Syndrome- Trellis Codes for Decreasing Preservative Contortion in Digital Cryptography

T.Mahammad Sharief

Assistant Professor, Department of Computer Science and Engineering (AI)

SVR ENGINEERING COLLEGE, Nandyal, Andhra Pradesh, India.

Abstract— Application of covering codes to data embedding improves embedding efficiency and security of steganographic schemes. The Gibbs sampler is the key tool for simulating the impact of optimal embedding and for constructing practical embedding algorithms. Here we will use an algorithm for binary quantization which is dependent on the Belief Propagation algorithm with decimation over factor graphs of Low-Density Generator Matrix (LDGM) codes. In this, we call Bias Propagation (BiP), which can be considered as a special case of the Survey Propagation algorithm proposed for binary quantization. This paper proposes a complete practical methodology for minimizing embedding impact in steganography based on syndrome coding and trellis-coded quantization and contrast its performance with bounds derived from appropriate rate-distortion bounds. Both the payload-limited sender and the distortion-limited sender are considered. The problem is to embed a given payload with minimal possible average embedding impact. Here, we propose a fast and very versatile solution to this problem that can theoretically achieve performance arbitrarily close to the bound. By replacing the individual bits in cover elements, the non-binary cases have been decomposed into several binary cases without losing the performance. To run these in dual domain nature, by the syndrome coding, using the linear convolutional codes with the optimal binary quantizer implemented using the Viterbi algorithm. The time complexity requirements and memory requirements of the embedding algorithm are linear.

Keywords— Trellis–Coded Quantization, Low Density Generator Matrix (LDGM), Belief Propagation Algorithm, Binary Quantization, Embedding Impact, Syndrome Coding.

I. Introduction

Binary quantization is an important problem for lossy source coding and other fields, such as information hiding. The recent work shows that Low Density Generator Matrix (LDGM) codes combined with Survey Propagation (SP) message-passing algorithms can be used to achieve near-optimal binary quantization in practice.

The most important property of any steganographic communication is statistically undetectable. In other words, the warden should not be able to distinguish between cover and stego objects. Due to the in-deterministic nature, steganographers expect that piece of the component can be reinstated with pseudo-random message bits, which obtains a protected steganographic process. At now, the most thriving standard for making practical steganographic systems that embed in experiential covers, such as digital images, is based on minimizing a suitably defined distortion measure. Implementation difficulties and a lack of practical embedding methods have so far limited the application of this principle to a rather special class of distortion measures that are additive over individual cover elements. With the development of near-optimal low-complexity coding schemes, such as the syndrome-trellis codes, this direction has essentially reached its limits. It is our firm belief that further substantial increase in secure payload is possible only when the sender leverages adaptive schemes that place embedding changes based on the local content, that dare to modify pixels in some regions by more than 1, and that consider interactions among embedding changes while preserving higher-order statistics among pixels.

There are mainly two existing approaches to do the steganography in experiential covers, such as digital media objects: steganography is designed to conserve a chosen cover model and minimizing a heuristically-defined inserting contortion. The sturdy dispute for the previous approach is that demonstrable are imperceptible can be achieved with respect to a specific model. The second approach is much realistic—it abandons modelling the cover source and instead tells the steganographers to embed payload while minimizing a distortion function which gives up any ambitions for ideal safety. This seems like to be much costly, it is not, as experiential covers have been disputed to be incognizable [1], which prevents model-preserving approaches from being perfectly secure as well.

In fact, today's least detectable steganographic schemes for digital images [2, 3, 4 and 5] were designed using this principle. Moreover, when the distortion is defined as a norm between feature vectors extracted from cover and stego objects, minimizing distortion becomes tightly connected with model preservation insofar the features can be considered as a low-dimensional model of covers. This type of analysis has been appeared in [5] and [6] and was further developed in [7]. Near-optimal coding schemes for this problem appeared in [8] and [9], together with other clever constructions and extensions [10]–[15].

This paper provides a general methodology for embedding while minimizing an arbitrary additive distortion function with a performance near the theoretical bound. We present a complete methodology for solving both the payload-limited and the contortion-limited sender. The way how the implementation is done described in this paper uses standard signal processing tools convolutional codes with a trellis quantizer and adapts them to our problem by working with their dual representation.

This paper is organized as follows. In the Section II, we begin with LDGM code representation, Bias Propagation and central notion of a distortion function. We recall a connection between steganography and statistical physics by showing that for a given expected embedding distortion, the maximal payload is embedded when the embedding changes follow a particular form of Gibbs distribution in Section III. The syndrome coding method for steganographic communication and the problem of embedding while minimizing distortion is formulated is reviewed in Section IV. In Section V, we show how to decompose the problem of embedding using practical embedding operations to a series of binary problems using a multilayered approach. So that practical algorithms can be realized using binary STCs and by pointing out the limitations of previous approaches. Section VI introduces a class of syndrome-trellis codes for binary embedding operations and the paper is finished in Section VII.

II. LDGM Code Representation, Bias Propagation and Distortion Function

A. *LDGM Code Representation:*

Codes based on sparse generator matrices are duals of LDPC codes. For a given linear code *C* with matrix G ∈ {0, 1}$^{n\times m}$, we define the factor graph of this code as a graph *G = (V,C,E)* with *n check nodes C = {1, 2, … , n}, m information bits V = {1, 2, … , m},* and *n source bits*. An example of a factor graph can be seen in Fig 1. We will use variables *a, b, c* ∈ *C* to denote the check nodes and variables *i, j, k* ∈ *V* to denote the information bits. The factor graph of an LDGM code is obtained randomly using degree distributions from the edge perspectives. The degree of the check node *a* is defined as the number of connected information bits.

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

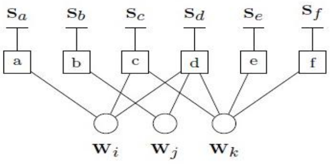


Fig 1: Factor graph representation of a linear code with generator matrix

B. *Bias Propagation Algorithm:*

Let C be an LDGM code with generator matrix $G \in \{0, 1\}^{n\times m}$, $S \in \{0, 1\}^{n}$, a fixed source sequence. The BiP is an iterative message-passing algorithm that performs bitwise MAP estimation of the optical vector for a given source sequence. This is done in rounds and in final round we use factor graph and source sequences to find the most probable bits to be fixed. The estimation of these bits is done by message-passing iterations over the factor graph. In last iteration, the bias messages and the constant source messages are sent from information bits and source bits to connected check nodes. Check nodes send their satisfaction messages to their connected information bits. Finally, the most probable information bits are fixed and removed from the graph by the decimation process.

Algorithm of Bias Propagation:

```
procedure w = BiP(G, s)
  G.B_saa = calc_src_msg(s, gamma)     /*(BiP-1)*/
  G.S_ai = calc_ai(1, G.B_saa)         /*(BiP-4)*/
  while not all_bits_fixed(w)
    bias = BiP_iter(G, s)
    bias = sort(bias)
    if max(|bias|)>t
       num = min(num_max, num_of_bits(|bias|>t))
    else
       num = num_min
    [G,s,w] = dec_most_biased_bits(G,s,w,num)
  end
end
procedure bias = BiP_iter(G, s)
   G.B_saa = calc_src_msg(s, gamma)              /*(BiP-1)*/
   while iter<max_iter
      G.B_ia_old = G.B_ia
      G.B_ia = calc_ia(G.S_ai)                   /*(BiP-2)*/
      if iter>start_damp then
        G.B_ia = damping(G.B_ia, G.B_ia_old)     /*(BiP-3)*/
      end
      G.S_ai = calc_ai(G.B_ia, G.B_saa))         /*(BiP-4)*/
      iter = iter+1
    end
    bias = calc_bias(G.S_ai)                     /*(BiP-5)*/
end
```

C. *Distortion Function:*

This is important for steganography design as we can test the effect of various design choices and parameters and then implement only the most promising constructs. The design of near-optimal schemes for a general D is, however, quite difficult. In this section, we give D a specific local form that will allow us to construct practical embedding algorithms; it will be a sum of local potentials defined on small groups of pixels called cliques. This local form is general enough to capture dependencies among pixels as well as embedding changes while allowing construction of practical embedding schemes.

For concreteness, and without loss of generality, we will call *X* image and x_i its *i*th pixel, even though other interpretations are certainly possible. For example, x_i may represent an RGB triple in a color image, a quantized DCT coefficient in a JPEG file, etc. Let X = $(x_1, x_2,..., x_n) = \{I\}^n$ be an *n-pixel* cover image with the pixel dynamic range *I*. For example, $I = \{0, 1,..., 255\}$ for 8-bit grayscale images. The impact of embedding modifications will be measured using a distortion function D. The sender will strive to embed payload while minimizing D.

$$D(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \rho_i(\mathbf{x}, y_i)$$

Note that ρ_i may arbitrarily depend on the entire cover image X, allowing thus the sender to incorporate inter-pixel dependencies [5]. The boundedness of *D* (x, y) is not limiting the sender in practice since the case when a particular value y_i is forbidden (a requirement often found in practical steganographic schemes [16]) can be resolved by excluding y_i from I_i.

III. Optimality of Gibbs Field

We start by introducing basic concepts, notation, and terminology. The calligraphic font will be used primarily for sets; random variables will be typeset in capital letters, while their corresponding realizations will be in lower-case. Vectors or matrices will be always typeset in boldface lower and upper case, respectively. Although it is certainly applicable to steganography in other objects than digital images, the entire approach is described using the terms “image” and “pixel” for concreteness to simplify the language and to allow a smooth transition from theory to experiments on digital images.

An important premise we now make is that the sender is able to define the distortion function so that it is related to statistically detectable. This assumption is motivated by a rather large body of experimental evidence that indicates that even simple distortion measures that merely count the number of embedding changes correlate well with statistically detectable in the form of decision error of steganalyzers trained on cover and stego images. In general, steganographic methods that introduce smaller distortion disturb the cover source less than methods that embed with larger distortion.

$$E_\pi[D] = \sum_{\mathbf{y} \in \mathcal{Y}} \pi(\mathbf{y}) D(\mathbf{y}).$$

IV. Problem Formulation

The case of embedding with non-additive distortion functions is addressed in [7] by converting it to a sequence of embeddings with an additive distortion. The problem of minimizing the embedding impact for single-letter distortion as described above was formulated, where appropriate bounds were derived. We adhere to the notation defined and define the embedding and extraction mappings as Emb : $\{0, 1\}^n \times \{0,1\}^m \rightarrow \{0, 1\}^n$ and Ext : $\{0,1\}^n \rightarrow \{0, 1\}^m$ satisfying respectively. In particular, we won’t guess the information of the contortion profile ρ_i at the receiver. In practice, we are interested in practical methods that can embed an *m-bit* message in an *n*-element cover, while keeping the expected contortion *E[D(x, Emb(x,m))]* as low as possible.

$$\mathrm{Ext}(\mathrm{Emb}(\mathbf{x}, \mathbf{m})) = \mathbf{m} \quad \forall \mathbf{x} \in \{0,1\}^n, \forall \mathbf{m} \in \{0,1\}^m$$

We assume the sender obtains his/her payload in the form of a pseudo-random bit stream, such as by compressing or encrypting the original message. We further assume that the embedding algorithm associates every cover image with a pair, where the set of all stego images into which can be modified and their probability distribution characterizing the sender’s actions, . Since the choice depends on the cover image, all concepts derived from these quantities necessarily depend on as well. We think of as a constant parameter that is *fixed in the very beginning* and thus we do not further denote the dependency on it explicitly.

While reducing the contortion, the duty of drive in will assume two forms: Payload-limited sender (PLS) and Distortion-limited sender (DLS).

a. Payload-limited sender (PLS):

The sender divides the payload of *m* bits into *s* equal parts of *m/s* bits, computes the local contortions for pixels and embeds the first message. Then, it updates the local contortions of all pixels and embeds the second part, updates the local contortions again, and embeds the next part etc. Because the embedding changes in each sub-lattice do not interact, the embedding can be realized, e.g., using the syndrome- trellis codes. By repeating these embedding sweeps, the introduced embedding pattern will converge to a sample.

$$\underset{\pi}{\text{minimize}}\ E_\pi[D] \qquad \text{subject to } H(\pi) = m.$$

Because each sub-lattice extends over a different portion of the cover image while we split the payload evenly across the sub-lattices may slightly vary. This represents a deviation from the Gibbs sampler. Fortunately, the sub-lattices can often be chosen so that the image does not differ too much on every sub-lattice, which will guarantee that the sets of individual contortions are also similar across the sub-lattices.

b. Distortion-limited sender:

A similar approach can be used to implement the distortion-limited sender with a contortion limit. Consider a simulation of such embedding by a Gibbs sampler with the correct and a sub-lattice. Assuming that, all sub-lattices have the same contortion properties, and then the contortion obtained from cliques containing pixels should be proportional to the number of such cliques.

$$\underset{\pi}{\text{maximize}}\ H(\pi) \qquad \text{subject to } E_\pi[D] = D_\epsilon.$$

The problem of embedding a fixed-size message while minimizing the total contortion (the PLS) is more commonly used in steganography when compared to the DLS. When the contortion function is content-driven, the sender may choose to maximize the payload with a constraint on the overall contortion. This DLS corresponds to a more intuitive use of steganography since images with different level of noise and texture can carry different amount of hidden payload and thus the contortion should be fixed instead of the payload (as long as the contortion corresponds to statistically detectable).

c. Performance Bounds and Comparison Metrics

Both embedding problems described above bear relationship to the problem of source coding with a fidelity criterion as described by Shannon and the problem of source coding with side information available at the transmitter, the so-called Gel'fand–Pinsker problem [18]. The PLS and DLS problems are dual to each other, meaning that the best sharing for the first problem is, for some value of *D*, also optimal for the second one. Following the maximum entropy principle, the optimal solution has the form of a Gibbs distribution (see [8, App. A] for derivation):

$$\pi(\mathbf{y}) = \frac{\exp\left(-\lambda D(\mathbf{y})\right)}{Z(\lambda)} \overset{(a)}{=} \prod_{i=1}^{n} \frac{\exp\left(-\lambda \rho_i(y_i)\right)}{Z_i(\lambda)} \triangleq \prod_{i=1}^{n} \pi_i(y_i)$$

d. Binary Embedding Operation:

The binary case is very important as the embedding method introduced in this paper is first developed for this special case and then extended to non-binary operations. Because the first sum does not depend on *y*, when minimizing *D* over *y* it is enough to consider only the second term. It now becomes clear that embedding in cover *Y* while minimizing is equivalent to embedding in cover Z.

$$\tilde{D}(\mathbf{z},\mathbf{y}) = \sum_{i=1}^{n} \tilde{\rho}_i(\mathbf{z}, y_i) \triangleq \sum_{i=1}^{n} \varrho_i \cdot [y_i \neq z_i]$$

Thus, from now on for binary embedding operations, we will always consider contortion functions of the form:

$$D(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} \varrho_i \cdot [y_i \neq x_i]$$

V. Practical Embedding and Syndrome Coding

A. *Practical Embedding:*

We are now ready to describe a practical embedding algorithm that uses the ideas and theory developed so far. Rather than the description of most general setting, we selected for a simple variant, hoping that generalization to more complex cases will appear transparent to the reader. In this we compare the performance of a specific embedding scheme with other practical embedding algorithms by simulating their optimal performance.

First and foremost, the potentials should measure the detectability of embedding changes. We have substantial freedom in choosing them and the design may utilize reasoning based on theoretical cover source models as well as heuristics stemming from experiments using blind steganalyzers. The proper design of potentials is a complicated subject in it and whose main purpose is introducing a general framework rather than optimizing the design. In this section, we describe an approach inspired by models used in blind steganalysis, where images are projected onto a lower-dimensional feature space carefully selected to model well the noise component of cover images and to be sensitive to embedding changes. Here, a good contortion measure could be some norm of the difference between the cover and stego features.

The features are constructed by considering the differences between neighboring pixels (e.g., horizontally adjacent pixels) as a higher-order Markov chain and taking the sample joint probability matrix (co-occurrence matrix) as the feature. The advantage of using the joint matrix instead of the transition probability matrix is that the norm of the feature difference can be readily upper-bounded by the desired local form.

Another reason for using a high-dimensional feature space is to avoid “overtraining” the embedding algorithm to a low-dimensional model as such algorithms may become detectable by a slightly modified feature set, an effect already reported in the DCT domain.

B. *Syndrome Coding:*

The PLS and the DLS can be realized in practice using a general methodology called *syndrome coding*. In this section, we briefly review this approach and its history paving our way, where we explain the main contribution of this paper—the syndrome-trellis codes. The versatile syndrome-coding approach can also be used to communicate via the wet paper channel using the so-called wet paper codes [19]. Wet paper codes minimizing the number of changed dry pixels were described in [13], [14].

Moreover the other contortion profiles, the linear profile have the great interest to steganography; no general solution with performance close to the bound is currently known. The authors of [2] approached the PLS problem by minimizing the contortion on a block-by-block basis utilizing a Hamming code and a suboptimal quantizer implemented using a brute-force search that allows up to three embedding changes. A similar approach based on BCH codes and a brute-force quantizer was described in [4] achieving a slightly better performance than Hamming codes. Neither Hamming nor BCH codes can be used to deal with the wet paper channel without significant performance loss.

To the best of our knowledge, no solution is known that could be used to solve the PLS problem with arbitrary contortion profile containing wet pixels. Unfortunately, to apply such codes, the number of pixels, *n*, must be very high, which may not be always satisfied in practice. We suppose that the projected syndrome-trellis codes may perform better tradeoffs when used in practical embedding schemes.

VI. Syndrome-Trellis Codes

Syndrome-trellis codes targeted to applications in steganography were described in [17], which was written for practitioners. In this paper, the reader to have a working knowledge of convolutional codes which are often used in data hiding applications such as digital watermarking. For a complete example of the Viterbi algorithm used in the context of STCs, we refer the reader to [17].

A. *From Convolutional Codes to Syndrome-Trellis Codes:*

Convolutional codes in standard trellis representation are commonly used in problems that are dual to the PLS problem, such as the distributed source coding. The main drawback of convolutional codes, when implemented using shift-registers, comes from our requirement of small relative payloads (code rates close to one) which is specific to steganography.

When adapted to the PLS problem, convolutional codes can be used for syndrome coding since the best stego image in can be found using the Viterbi algorithm. These compose convolutional codes appropriate for our application because the entire cover object can be used and the speed can be traded for performance by adjusting the constraint length. By increasing the constraint length, we can achieve the average per-pixel contortion that is arbitrarily close to the bounds and thus make the coding loss approach zero. Convolutional codes are often represented with shift-registers that generate the codeword from a set of information bits.

B. *Description of Syndrome-Trellis Codes:*

Although syndrome-trellis codes form a class of convolutional codes and thus can be described using a classical approach with shift-registers, it is advantageous to stay in the dual domain and describe the code directly by its parity-check matrix. The parity-check matrix of a binary syndrome-trellis code of length *n* and co-dimension *m* is obtained by placing a small sub-matrix of size $h \times w$ along the main diagonal as in Fig. 2. The sub-matrices are placed next to each other and shifted down by one row leading to a sparse and banded. The height of the sub-matrix (called the *constraint height*) is a design parameter that affects the algorithm speed and efficiency.

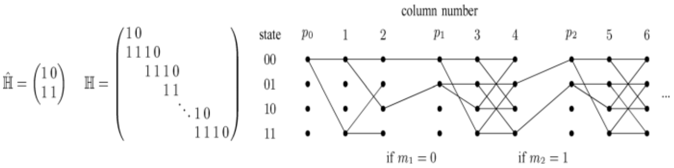
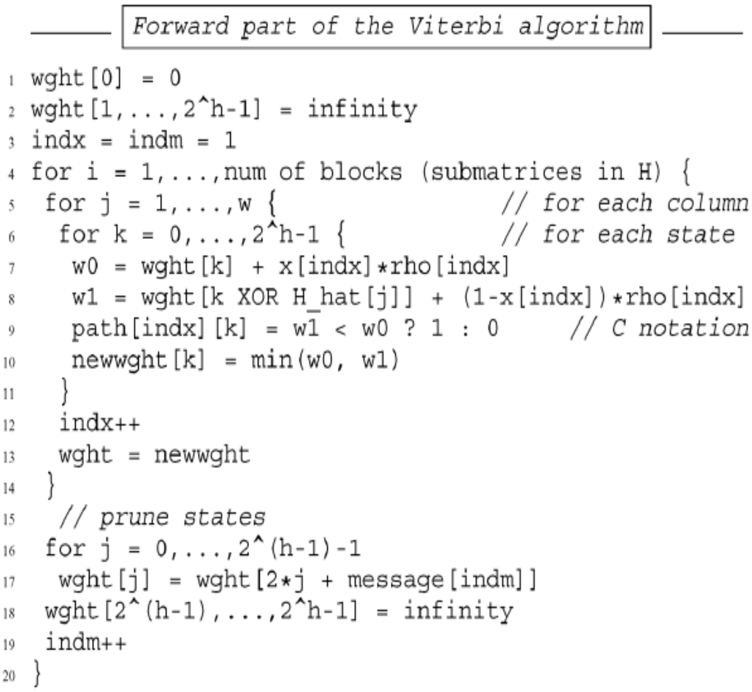


Fig 2: Example of a Parity Check Matrix

Similar to convolutional codes and their trellis representation, every codeword of an STC can be represented as a unique path through a graph called the *syndrome trellis*. Each block of the trellis represents one sub-matrix used to obtain the parity-check matrix. The nodes in every column are called *states*. Each path starts in the leftmost all-zero state in the trellis and extends to the right. The embedding problem for binary embedding can now be optimally solved by the *Viterbi algorithm* with time and space complexity. This algorithm consists of two parts, the *forward* and the *backward* part. The Viterbi algorithm modified for the syndrome trellis is described in Fig. 3 & 4 using a pseudo code.

C. Experimental Consequences:

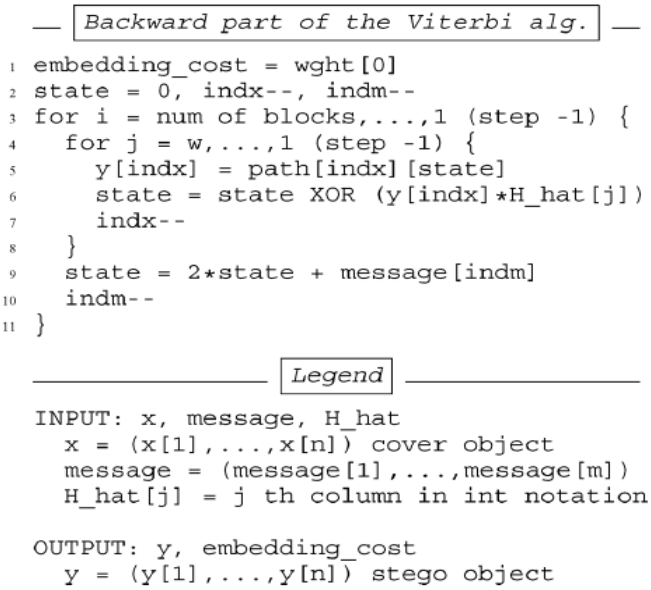
By using Streaming SIMD Extension directives, we have implemented the Viterbi algorithm and then optimized its performance. With the help of the contortion profile, the algorithm decides between the float and 1 byte unsigned integer data type to represent the weight of the paths in the trellis. The following results were obtained using an Intel Core2 X6800 2.93 GHz CPU machine utilizing a single CPU core. The concept of dividing a set of samples into different bins (the so-called binning) is a common tool used for solving many information-theoretic and also data-hiding problems [20]. From this point of view, the steganographic embedding problem is a pure source-coding problem, i.e., given cover *X*, what is the "closest" stego object in the bin indexed by the message. In digital watermarking, the same problem is extended by an attack channel between the sender and the receiver, which calls for a combination of good source and channel codes.



```
Forward part of the Viterbi algorithm

wght[0] = 0
wght[1,...,2^h-1] = infinity
indx = indm = 1
for i = 1,...,num of blocks (submatrices in H) {
 for j = 1,...,w {                 // for each column
  for k = 0,...,2^h-1 {            // for each state
   w0 = wght[k] + x[indx]*rho[indx]
   w1 = wght[k XOR H_hat[j]] + (1-x[indx])*rho[indx]
   path[indx][k] = w1 < w0 ? 1 : 0      // C notation
   newwght[k] = min(w0, w1)
  }
  indx++
  wght = newwght
 }
  // prune states
 for j = 0,...,2^(h-1)-1
  wght[j] = wght[2*j + message[indm]]
 wght[2^(h-1),...,2^h-1] = infinity
 indm++
}
```

Fig 3: Forward Part of Viterbi Algorithm



```
Backward part of the Viterbi alg.
embedding_cost = wght[0]
state = 0, indx--, indm--
for i = num of blocks,...,1 (step -1) {
  for j = w,...,1 (step -1) {
    y[indx] = path[indx][state]
    state = state XOR (y[indx]*H_hat[j])
    indx--
  }
  state = 2*state + message[indm]
  indm--
}

Legend
INPUT: x, message, H_hat
  x = (x[1],...,x[n]) cover object
  message = (message[1],...,message[m])
  H_hat[j] = j th column in int notation

OUTPUT: y, embedding_cost
  y = (y[1],...,y[n]) stego object
```

Fig 4: Backward Part of Viterbi Algorithm

These arrangements can be implemented using nested convolutional (trellis) codes and is better known as Dirty-paper codes. Convolutional codes are attractive for solving these problems mainly because of the existence of the optimal quantizer—the Viterbi algorithm.

VII. CONCLUSION

Recent developments in steganography for real digital media suggest that substantial increase in secure payload can no longer be achieved by improving embedding efficiency of systems that minimize additive embedding contortion, such as the number of embedding changes. As this approach has essentially reached its limits, further increase in secure payload can only be achieved by adaptive embedding algorithms modifying the cover object by larger than minimal amplitudes while minimizing a suitably-defined non-additive contortion function capable of capturing the interaction among embedding changes and preserving inter-pixel dependencies.

The concept of embedding in steganography that minimizes a contortion function is connected to many basic principles used for constructing embedding schemes for complex cover sources today. Many principles for embedding in steganography include the principle of minimal-embedding impact [16], approximate model-preservation [5], or the Gibbs construction [7]. The implicit premise of this paper is the direct relationship between the contortion function *D* and statistical detectability. Designing (and possibly learning) the contortion measure for a given cover source is an interesting problem by itself.

VIII. REFERENCES

[1] R. Böhme, "Improved Statistical Steganalysis UsingModels of Heterogeneous Cover Signals," Ph.D. dissertation, Faculty of Comput. Sci., Technische Universität, Dresden, Germany, 2018.

[2] Y. Kim, Z. Duric, and D. Richards, "Modified matrix encoding technique for minimal distortion steganography," in *Proc. 8th Int. Workshop Inf. Hiding*, J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee, Eds., Alexandria, VA, Jul. 10–12, 2016, vol. 4437.

[3] R. Zhang, V. Sachnev, and H. J. Kim, "Fast BCH syndrome coding for steganography," in *Proc. 11th Int. Workshop Inf. Hiding,*, S. Katzenbeisser and A.-R. Sadeghi, Eds., Darmstadt, Germany, Jun. 7–10, 2019, vol. 5806, Lecture Notes in Computer Science, pp. 31–47.

[4] V. Sachnev, H. J. Kim, and R. Zhang, “Less detectable JPEG steganography method based on heuristic optimization and BCH syndrome coding,” in *Proc. 11th ACM Multimedia Security Workshop*, J. Dittmann, S. Craver, and J. Fridrich, Eds., Princeton,NJ,Sep.7–8, 2019.

[5] T. Pevný, T. Filler, and P. Bas, “Using high-dimensional image models to perform highly undetectable steganography,” in *Proc. 12th Int. Workshop Inf. Hiding*, P. W. L. Fong, R. Böhme, and R. Safavi-Naini, Eds., Calgary, Canada, Jun. 28–30, 2020, vol. 6387.

[6] J. Kodovský and J. Fridrich, “On completeness of feature spaces in blind steganalysis,” in *Proc. 10th ACMMultimedia SecurityWorkshop*, A. D. Ker, J. Dittmann, and J. Fridrich, Eds., Oxford, U.K.

[7] T. Filler and J. Fridrich, “Gibbs construction in steganography,” *IEEE Trans. Inf. Forensics Security*, vol. 5, pp. 705–720, Sep. 2020.

[8] J. Fridrich and T. Filler, “Practical methods for minimizing embedding impact in steganography,” in *Proc. SPIE, Electron. Imag., Security, Steganography, Watermark. Multimedia Contents IX*, E. J. Delp and P. W. Wong, Eds., San Jose, CA, Jan. 29–Feb. 1, 2017, vol. 6505.

[9] T. Filler and J. Fridrich, “Binary quantization using belief propagation over factor graphs of LDGM codes,” presented at the 45th Annu. Allerton Conf. Commun., Control, Comput., Allerton, IL, Sep. 26–28, 2017.

[10] X. Zhang, W. Zhang, and S. Wang, “Efficient double-layered steganographic embedding,” *Electron. Lett.*, vol. 43, pp. 482–483, Apr. 2007.

[11] W. Zhang, S. Wang, and X. Zhang, “Improving embedding efficiency of covering codes for applications in steganography,” *IEEE Commun. Lett.*, vol. 11, pp. 680–682, Aug. 2017.

[12] W. Zhang, X. Zhang, and S. Wang, “Maximizing steganographic embedding efficiency by combining Hamming codes and wet paper codes,” in *Proc. 10th Int. Workshop Inf. Hiding*, , K. Solanki, K. Sullivan, and U. Madhow, Eds., Santa Barbara, CA, Jun. 19–21, 2018, vol. 5284, Lecture Notes in Computer Science, pp. 60–71.

[13] T. Filler and J. Fridrich, “Wet ZZW construction for steganography,” presented at the 1st IEEE Int. Workshop Inf. Forensics Security, London, U.K., Dec. 6–9, 2019.

[14] W. Zhang and X. Zhu, “Improving the embedding efficiency of wet paper codes by paper folding,” *IEEESignal Process.Lett.*,vol. 16, pp.794–797, Sep. 2009.

[15] W. Zhang and X. Wang, “Generalization of the ZZW embedding construction for steganography,” *IEEE Trans. Inf. Forensics Security*, vol. 4, pp. 564–569, Sep. 2019.

[16] J. Fridrich, T. Pevný, and J. Kodovský, “Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities,” in *Proc. 9th ACM Multimedia Security Workshop*, J. Dittmann and J. Fridrich, Eds., Dallas, TX, Sep. 20–21, 2017, pp. 3–14.

[17] T. Filler, J. Judas, and J. Fridrich, “Minimizing embedding impact in steganography using trellis-coded quantization,” in *Proc. SPIE, Electron. Imag., Security, Forensics Multimedia XII*, N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, Eds., San Jose, CA, Jan. 17–21, 2020, vol. 7541, pp. 05-01–05-14.

[18] S. I. Gel’fand and M. S. Pinsker, “Coding for channel with random parameters,” *Problems Control Inf. Theory*, vol. 9, no. 1, pp. 19–31, 2019.

[19] J. Fridrich, M. Goljan, and D. Soukal, “Perturbed quantization steganography,” *ACM Multimedia Syst. J.*, vol. 11, no. 2, pp. 98–107, 2015.

[20] P. Moulin and R. Koetter, “Data-hiding codes,” *Proc. IEEE*, vol. 93, no. 12, pp. 2083–2126, 2015.