1/2$ and $t_0^{(1)}\{\mathcal{E}\} = 1$. The leaves of $\mathbb{F}$ are time-scaled by a, say $\mathbb{F}_a$. Of course, the energy is also scaled by a, as reflected by the shrinking support.

B. Nodes orthogonality and energy preservation

In Sec. II-B, we cited the fact that the generalized even-odd decomposition produces the even and odd signals $x_e(t)$ and $x_o(t)$, that are orthogonal when using the scalar product defined in $\mathcal{L}^2(\mathbb{R})$. By multiplying the two signals by the indicator function to only take the causal and anticausal part, even without detaching the tail signal, orthogonal components are still obtained, because they have disjoint supports. In the end, this decomposition step determines an orthogonal projection on a subspace of the original signal. In such a subspace, the even and odd parts are orthogonal to each other, and they can be considered as new, "independent" signals. The evenness and oddness nature of the considered signals enable at reconstruction to revert the projection from their halved causal or anticausal parts.

Moving on to the Mirror Transform of Sec. III-B, the proposed framework also constructs a third signal, the tail signal $x_t(t)$. Considering its disjoint support with respect to both the truncated components of $x_e(t)$ and $x_o(t)$, it turns out that it is orthogonal to both of them. Therefore, our globally optimal decomposition separates the parent node into three children nodes that are orthogonal to each other. In Fig. 5, there is an approximate depiction of the effects of the decomposition process adopting a vector space representation (the approximate nature of the representation comes from the inherent limitation conveyed by the used 3D plot). In the end, the decomposition constructs three already orthogonal vectors, and it is important to highlight that the resulting children nodes $x^{(1)}_{\{\mathcal{E}\}}(t)$, $x^{(1)}_{\{\mathcal{O}\}}(t)$, and $x^{(1)}_{\{\mathcal{T}\}}(t)$ remain orthogonal, after taking the causal/anticausal parts through the application of the indicator function, and separating the tail.

Given the orthogonality property of the even/odd signals, their energies can be summed as in Eq. (6). However, as a consequence of keeping just the causal and anticausal parts to form $x^{(1)}_{\{\mathcal{E}\}}(t)$ and $x^{(1)}_{\{\mathcal{O}\}}(t)$, the sum of the energy of the first-level children nodes is not E. To preserve the energy of the original signal into its resulting children nodes, it is sufficient to introduce a scale factor $\sqrt{2}$, to compensate for the discarded mirrored parts of the even and odd children. Of course, this can be done for any tree level, which leads to having the same energy E when summing the energy of all nodes in each tree level. It should be observed that this expedient does not alter in any way the tree information $\mathbb{G}$, since as we stated the tree information is invariant to amplitude scaling. However, this caveat is important for the sparsity experimental evaluation which will be reported later on in Sec. VI.

C. Infinitely precise reconstruction of any $\mathcal{L}^2(\mathbb{R})$ function from its coded MT-decomposition

An interesting consequence of the preservation of the $\mathcal{L}^2$-norm in the construction of a fully invertible representation is the ability of the MT to approximate any finite-energy signal $x(t)$ using a peculiar algorithm. Let us suppose we know its partial MT decomposition (*i.e.*, not computed *ad infinitum*), so $\{\mathbb{F}, \mathbb{G}\}$ defines the optimal decomposition tree up to a certain precision level. Consider an arbitrary truncation of such a tree (*e.g.*, by keeping only nodes with energy larger than a threshold), and the nodes associated with the truncated tree. We may keep the tree deep enough, to prevent introducing errors by removing too many low energy nodes from the start.

We may now replace the node waveforms by approximating pairs of even or odd functions (as a simple example, a rectangular impulse and the difference between 2 rectangular impulses, respectively), so as to be able to regenerate an approximation of $x(t)$, which crucially lies at a known distance from it. This algorithm can be optimally driven by incrementally adding nodes in a ranked fashion so that the rounded

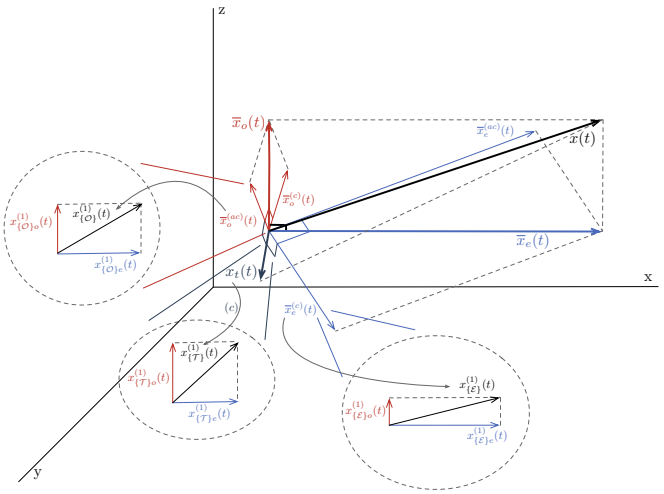
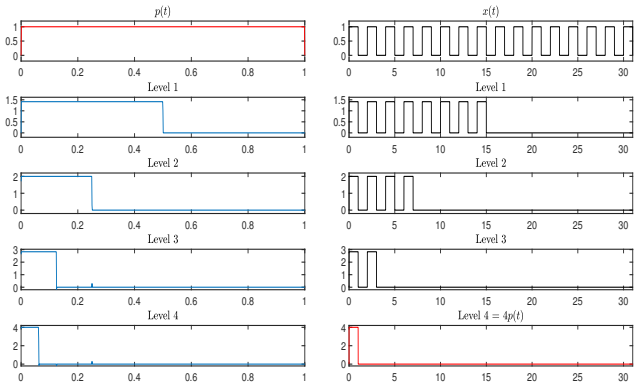


Figure 5: Even-odd ternary iterative decomposition presented on a 3D plot. The original signal $x(t)$ (black) is decomposed into three orthogonal vectors, the truncated even signal $\overline{x}_e(t)$ (blue), the truncated odd signal $\overline{x}_o(t)$ (red), and the tail signal $x_t(t)$ (gray). We have decided to align the x-axis and the z-axis along the direction of $\overline{x}_e(t)$ and $\overline{x}_o(t)$, respectively. They are in turn split into their causal and anticausal parts (we omitted this operation for the tail signal), which are again orthogonal to each other. They represent the first-level nodes of the decomposition tree, respectively $x^{(1)}_{\{\mathcal{E}\}}(t)$, $x^{(1)}_{\{\mathcal{O}\}}(t)$, and $x^{(1)}_{\{\mathcal{T}\}}(t)$. In the ovals, we only sketched the next iteration of the decomposition applied to the children nodes, without tail separation. Recall that for each decomposition a vector is split by selecting the optimal t_0 into the 2 orthogonal components that have the associated maximal and minimal norms.



(a) Constant decomposition.

(b) Decomposition of 2^4 periods of $p(t)$. Level 4 converges to $4p(t)$.

Figure 6: Exemplification of the single branch tree property. Just the first 4 levels are shown. Each level is represented by concatenating the node signals at that level. The fourth level in Fig. 6b converges to $p(t)$ in Fig. 6a except for a constant.

node waveforms introduce the least approximation error with respect to their individual representation. The distance at which the reconstructed signal lies simply corresponds to the sum of the energy loss of the errors introduced in the approximation of the nodes[1].

Overall, since the MT leads to a very compact representation, a simple generative algorithm is constructed, obtained by the inverse MT reconstruction of a limited set of approximated nodes of $\mathbb{F}$ and the tree information $\mathbb{G}$.

D. Single branch trees

When a node is a perfectly even(/odd) signal, the optimal symmetry point is the midpoint, so there is no tail child node and the odd(/even) child node is just a zero-energy signal. Thus, the subtree associated to the latter stops, and so there is a single child spawning from the symmetric parent node. The resulting child may or, more likely, may not be symmetric: in the latter case, it spawns a standard ternary subtree.

Consider now the decomposition of the constant signal $\mathbb{1}(0,T)$, which is illustrated in Fig. 6a until the fourth level. Since it is an even signal, the first level of the decomposition consists in the single even node, which of course is again a constant, with half the support of the original signal. In Fig. 6a, each level is represented by concatenating the node signals at that level, indeed the first level is formed by the constant (even node) and the zero signal (odd node). This decomposition goes on *ad infinitum*, forming a single branch tree with each level having a single node with a constant signal and support shrunk by half.

Employing the energy normalization discussed in Sec. IV-B, to keep the energy contained in each level equal to that of the root node (in this case T), the magnitude of the constant signal in the l-th level node is $2^{\frac{l}{2}}$. Note that the support of the node as the decomposition level goes to infinity tends to 0, but the limit signal is in fact not a Dirac delta (that would have occurred if we imposed constant area instead of constant energy going from one level to the next).

This example can be generalized as follows. Divide the initial support into 2^k intervals of equal length, say I_j, $j = 1,\dots,2^k$. Then build the signal $x(t) = \sum_j H_{ij}(2^k) \cdot \mathbb{1}(I_j)$, where $H(2^k)$ is the Hadamard matrix and $H_i(2^k)$ is its i-th row. In essence, instead of just taking the constant signal (the first row of the matrix), the signal $x(t)$ is an alternating square wave of either even or odd symmetry, such that bisecting it (up to k times) the resulting signal is still either even or odd. In this case the resulting decomposition tree is still a single branch tree, but instead of the even node always surviving, a combination of even and odd nodes do for the first k levels, depending on the selected row of the Hadamard matrix. Then, the surviving node becomes an even constant signal and the previous case applies (see an example in Fig. 6b, where the last row of the Hadamard matrix is considered).

There are other instances where single branch subtrees emerge. Take a signal $p(t)$ with support $[0,T]$, which is either perfectly even or odd ($t_0 = T/2$). Now consider $x(t) = \sum_{n=0}^{2^N-1} p(t-nT)$, that is, 2^N repetitions of $p(t)$, with support $[0,2^N T]$. This is of course a perfectly even signal with midpoint in $t_0 = 2^{N-1}T$, so an even single

[1]Alternative strategies may be considered so as to limit the size of the tree decomposition, for example, at any step, decomposing the node that introduces the largest approximation error in its waveform with respect to all other nodes.

child is built, consisting in 2^{N-1} repetitions of $p(t)$, which is again even and is decomposed into a single even child with 2^{N-2} repetitions of $p(t)$. Such single branch tree goes on for the first $\log_2(N)$ levels of the decomposition tree. In $l = \log_2(N)$, the even child is in fact $p(t)$, and from this point on the decomposition follows that of $p(t)$: a last single branch decomposition, usually followed by a ternary subtree. Therefore, by repeating $p(t)$ 2^N times, we have added a single branch tree $\log_2(N)$ levels deep on top of the original MT of $p(t)$ (see again Fig. 6b). Note that single branches do not occur when an odd number of repetitions of $p(t)$ is instead considered, unless $p(t)$ is already symmetric.

E. Uniqueness

A signal $x(t)$ is uniquely represented by the Mirror Transform $\{\mathbb{F}, \mathbb{G}\}$, the leaves and the tree information, assuming that the decomposition has stopped according to some criterion. The transform $\{\mathbb{F}, \mathbb{G}\}$ is generated by $x(t)$ through a deterministic process, and *vice-versa*, $x(t)$ is uniquely reconstructed starting from the leaves and the tree information. We ascertain in what follows if any of the two components is enough to infer $x(t)$. The answer is no.

First, the leaves $\mathbb{F}$ alone are not sufficient to represent $x(t)$: an immediate proof is the decomposition tree nodes which are unchanged when the root is shifted, as stated in Sec. IV-A.

Instead, it is more subtle that the tree $\mathbb{G}$ does not uniquely identify $x(t)$. Let us consider a node signal anywhere in the tree, say, $z(t)$, with energy E_Z, and its associated optimal mirroring point t_0, found as usual by applying the procedure described in Sec. III-B. The decomposition produces two children nodes $z_e(t)$ and $z_o(t)$, with respective energy E_E and E_O, satisfying $E_Z = E_E + E_O$: we do not bother with separating the tail because we want t_0 to remain fixed, therefore the tail plays no part in this discussion. It can be assumed without loss of generality that t_0 corresponds to the maximum of the even part energy. Recall that t_0 is found by computing $E_e(t_0)$, the even part energy varying the mirroring point, and then taking the maximum, so the maximum of $E_e(t_f)$ is E_E at the t_0 time instant. If $z_e(t)$ is multiplied by a factor $K > 1$, the position of the maximum of $E_e(t_f)$ is not changed, since the entire function is simply multiplied by K^2, and E_E is scaled by the same factor. It is easy to show that, to keep the energy of the parent $z(t)$ equal to E_Z, the odd part $z_o(t)$ must be multiplied by C with $C^2 E_O = E_Z - K^2 E_E$. As $C < 1$, the maximum of $E_o(t_f)$ is still smaller than the maximum of $E_e(t_f)$, which is still located at t_0. Therefore, the signal reconstructed by $Kz_e(t)$ and $Cz_o(t)$ has the same energy as $z(t)$ and admits the same optimal mirroring point t_0. Furthermore, the entire successions of the optimal mirroring points, generated starting from the nodes $Kz_e(t)$ and $Cz_o(t)$, are not altered with respect to the ones associated with $z_e(t)$ and $z_o(t)$, as remarked in Sec. IV-A. This shows that, by applying this procedure to any node, it is possible to construct infinite signals that all have the same tree $\mathbb{G}$ information. Of course, the leaves are different because they have been scaled in the process, so $\mathbb{F}$ is needed to uniquely reconstruct the root signal.

Table I: MT Properties.

Homogeneity	$Kx(t) \xrightarrow{MT} \{K\mathbb{F}, \mathbb{G}\}$
Shift invariance	$x(t-t_1) \xrightarrow{MT} \{\mathbb{F}, \mathbb{G}_{t_1}\}$
Time scaling	$x(at) \xrightarrow{MT} \{\mathbb{F}_a, \mathbb{G}/a\}$
Orthogonality	$x^{(l)}_{\{\ldots,\mathcal{E}\}}(t) \perp x^{(l)}_{\{\ldots,\mathcal{O}\}}(t) \perp x^{(l)}_{\{\ldots,\mathcal{T}\}}(t)$
Constant energy	Energy of $x(t)$ is $E \Longrightarrow$ For each level l, the sum of energies of $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}(t)$ is E
Controlled distortion	Distortion δ in $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}(t) \Longrightarrow$ distortion δ in reconstructed $x(t)$
Single branch trees	Infinite: (pseudo-)constant nodes Finite: 2^n repetitions of a symmetric wave
Uniqueness	$x(t) \xleftrightarrow{MT} \{\mathbb{F}, \mathbb{G}\}$

V. The Discrete Mirror Transform

The Mirror Transform can be cast in the discrete domain as well. The formal definition of what we refer to as the Discrete Mirror Transform (DMT) is given in Sec. V-A. Furthermore, the Mirror Transform, both for continuous- and discrete-time signals, can be easily extended in multidimensional spaces. For the sake of brevity, only the case for 2D finite energy sequences is discussed in Sec. V-B, as it leads well to the possibility to perform experimental evaluation on real data.

A. DMT definition in 1D domain

Let us consider a discrete-time, finite-energy sequence $x[n] \in \ell^2(\mathbb{Z})$. Assuming that the support of $x[n]$ is $1, \ldots, L$ with L finite, Eq. (5) becomes:

$$x_e[n; n_f] = \frac{x[n] + x[2n_f - n]}{2}; \; x_o[n; n_f] = \frac{x(n) - x[2n_f - n]}{2}, \tag{13}$$

where $x[n] = x_e[n; n_f] + x_o[n; n_f]$. Searching for the optimal symmetry point $n_f = n_0$ still consists in choosing the one that lets either E_e or E_o be the global maximum. As a note, always choosing n_f as the sequence midpoint (namely, applying the standard even/odd decomposition) in place of the optimal symmetry point provides approximately balanced decomposition trees strictly related to the Walsh-Hadamard Transform (WHT) of the sequence [16], the only difference being the order with which the transformed coefficients may be output by the standard WHT (see [17] for more details).

In the discrete case, it is better to directly evaluate the energy of the even sequence rather than its derivative, since the discrete domain is not suited for differential operands. Then:

$$\begin{aligned} E_e(n_f) &= \sum_{n=-\infty}^{+\infty} |x_e[n; n_f]|^2 = \sum_{n=-\infty}^{+\infty} \left| \frac{x[n] + x[2n_f - n]}{2} \right|^2 = \\ &= \frac{1}{4} \sum_{n=-\infty}^{+\infty} |x[n]|^2 + |x[2n_f - n]|^2 + 2\,\mathrm{Re}\{x[n] x^*[2n_f - n]\} = \\ &= \frac{1}{2} E + \frac{1}{2} \sum_{n=1}^{L} \mathrm{Re}\{x[n] x^*[2n_f - n]\}. \end{aligned} \tag{14}$$

Using the linear convolution for energy sequences yields:

$$E_e(n_f) = \frac{1}{2}E + \mathrm{Re}\{(x * x^*)\}[2n_f], \tag{15}$$

so in the end:

$$2n_0 = \arg\max_m \; |\mathrm{Re}\{(x * x^*)\}[m]|. \tag{16}$$

The optimal symmetry point n_0, found through Eq. (16), cannot be arbitrary, but it must either be an integer or a half-integer position. For $x[n], n = 1, \ldots, L$, n_0 can then take values in the $2L$–1 cardinality set $\{1, \frac{3}{2}, 2, \ldots, L\text{–}1, L\text{–}\frac{1}{2}, L\}$. In the integer case, by definition $x_e[n_0; n_0] = x[n_0]$ whereas $x_o[n_0; n_0] = 0$, which is non informative.

Note that again the decomposition does not increase the support needed for the original signal reconstruction, as it was for the continuous-time case. In addition, Eq. (3) still holds, provided the integration operator is substituted by the summation one:

$$\begin{aligned} E &= \sum_{n=-1}^{L} |x[n]|^2 = \sum_n |x_e[n] + x_o[n]|^2 = \\ &= \sum_n |x_e[n]|^2 + \sum_n |x_o[n]|^2 = E_e + E_o, \end{aligned} \tag{17}$$

where the summation limits for the even and odd sequences may extend beyond $[1, L]$, depending on n_0. Again, we retain the causal part of the even subsequence $x_e^{(c)}[n]$, and the anti-causal part of the odd subsequence $x_o^{(ac)}[n]$, whose support is one sample shorter with respect to $x_e^{(c)}[n]$ for an integer valued n_0. These two subsequences define non-redundant informative parts. The tail subsequence $x_t[n]$ can then be separated as before, and it corresponds to the part of $x[n]$ not involved in any computation. Additional details on this single step of the decomposition process in the discrete case, complete with graphical depictions, can be found in the accompanying documentation of the publicly available code [13].

The decomposition process then repeats the single decomposition step in an iterative fashion. Given the discrete, finite support nature of the signal domain, the iteration is bound to stop when single-sample sequences are encountered, which are thus necessarily decomposition tree leaves as they cannot be further decomposed: a situation not found for continuous-time signals. Therefore, decomposition trees for discrete-time, finite support sequences are always finite. Of course, zero-energy sequences in any node can still occur as it was for continuous-time signals, so the corresponding subtree could also be trimmed.

The notation introduced in Sec. III-B for continuous-time decomposition trees can be used for discrete-time sequences, allowing all the nodes to be specified by $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}[n]$, as l varies, with all the symbols sharing the meaning as for the continuous-time case, in particular $\mathcal{S}_i$ which is still either $\mathcal{E}$, $\mathcal{O}$, or $\mathcal{T}$. Likewise, the optimal symmetry point of a given node can be defined by $n_0^{(l)}\{\mathcal{S}_1, \ldots, \mathcal{S}_l\}$, as l varies. In the end, the Discrete Mirror Transform (DMT) of $x[n]$ is fully described by the $\{\mathbb{F}, \mathbb{G}\}$ pair, where $\mathbb{F}$ identifies the set of the tree leaves and $\mathbb{G}$ the set of optimal symmetry locations.

Algorithm 1 1D DMT algorithm.

Require: Original sequence $x[n]$, $n = 1, \ldots, L$
Ensure: $\{\mathbb{F}, \mathbb{G}\}$

1: Level $l = 0$, find $n_0^{(0)} \Leftarrow$ Eq. (16)
2: Encode $n_0^{(0)}$ in $\mathbb{G}$
3: Build children nodes:

$$\begin{cases} x^{(1)}_{\{\mathcal{E}\}}[n] \\ x^{(1)}_{\{\mathcal{O}\}}[n] \ ^a \\ x^{(1)}_{\{\mathcal{T}\}}[n] \ ^b \end{cases} \Leftarrow \text{Single step}$$

4: **repeat**
5: $l = l + 1$
6: **for** non-empty nodes at level l, $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}[n]$ **do**
7: $L \Leftarrow \text{length}\left(x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}[n]\right)$
8: **if** $L > 1$ **then**
9: **if** Pruning condition ^{c} **then**
10: Encode symbol 0 in $\mathbb{G}$
11: **continue**
12: **end if**
13: Find $n_0^{(l)}\{\mathcal{S}_1, \ldots, \mathcal{S}_l\} \Leftarrow$ Eq. (16)
14: Encode $n_0^{(l)}\{\mathcal{S}_1, \ldots, \mathcal{S}_l\}$ in $\mathbb{G}$
15: Build children nodes:

$$\begin{cases} x^{(l+1)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l,\mathcal{E}\}}[n] \\ x^{(l+1)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l,\mathcal{O}\}}[n] \ ^a \\ x^{(l+1)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l,\mathcal{T}\}}[n] \ ^b \end{cases} \Leftarrow \text{Single step}$$

16: **else if** $L = 1$ (this node is a leaf) **then**
17: Encode single-valued $x^{(l)}_{\{\mathcal{S}_1,\ldots,\mathcal{S}_l\}}[n]$ in $\mathbb{F}$
18: **end if**
19: **end for**
20: **until** there are no nodes at level l with $L > 1$

^{a} The odd child node may not exist if n_0 is integer and $L = 2$. In that case, only the even and tail leaves are created.
^{b} The tail child node may not exist if n_0 is the midpoint.
^{c} This condition applies if tree pruning is employed. See Sec. VI-A and Fig. 7c. Zero-energy nodes are always pruned.

Algorithm: The complete algorithm expressed as pseudo-code is given in Alg. 1. Therein, the single step decomposition that we have just described is referred to as "Single step". This operation outputs the three even, odd, and tail children nodes (or sometimes only two of them, see footnotes a and b).

After finding the n_0 for the input sequence (line 1), encoding it in $\mathbb{G}$ (line 2), and completing the first decomposition step (line 3), the decomposition process continues in ever deeper levels until all the nodes in a given level are leaves (nodes with length 1) which no longer require to be decomposed (line 20). In fact, when the node is a leaf it is encoded in $\mathbb{F}$ and no further decomposition takes place (lines 16-17). In a given level l, not every combination $\{\mathcal{S}_1, \ldots, \mathcal{S}_l\}$ corresponds to a tree node, since the tree could have stopped at a previous level because a leaf was obtained (or a pruning condition applied, see footnote c). For example, in Fig. 7, for $l = 3$ only 4 label combinations are non-empty nodes (and in this case, all

of them correspond to leaves, so the decomposition does not proceed to $l = 4$).

A non-leaf node, instead, undergoes the single step decomposition process as before (lines 11-13), building children nodes in the next level. As mentioned, when tree pruning is employed (see Sec. VI-A), prior to the decomposition process each non-leaf node is considered for pruning (line 9). If the pruning condition is met (*i.e.*, zero or negligible energy node), the appropriate symbol 0 is encoded in the tree information $\mathbb{G}$ (line 10) and the decomposition for this node stops.

Remarks: Using the ternary decomposition process on discrete sequences may lead to the construction of a very unbalanced tree. In fact, the decomposition generates three sequences: the even and odd sequences support does not differ by more than one sample, but the tail signal may be much longer or much shorter. The total support remains that of the parent node. Thus, when considering the complete decomposition of a length L sequence, its L single-valued leaves may be all over the tree levels, depending on the particular n_0 associated with each node.

The number of decomposition levels is minimized whenever symmetry points are found near the midpoints since the maximum length of the children even/odd sequences is limited by approximately half the one of the parent sequence and the tail node in such case remains very short. On the other hand, whenever symmetry points occur near the end points of the original sequence, this generates very short even/odd sequences and a very long tail, almost as long as the parent sequence. If this latter case happens frequently in the decomposition process, the number of decomposition levels is bound to increase. In the extreme cases, if the global symmetry is always found in the midpoint of any given node the number of decomposition levels is $\lceil \log_2 L \rceil$, while if it is always put on the first or last sample the number of levels tends to L.

Nevertheless, there are always at most $L-1$ values in $\mathbb{G}$. In fact, a length 2 sequence has a single n_0 that generates two leaves, stopping the decomposition. A length 3 sequence can be decomposed either into three leaves or a leaf and a length 2 sequence, thus it needs at most two n_0. The preceding statement can thus be inferred by induction.

The computational complexity of the decomposition process, as expected, depends on the shape of the original sequence. Assuming a real-valued input sequence of length N power of 2, and assuming to employ radix-2 algorithms to compute the convolution, a single step of the decomposition involves $N/4 \cdot \log_2(N)$ complex multiplications. Following the just expounded considerations on the number of levels, in the best case scenario, where the optimal symmetry point is always in the midpoint, the overall complexity is $O\left(N \cdot (\log_2 N)^2\right)$, whereas in the worst case scenario, where the optimal symmetry point is always on the edge, the overall complexity is $O\left(N^2 \cdot \log_2 N\right)$.

Furthermore, the decomposition of a length N sequence requires $2N$ additions and a flipping operation. Therefore, $O\left(N \cdot \log_2 N\right)$ additions, for the best case scenario, or $O\left(N^2\right)$ additions, for the worst case scenario, are also needed. The flipping operations are at most $N-1$, depending on the tree shape.

A further observation concerns the synthesis, or reconstruction, process of the DMT. We want to highlight that the knowledge of the $\{\mathbb{F}, \mathbb{G}\}$ pair is sufficient to reconstruct the original signal. This may not appear clear at first glance, since the position of the leaves, which is a necessary information to reconstruct the original signal, is not directly present in $\mathbb{G}$. However, note that the position of the leaves can be extracted from $\mathbb{G}$ by reading the optimal positions n_0 in increasing level order as these values allow to determine the sequence length of the children nodes. The tree leaf values, *i.e.*, the transform coefficients in $\mathbb{F}$, that are located all over the tree in different levels, can be identified by those nodes with length 1, ordered in the same fashion. The example that follows in the next paragraph helps to clarify this point.

As a last remark on the reconstruction process, we note that it has a computational complexity much lower than the decomposition stage. To reconstruct a length N node, at most 2 mirroring operations (that is, creating a copy of the even and odd children to build back the even and odd sequences) and N additions, with no multiplications besides some sign reversing, are needed. To reconstruct the perfectly balanced tree, the number of required additions is $N \cdot \log_2 N$, whereas for the perfectly unbalanced tree $O\left(N^2\right)$ additions are needed.

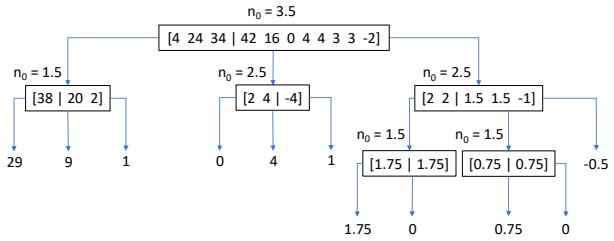
Example: As an instructive example, consider Fig. 7. A length-11 sequence $x[n]$, with support $n = 1, 2, \ldots, 11$ is decomposed in Fig. 7a. The optimal symmetry point of $x[n]$ is $n_0^{(0)} = 3.5$. Therefore, in the first level, the even child $x^{(1)}_{\{\mathcal{E}\}}[n]$ (left branch) is $1/2 \cdot \{42+34, 16+24, 0+4\}$, the odd child $x^{(1)}_{\{\mathcal{O}\}}[n]$ (center branch) is $1/2 \cdot \{4-0, 24-16, 34-42\}$ (recall the we retain the anti-causal part of the odd sequence), and the tail child $x^{(1)}_{\{\mathcal{T}\}}[n]$ is $1/2 \cdot \{4, 4, 3, 3, -2\}$.

Regarding the second level, for $x^{(1)}_{\{\mathcal{E}\}}[n]$, which is a length 3 sequence, $n_0^{(1)}\{\mathcal{E}\} = 1.5$, so it is decomposed into three leaves: $x^{(2)}_{\{\mathcal{E},\mathcal{E}\}}[n]$ is $1/2 \cdot \{20+38\}$, $x^{(2)}_{\{\mathcal{E},\mathcal{O}\}}[n]$ is $1/2 \cdot \{38-20\}$, and $x^{(2)}_{\{\mathcal{E},\mathcal{T}\}}[n]$ is $1/2 \cdot \{2\}$. For $x^{(1)}_{\{\mathcal{O}\}}[n]$, which is also a length 3 sequence, $n_0^{(1)}\{\mathcal{O}\} = 2.5$, so it is decomposed into three leaves as well: $x^{(2)}_{\{\mathcal{O},\mathcal{E}\}}[n]$ is $1/2 \cdot \{-4+4\}$, $x^{(2)}_{\{\mathcal{O},\mathcal{O}\}}[n]$ is $1/2 \cdot \{4-(-4)\}$, and $x^{(2)}_{\{\mathcal{O},\mathcal{T}\}}[n]$ is $1/2 \cdot \{2\}$. Then, $n_0^{(1)}\{\mathcal{T}\} = 2.5$ for the length 5 $x^{(1)}_{\{\mathcal{T}\}}[n]$ generates the length 2 even child $x^{(2)}_{\{\mathcal{T},\mathcal{E}\}}[n] = 1/2 \cdot \{1.5+2, 1.5+2\}$, the length 2 odd child $x^{(2)}_{\{\mathcal{T},\mathcal{O}\}}[n] = 1/2 \cdot \{2-1.5, 2-1.5\}$, and the tail leaf $x^{(2)}_{\{\mathcal{T},\mathcal{T}\}}[n] = 1/2 \cdot \{-1\}$.

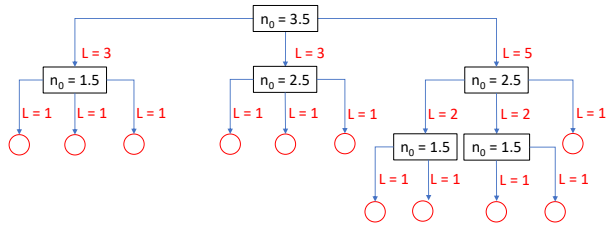
Finally, in the third level, only 4 leaves are present: $x^{(3)}_{\{\mathcal{T},\mathcal{E},\mathcal{E}\}}[n]$, $x^{(3)}_{\{\mathcal{T},\mathcal{E},\mathcal{O}\}}[n]$, $x^{(3)}_{\{\mathcal{T},\mathcal{O},\mathcal{E}\}}[n]$, and $x^{(3)}_{\{\mathcal{T},\mathcal{O},\mathcal{O}\}}[n]$, resulting from the decomposition of $x^{(2)}_{\{\mathcal{T},\mathcal{E}\}}[n]$ and $x^{(2)}_{\{\mathcal{T},\mathcal{O}\}}[n]$, with $n_0^{(2)}\{\mathcal{T},\mathcal{E}\}$ and $n_0^{(2)}\{\mathcal{T},\mathcal{O}\}$ both equal to 1.5 (so the leaves are the semi-sum and semi-difference of the parent node values).

In the end, the DMT of $x[n]$ is given by the $\{\mathbb{F}, \mathbb{G}\}$ pair, where $\mathbb{F}$ is the set of 11 leaves. Their positions in the tree are not explicitly specified in $\mathbb{F}$, but are encoded through $\mathbb{G} = \{3.5, 1.5, 2.5, 2.5, 1.5, 1.5\}$, which is the succession of n_0 obtained above.

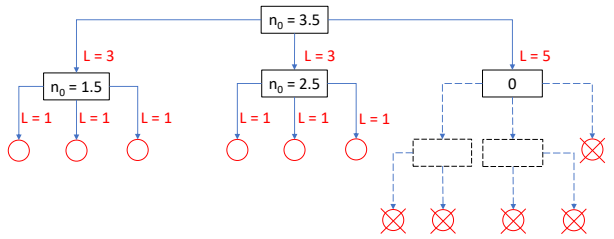
Therefore, there is indeed no need to include the length of



(a) Decomposition tree of a length-11 sequence. The vertical bar represents the position of the corresponding n_0 (in this case, n_0 is by chance always haf-integer).



(b) The same tree, highlighting the relation between nodes length and symmetry points.



(c) Tree pruning example, in this case the first-level tail is removed.

Figure 7: An example of ternary tree for a sample length-11 sequence $x[n]$. In (a), the decomposition tree of an example length 11 sequence is depicted. In (b), the optimal decomposition points of $\mathbb{G}$ inside the black rectangles allow to infer the children node lengths generated from a parent node, and thus the transform coefficient positions (red circles). In (c), the same ternary tree pruned after zeroing the coefficients derived from the node representing $x^{(1)}_{\{\mathcal{T}\}}[n]$. The entire subtree originated from that node can be removed (dashed lines), and the associated $n^{(1)}\{\mathcal{O}\}$ is replaced with the symbol 0. The cardinality of $\mathbb{G}$ is thus reduced by the resulting pruning.

the nodes in the tree information, as shown in Fig. 7b. Since the first value of $\mathbb{G}$ is $n_0^{(0)}=3.5$, as a consequence of how the decomposition works and knowing L, the lengths of $x^{(1)}_{\{\mathcal{E}\}}[n]$, $x^{(1)}_{\{\mathcal{O}\}}[n]$, and $x^{(1)}_{\{\mathcal{T}\}}[n]$ are deduced as 3, 3, and 5, respectively. The next values in $\mathbb{G}$ are associated with the second level, thus $n_0^{(1)}\{\mathcal{E}\} = 1.5$, $n_0^{(1)}\{\mathcal{O}\} = 2.5$, and $n_0^{(1)}\{\mathcal{T}\} = 2.5$. All the children nodes descending from the even and odd nodes are length-1 nodes, as well as the tail of the tail node: these correspond to the first seven transform coefficients in $\mathbb{F}$. Instead, $x^{(2)}_{\{\mathcal{T},\mathcal{E}\}}[n]$ and $x^{(2)}_{\{\mathcal{T},\mathcal{O}\}}[n]$ are length-2 nodes, which generate the final four leaves in the third and last level. The complete tree structure, including the leaf positions, is thus obtained, and the original sequence can then be reconstructed starting from $\{\mathbb{F},\mathbb{G}\}$.

B. Extension to 2D domain

Considering a 2D sequence $x[m,n]$, in this case real-valued for simplicity, and having finite support D with M rows and N columns. Then, a generic point reflection even/odd decomposition may be defined around an arbitrary point $[m_f, n_f]$. Formally, we can write:

$$\begin{aligned} x_e[m,n;m_f,n_f] &= \frac{x[m,n]+x[2m_f-m,2n_f-n]}{2}; \\ x_o[m,n;m_f,n_f] &= \frac{x[m,n]-x[2m_f-m,2n_f-n]}{2}. \end{aligned} \tag{18}$$

In order to find the optimal symmetry point $[m_0,n_0]$, the main steps in Sec. V-A for the 1D domain can be properly adjusted to the 2D domain. In the end, the energy of the even part, as the point $[m_f,n_f]$ varies, can be written as:

$$E_e(m_f,n_f) = \frac{1}{2}E + \frac{1}{2}\underbrace{\sum_{(m,n)\in D} x[m,n]\cdot x[2m_f-m,2n_f-n]}_{(:=x**x)[2m_f,2n_f]}, \tag{19}$$

where the last term is the 2D discrete linear convolution, using the symbol $**$. Then, the optimal $[m_0,n_0]$ point that maximally splits the energy of x_e and x_o is computed as:

$$[2m_0,2n_0] = \arg\max_{(h,k)} \left|(x**x)[h,k]\right|. \tag{20}$$

In the 2D domain we can also examine symmetries with respect to an arbitrarily oriented line, instead of a generic point. For example, by considering $n=n_f$ as the vertical reflection line, the even/odd components of $x[m,n]$ with respect to it are expressed as:

$$\begin{aligned} x_e[m,n;n_f] &= \frac{x[m,n]+x[m,2n_f-n]}{2}; \\ x_o[m,n;n_f] &= \frac{x[m,n]-x[m,2n_f-n]}{2}. \end{aligned} \tag{21}$$

In this case, searching for the optimal n_0 still uses the 1D convolution instead:

$$\begin{aligned} E_e(n_f) &= \frac{1}{2}E + \frac{1}{2}\underbrace{\sum_{(m,n)\in D} x[m,n]\cdot x[m,2n_f-n]}_{(x*x)[2n_f]}; \\ 2n_0 &= \arg\max_{h} \left|(x*x)[h]\right|. \end{aligned} \tag{22}$$

Of course, when considering the reflection with respect to $m=m_f$, that is using a horizontal reflection line, the same result is obtained by just applying the mirroring in the orthogonal direction.

All considerations made previously for the DMT can also be extended to 2D sequences, regardless of the considered reflection. It must be highlighted that by iterating the optimal even/odd decomposition with respect to a horizontal or vertical line, it will converge to a number of 1D sequences. For example, for the decomposition using the vertical reflection plane the support (namely, the number of columns) of the node sequences decreases with the tree depth. In the end the original 2D sequence $x[m,n]$ is decomposed into N sequences of length M. For those sequences, the 1D DMT can be employed in turn to lead to a full 2D transformed representation.

Variations to this process may be considered [18], *e.g.*, by alternating horizontal and vertical mirroring decomposition steps, or by changing arbitrarily the direction of mirroring from one step to the next. With no regard to the adopted scheme and with the right attention, the concepts can be easily extended from the 1D case, so as to preserve the constant nature of the informative support from one decomposition stage to following one. We refer the interested reader to the publicly available code for further details on the 2D implementation of the DMT.

VI. Experimental use of the DMT - Preliminary results

In this section, more specific peculiarities of the Mirror Transform are examined. In contrast with the properties previously described in Sec. IV, they are not a direct consequence of the transform definition. As a matter of fact, they are shown to be valid through a series of experimental tests. They could also hint at potential application for signal processing tasks, though at this stage they should be considered strictly as proof-of-concept trials. The associated code and the examined datasets are publicly available at [13].

We ran two sets of experiments. In the first, discussed in Sec. VI-A, we delve into the sparsity properties of the MT, by studying its ability to concentrate the signal energy in a small set of transform coefficients. Then, in Sec. VI-B we inspect what we call the one-wayness property, namely, the one-way relation between the original signal and its associated decomposition tree structure.

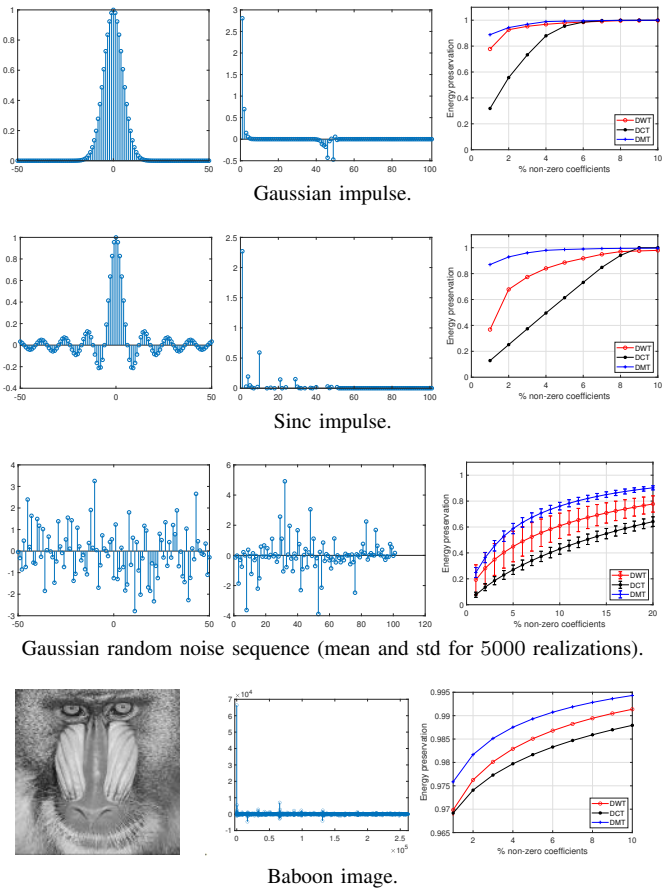
Of course, the scope of the tests is limited for practical reasons to finite-support signals in the discrete domain. In those cases, we recall that the decomposition tree has a finite number of levels and, for a L samples original signal, if the process is carried on until all the single-valued leaves are reached, there are up to L coefficients stored in $\mathbb{F}$. They are distributed all over the tree, depending on the optimal symmetry points in $\mathbb{G}$ found in the decomposition process.

A. Sparsity

Through a single step of the optimal ternary decomposition, a relevant part of the signal energy is bound to move into a support smaller than the starting one, and consequently this intrinsically leads to a new, sparser representation of the signal.

We prove this statement through a series of experimental tests. In particular, we investigated the ability of the DMT to concentrate more signal energy into a small transform coefficient set when opposed to the ability of other transforms, which are known to be efficient to approximate several classes of signals.

The comparison has been performed on various test sequences, by studying the energy preservation associated with a variable number of the transform coefficients sorted from largest to smallest value (in modulus). Specifically, for the DMT the whole decomposition tree is first generated, then the energy of the representation is computed using a certain number of the most significant coefficients in terms of energy. The same process is performed for the competitive transforms,



Gaussian impulse.

Sinc impulse.

Gaussian random noise sequence (mean and std for 5000 realizations).

Baboon image.

Figure 8: A sample of the investigated test sequences. The first two columns display the signal with its associated DMT. The third column reports the DMT, DCT and DWT energy curves.

by retaining the most significant coefficients in the transform domain, according to their magnitude.

First, Fig. 8 illustrates the performance comparison between the proposed transform and two classical transforms, namely, the Discrete Cosine Transform (DCT) and the Discrete Wavelet Transform (DWT) implemented using the 'db4' wavelet with the maximum decomposition level, for some test sequences. In the first rows, three 1D sequences are examined, namely, a Gaussian impulse, a sinc impulse, and one instance of a Gaussian random noise sequence. The fourth row addresses the popular *Baboon* image, and the 2D DMT described in Sec. V-B has been applied. In this case, the decomposition is first performed using the vertical reflection plane, namely, along the image columns, and then the 1D DMT is performed on the resulting sequences. The energy curves show that the DCT and DWT performance is consistently worse than the one given by the DMT for all test sequences. We recall that the DMT significantly goes beyond the concept of representing a vector through a linear expansion. Instead, it provides a non-linear representation, which can describe the nature of the signal more effectively.

The superior sparsity properties of the DMT with respect to DCT and DWT can be explained in terms of coefficients energy decay rate. For the DCT, it is known to be polynomial, while for DWT is exponential only in the case of piece-wise

linear functions. The DMT, instead, always enjoys exponential energy decay rate, provided that at each step of the decomposition most the energy is concentrated in at most half of the previous support. This occurrence is highly likely, namely, it is verified if the energy decoupling between the even and odd parts is effective. Since the decomposition is pursued exactly to maximize the energy concentration in just one of the parts, the considered decomposition is optimal in this respect.

To reinforce the reliability of the results, we have expanded our experiments to more datasets, composed by:

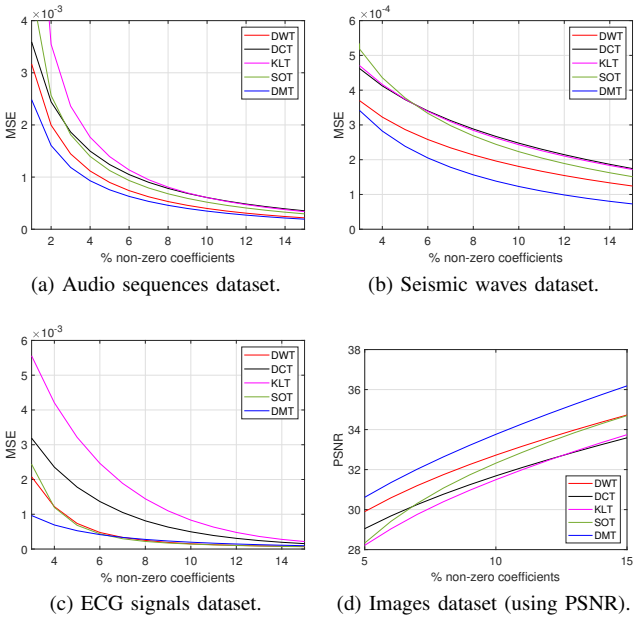
- Short audio sequences, extracted from mixed music genre songs and vocal sound effects [13] (Fig. 9a);
- Seismic waves, part of the IRIS database [19] (Fig. 9b);
- Electrocardiogram (ECG) sequences, taken from the PhysioNet database [20] (Fig. 9c);
- Different types of images, from natural to textures, with various resolutions [13] (Fig. 9d).

Furthermore, to evidence more effectively the sparsity property of the DMT, we have included two data-driven transforms in the comparison: the Karhunen-Love Transform (KLT) and the Sparse Orthonormal Transform (SOT).

We recall that the KLT is the matrix of orthonormal eigenvectors of the covariance matrix of a stochastic process. Assuming that the columns of the KLT are ordered so that the associated eigenvalues are in decreasing order, then it is well-known that the KLT optimizes n-term linear approximation performance, where the optimality is expressed as the expected mean square error when a signal is projected on the first n vectors. However, the sparsity property of a transform is examined by applying n-term non-linear approximation, where the expected mean square error is computed by projecting a signal on the n vectors associated to the most representative transform coefficients, *i.e.*, the ones corresponding to the highest energy. For this reason, we have also considered the SOT in our experimental comparison, since it is a data-driven orthonormal transform designed to achieve the minimum distortion possible when keeping the least number of non-zero transform coefficients. More details on the SOT derivation can be found in [21].

To compute the KLT and SOT bases for each class of signals, each dataset has been first randomly split in a training and in a test set. Then, each data of the training set has been divided in sub-sequences, which have been finally used to construct the KLT and the SOT matrices. Specifically, 512-length sub-sequences have been considered for the audio dataset, while the seismic and ECG data have been both separated into 400-length sub-sequences; finally, the images have been partitioned into 8×8 blocks.

Fig. 9 reports the average performance in terms of MSE and PSNR for each test set. It confirms the superior sparsity ability of the DMT with respect to all the competitive transforms for all the considered datasets. As expected, the SOT outperforms the KLT, since a non-linear approximation of the signals is considered. However, the DMT performs consistently better than the SOT, proving that the iteration of the optimal symmetry decomposition is able to efficiently describe a signal, independently of its different nature.



(a) Audio sequences dataset. (b) Seismic waves dataset.

(c) ECG signals dataset. (d) Images dataset (using PSNR).

Figure 9: Performance comparison between the competing transforms when applied to the considered datasets, given in terms of average performance.

In particular, the comparison results with DCT and DWT are quite remarkable, since these transforms are key elements in practical coding standards, given their ability to efficiently represent various signal classes. However, it is also important to state that the sparsity properties of a transform are not generally sufficient to ensure that it would perform well in a coding algorithm. Indeed, in addition to the non-zero coefficient values, their positions must be specified too in order to reconstruct the original image. Encoding the positions of the non-zero coefficients usually requires a thorough investigation, which is specific for each given transform and application context, *e.g.*, for image compression, zig-zag scanning for the DCT in JPEG and H.26x, or the EZ of wavelet transforms in JPEG2000.

Thus, in a coding framework, in order to take advantage of the previously shown sparsity properties of the coefficients in $\mathbb{F}$, an efficient representation of the tree information is needed. While proposing the MT for coding applications is beyond the scope of this paper, we include here some observations hinting at how an application like this could be designed.

As we previously mentioned, after zeroing some of the leaves in a controlled way, for example, following a quantization of the coefficient values, as it happens for traditional image/video compression standards, more tree information compaction can be achieved. In fact, if the leaves associated to a same subtree have all been zeroed, that subtree can be entirely removed without losing any information, which would allow for the pruning of a potentially large number of nodes, thus reducing the size of $\mathbb{G}$. A possible way to exploit this fact is to signal that the decomposition process must stop for that zeroed subtree. In other words, an *ad-hoc* symbol, for example 0, can be included in $\mathbb{G}$ to indicate a node that would not generate any non-zero coefficients. In Fig. 7c an example

of pruning the tree in Fig. 7 is shown. In this example, 5 coefficients have been zeroed, all of them deriving from the $x^{(1)}_{\{\mathcal{T}\}}$ node. Then, the entire subtree which originated from that node can be removed, and the associated $n^{(1)}_{\{\mathcal{T}\}}$ is replaced with 0. This process leads to a more compact representation of the tree $\mathbb{G}$.

In the next section, the compact description $\mathbb{G}$ of pruned trees is exploited in a different direction, by showing another distinctive experimental property of the Mirror Transform.

B. One-wayness

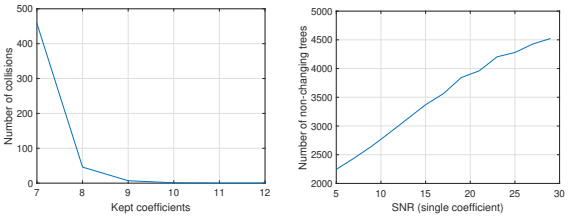
The one-wayness property is akin to what typically characterizes hash functions. It is stated here for the Discrete Mirror Transform of finite support sequences, but it is also easy to show its validity for the standard Mirror Transform.

Assuming that a given sequence $x[n]$ is decomposed through Eq. (13) into $x_e[n; n_0]$ and $x_o[n; n_0]$ around the optimal symmetry point n_0, with respective energy E_e and E_o, it is of course possible to define another sequence $y_e[n; n_0]$, with energy still E_e, and then obtain the reconstructed sequence $y[n] = y_e[n; n_0] + x_o[n; n_0]$. As shown in Sec. IV-E, it is still possible for the alternative parent node to have the same optimal mirroring point (in that case, both the new children nodes are changed and are strictly scaled versions of the original ones). In general, however, the optimal symmetry point for $y[n]$ is not the same as for $x[n]$.

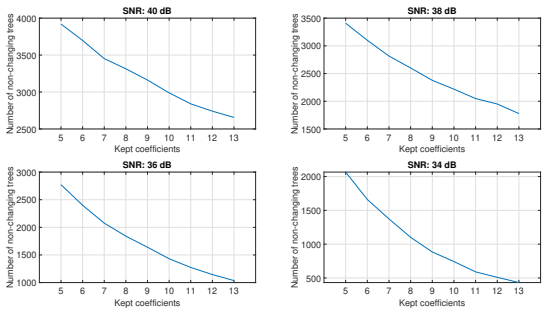
This means that, given the tree $\mathbb{G}$, it is not possible to assign arbitrary sequences to the nodes (even with the correct support, as inferred by the tree information). In fact, if the root is first reconstructed from these arbitrary nodes and then decomposed again, the decomposition tree is likely going to be different (a possible exception being the new nodes which are all scaled versions of the previous ones, as stated earlier). Therefore, while it is always possible, with a given $\mathbb{G}$, to reconstruct a root sequence using an arbitrary $\mathbb{F}$, such decomposition is almost certainly incorrect, that is to say the decomposition steps are not optimal in each and every node.

In this sense, the root sequence $x[n]$ and the tree $\mathbb{G}$ enjoy a one-way relation, which means that $\mathbb{G}$ can be easily obtained from $x[n]$, but the opposite is very difficult, and it becomes increasingly harder the longer the length of the root sequence. This also holds for the continuous-time MT with a sufficient number of levels. In other words, given just the complete tree information $\mathbb{G}$, it is difficult to guess the right values to be assigned to the leaves in $\mathbb{F}$ to keep the decomposition of the reconstructed sequence which is still consistent with the given $\mathbb{G}$. Using a term typical of hashing applications, such a very rare occurrence constitutes a collision. Thus, the tree information can be thought of as a hash of the root signal.

However, the complete tree information of even moderately short sequences can be very cumbersome. Of course, zeroing or quantizing the coefficients in $\mathbb{F}$, as we suggested in Sec. VI-A, compacts the tree by pruning zeroed subtrees, making it more manageable. Since we expect the collision probability to be very low, we have run some experiments using this last configuration, as the collision probability is bound to increase if more root sequences correspond to the same pruned tree associated with the considered one.



(a) Collision occurrence. (b) Tree analysis after adding noise to one coefficient.



(c) Tree analysis after adding noise to the original signal.

Figure 10: Experiments on one-wayness property.

Fig. 10 shows the experiments performed to highlight the one-wayness property. The results are averaged over 5 000, length 64 sequences (quite short for practical purposes, but able to show noticeable collisions), obtained at random: in particular, they are a combination of pixel values taken from rows and columns of the images of the aforementioned datasets, supplemented by white Gaussian noise sequences.

In Fig. 10a, the collision probability across different input sequences is estimated. The sequences are first decomposed, and then only the number shown on the x-axis of the most significant coefficients are kept, exactly in the same way as for the previous sparsity experiments. Each tree is then pruned and compared to every other one. A collision is declared if any two pruned trees are identical, namely, two root sequences decompose into the same $\mathbb{G}$. The collision probability drops under 1% already at $1/8$ rate (keeping 8 coefficients out of 64). For longer sequences, the same collision probability is expected to be met with even lower rates. Keeping 12 coefficients, the collision probability becomes negligible.

Differently from hash functions, it can be expected that the tree information is invariant to slight modifications of the root sequence, which is a very unique property for a one-way function. In fact, if the original sequence values are just barely modified, besides the already mentioned nodes scaling, the optimal decomposition points may be unchanged all the way to the leaves. Of course, the introduced modifications are still reflected by the nodes (and final leaves) values. The low collision probability that we have just shown implies that the introduced non-scaling modification, *i.e.*, the added noise power, must be very slight. The tolerance of the tree to noise addition is amplified by coefficient quantization followed by zeroed subtree pruning, because fewer decomposition opera-

tions influence the definition of the final tree structure.

To show this interesting facet of the one-wayness, we have run two different experiments. First, we have reported in Fig. 10b the effect of modifying the most significant coefficient after the decomposition is performed, while keeping the whole tree (without pruning). The modification is computed as the SNR on the single affected leaf. In more than 50% of the cases, a 10% change on the most significant leaf value is sufficient to generate the tree structure of the (modified) reconstructed sequence that is different to the original tree. Therefore, the complete tree is quite fragile to changes in coefficient values, as expected, although slight, individual changes are sometimes tolerated.

In the next experiment, depicted in Fig. 10c, white Gaussian noise is added to the input sequences for various SNR. The decomposition tree is computed and successively pruned in the usual way, keeping the most significant coefficients. The number of unchanged trees, *i.e.*, that are not modified by the noise addition, are then counted. The fewer coefficients are kept, the more the trees are pruned. Thus, in this case the tree of the noisy sequence may be the same as the one of the original sequence. As expected, even moderate noise modifies a large percentage of the tree structures, unless very few coefficients are kept.

In conclusion, the decomposition tree can be used as a particular kind of hashing function of the original sequence. Given $\mathbb{G}$ alone, without knowing the original sequence, it is difficult to guess the nodes that would imply the same tree structure. However, it is possible to slightly modify the original sequence and still preserve the decomposition tree, either in a deterministic way by nodes scaling or through random noise addition (however, no methods exist to find a colliding sequence with a predetermined shape). Therefore, the decomposition process is one-way, but it does not strictly possess weak collision resistance.

VII. Conclusions

In this paper, we presented the Mirror Transform (MT), which is a new signal transform based on the iterative application of the even/odd decomposition around optimal (in a decoupling energy sense) mirroring points, valid for both continuous and discrete time domains. In the former case, in principle the ternary decomposition tree constituting the transform is infinitely deep, even for finite support signals, while a finite number of levels is ensured for finite-length sequences.

We listed several properties of the proposed transform, including the fact that each decomposition level preserves the energy of the original signal by distributing it across a set of orthogonal tree nodes. This allows the reconstruction of finite energy signals through a constructive algorithm by approximating leaf nodes while introducing controllable distortion.

We employed the MT in several experiments. The first set aimed at analyzing its sparsity properties, limiting our scope to finite trees obtained from 1D and 2D signals of various origin. Of course, the compactness of the alternative representation is affected by the amount of information concerning the symmetry point employed in each decomposition step. Nevertheless, it is possible to prune the decomposition tree after a given subtree is zeroed through coefficients quantization.

Further experiments investigated the one-way relation between a signal and its associated tree information. Akin to hashing, we have shown that it is indeed very rare to incur into a collision. An interesting feature is that mild modifications to the original signal, when combined with coefficient quantization and tree pruning, is likely to produce the same hash, which is a very peculiar property for a hashing algorithm.

The nature of this work remains very foundational. Through its experimentation, it is hoped that many signal processing applications may benefit from its use, in particular domains. It emphasizes how intrinsic characteristics in the data can be captured to cope with desirable features or robustness to noise.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[2] G. Awad *et al.*, "Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval," *arXiv preprint arXiv:2009.09984*, 2020.

[3] E. Hjelmås and B. K. Low, "Face detection: A survey," *Computer Vision and Image Understanding*, vol. 83, no. 3, pp. 236–274, 2001.

[4] J. Fourier, *Théorie analytique de la chaleur, par M. Fourier*. Chez Firmin Didot, père et fils, 1822.

[5] A. Haar, "Zur theorie der orthogonalen funktionensysteme," *Mathematische Annalen*, vol. 69, no. 3, pp. 331–371, 1910.

[6] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[7] D. Donoho, "Wedgelets: Nearly minimax estimation of edges," *The Annals of Statistics*, vol. 27, no. 3, pp. 859–897, 1999.

[8] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, oct 2012.

[9] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[10] G. Peyré, "Manifold models for signals and images," *Computer Vision and Image Understanding*, vol. 113, no. 2, pp. 249–260, 2009.

[11] A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. on Image Processing*, vol. 1, no. 1, pp. 18–30, 1992.

[12] N. Huang, Z. Shen, S. Long, M. Wu, H. Shih, Q. Zheng, N.-C. Yen, C.-C. Tung, and H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, pp. 903–995, 03 1998.

[13] "Discrete Mirror Transform: code, experiments and results," https://github.com/AlessandroGnutti/DiscreteMirrorTransform/, accessed: 2021-10-01.

[14] A. Gnutti, F. Guerrini, and R. Leonardi, "Representation of signals by local symmetry decomposition," in *Proc. EUSIPCO*, 2015, pp. 983–987.

[15] F. Guerrini, A. Gnutti, and R. Leonardi, "Iterative mirror decomposition for signal representation," in *Proc. ICASSP*, 2019, pp. 5541–5545.

[16] B. Fino and V. Algazi, "Unified matrix treatment of the fast Walsh-Hadamard transform." *IEEE Trans. on Computers*, vol. 11, no. 100, pp. 1142–1146, 1976.

[17] F. Guerrini, A. Gnutti, and R. Leonardi, "Even/odd decomposition made sparse: A fingerprint to hidden patterns," *Signal Processing*, vol. 141, pp. 273–286, 2017.

[18] A. Gnutti, F. Guerrini, and R. Leonardi, "2d discrete mirror transform for image non-linear approximation," in *Proc. ICPR*, 2021, pp. 9311–9317.

[19] "IRIS: Incorporated research institutions for seismology," https://ds.iris.edu/ds/, accessed: 2021-10-01.

[20] A. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. 215–220, 2000.

[21] O. G. Sezer, O. G. Guleryuz, and Y. Altunbasak, "Approximation and compression with sparse orthonormal transforms," *IEEE Transactions on Image Processing*, vol. 24, no. 8, pp. 2328–2343, 2015.

Alessandro Gnutti received the MS degree (cum laude) in Telecommunications engineering and the Ph.D. degree in Information engineering from the University of Brescia, Italy, in 2014 and 2017, respectively. He is currently tenure-track assistant professor with the Department of Information Engineering, University of Brescia. In 2017, he was a visiting fellow at the University of Southern California, working on graph signal processing. His main research interests cover signal and image representation, including transform and graph analysis for image and video compression, and pattern recognition.

Fabrizio Guerrini received the MS degree (cum laude) in Electronic engineering and the Ph.D. degree in Information engineering from the University of Brescia, Italy, in 2004 and 2008, respectively. He is currently a tenure-track assistant professor with the Department of Information Engineering, University of Brescia. His main research interests cover image and video processing and applications, including transform coding and feature extraction, image security and watermarking, and pattern analysis and recognition.

Riccardo Leonardi (S'1979 – M'1988 – SM'2008 – F'2018) received the Diploma and the Ph.D. degrees in Electrical Engineering from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 1984 and 1987, respectively. He has been a researcher at UC Santa Barbara and Bell Laboratories from 1987 till 1991. Since 1992 he was appointed at the University of Brescia, Italy, where he established the Signal, Imaging, Networking and Communications (SINC) Group. He conducts research in signal and image representation for visual communications and visual content protection. He also is an expert in machine learning tools with application to multimedia content analysis, and medical imaging. Prof. Leonardi is as an expert evaluator for the European Commission, and is currently President of the Italian Telecommunication and Information Technology Association (GTTI).