Fast Length-Constrained MAP Decoding of Variable Length Coded Markov Sequences over Noisy Channels

Zhe Wang, Xiaolin Wu*and Sorina Dumitrescu
Department of Electrical and Computer Engineering
McMaster University, Hamilton, Ontario, Canada L8S 4K1
zwang@grads.ece.mcmaster.ca xwu/sorina@mail.ece.mcmaster.ca

Abstract - We consider the problem of maximum a posterior probability (MAP) decoding of a Markov sequence that is variable length coded and transmitted over a binary symmetric channel. The number of source symbols in the sequence, if made known to the decoder, can improve MAP decoding performance. But adding a sequence length constraint to MAP decoding problem increases its complexity. In this paper we convert the length-constrained MAP decoding problem into one of maximum-weight k-link path in a weighted directed acyclic graph. The corresponding graph optimization problem can be solved by a fast parameterized search algorithm that finds either the exact solution with high probability or a good approximate solution otherwise. The proposed algorithm has lower complexity and superior performance than the previous heuristic algorithms.

1 Introduction

In practice, due to the constraint of system complexity, the source encoder is almost always suboptimal in the sense that it fails to remove all the redundancy from the source. This residue redundancy makes it possible for the decoder to detect and correct channel errors, even in the absence of channel code. Consider the situation that a scalar-quantized Gaussian Markov source sequence $\{X_i\}$ is compressed by Huffman code that only approaches the self-entropy $H(X_i)$ of the source. The residue redundancy $H(X_{i+1}|X_i) = H(X_i, X_{i+1}) - H(X_i)$ can be used by a MAP decoding scheme that exploits the source memory to combat channel

*This work is supported by Natural Sciences and Engineering Council of Canada, and by National Science Foundation, USA.

noise. The objective of MAP decoding is to estimate the channel input that maximizes the *a posterior probability* of the channel output. In other words, the decoder examines all the possible channel input sequences and finds the one with the maximal a posterior probability.

Throughout the paper a binary symmetric channel (BSC) is assumed. Since the source code is of variable length, a single bit error will most likely desynchronize the encoder and decoder. Therefore, the MAP decoder has to parse the input tentatively and examine all possible parsings in the search for maximum a posterior probability. This greatly increases the decoding complexity. Existing algorithms for this problem are variants of the Viterbi algorithm based on an extended trellis model [4, 7, 10, 11]. Demir and Sayood [4], Park and Miller [7] assumed the number of symbols in the sequence is known, while Subbalakshmi and Vaisey [10], Wang and Wu [11] didn't assume that prior knowledge. The sequence length needs to be transmitted as a side information except implied in specific applications, such as the size of a sample block. The length constraint may be viewed as a form of added redundancy, and thus can improve the MAP decoding performance.

In length-constrained MAP decoding, the decoder needs to find the sequence that maximizes the *a posterior probability* while having a specified number of source symbols. Park and Miller gave an $O(N^2M^2)$ time algorithm (N being the size of the VLC codebook, and M being the length of the received sequence in bits) that can solve this constrained optimization problem exactly [7]. Having admitted this high-complexity algorithm being impractical, they resorted to the heuristics of limiting the number of survived paths at each node of the underlying trellis. This heuristic technique offers a fast approximation solution. Demir and Sayood [4] proposed a similar algorithm for the same problem. Park and Miller [8] described these two approximation algorithms in detail and compared them in performance.

In this paper we reexamine the length-constrained MAP decoding problem. First, in Section 2 we formulate the problem of MAP decoding with and without length constraint.

In Section 3, we induce a weighted directed acyclic graph G to facilitate the development of efficient MAP decoding algorithms. The induced graph G maps the MAP decoding of a VLC-coded Markov sequence of k source symbols to the problem of finding the path with the maximal weight among all paths of k edges (the maximum-weight k-link path). We start our algorithm development by presenting a dynamic programming technique to solve the MAP decoding problem without length constraint. Then we parameterize the graph G to $G(\tau)$, where τ is a real value. Through a binary search on the value of τ, the dynamic programming method can also be applied to this parameterized graph $G(\tau)$, which can either solve exactly or approximately the maximum-weight k-link path problem corresponding to the length-constrained MAP decoding. For most general settings, there is no guarantee of the existence of a parameter τ for any given k. In this case our algorithm can not find the maximum-weight path with the exact number of edges. Fortunately, however, we found that in practice this strategy has a very high probability to solve the problem exactly. Even when failing to find the maximum-weight path of the exact number of edges, the algorithm offers an approximate solution after some adjustment of the number of edges. The proposed approximation algorithm outperforms the approximation algorithm of Park and Miller.

Complexity analysis of various MAP decoding algorithms is also offered in Section 3. Interestingly, if the source sequence is Gaussian-Markov, we can adopt a fast matrix search technique to reduce the complexity of MAP decoding drastically. This is made possible by a so-called Monge property that the objective function of MAP decoding holds for Gaussian-Markov sources.

Finally, the experimental results are presented in Section 4.

2 Problem Formulation

We assume that the input of a binary symmetric channel (BSC) is a scalar-quantized first-order Markov source, which is a first-order discrete Markov process with alphabet $I = \{0, 1, \cdots, N-1\}$. The process X can be completely characterized by transition probabilities $p(i|j) = Pr(X_t = i|X_{t-1} = j)$, for all $i, j \in I$, and the prior probability distribution, $p(i_0) = Pr(X_0 = i_0)$, $i_0 \in I$. The scalar quantizer indexes are compressed by a VLC source encoder, such as Huffman code, whose codebook is $C = \{c_0, c_1, \cdots, c_{N-1}\}$. If the input of the VLC encoder is i, we denote the output of the source coder by $c_i = C(i)$.

A block diagram of the communication system being considered is presented in Fig. 1. An input sequence of K symbols generates a sequence of quantizer indexes $\mathbf{I} = i_0 i_1 \cdots i_{K-1}$ after quantization. The input to the BSC is $C(\mathbf{I}) = C(i_0)C(i_1)\cdots C(i_{K-1})$, which is a binary sequence of length $M = \sum_{t=0}^{K-1} |C(i_t)|$, where $|\cdot|$ is the length of a source codeword in bits. Since the channel has no insertion/deletion errors, the decoder will receive a binary sequence of the same length, denoted by $\mathbf{y} = y_0 y_2 \cdots y_{M-1}$. Given a received binary sequence $\mathbf{y}$ and the side information that K codewords are transmitted, the input must be a binary string of length $|\mathbf{y}|$ that can be parsed into a sequence of K codewords. Let $\mathbf{I} = i_0 i_1 \cdots i_{K-1}$ be the input sequence of K codewords. A permissible parsing of $\mathbf{y}$ with respect to $\mathbf{I}$, denoted by $\mathbf{y}(m_0, m_1, \cdots, m_{K-1} : \mathbf{I})$, is the one such that the subsequence of $\mathbf{y}$ from position m_t to $m_{t+1} - 1$ is the channel output of codeword $C(i_t)$. For a cleaner notation we write the subsequence $y_{m_t} \cdots y_{m_{t+1}-1}$ as $\mathbf{y}[m_t, m_{t+1})$.

Given a received binary sequence $\mathbf{y}$, the decoder should estimate the input $\mathbf{I}$ that maximizes $P(\mathbf{I}|\mathbf{y})$ over all permissible $\mathbf{I}$. Using the Bayesian rule, this is to maximize $\frac{P(\mathbf{y}|\mathbf{I})P(\mathbf{I})}{P(\mathbf{y})}$, which is equivalent to maximizing $P(\mathbf{y}|\mathbf{I})P(\mathbf{I})$. For a particular first-order Markov sequence

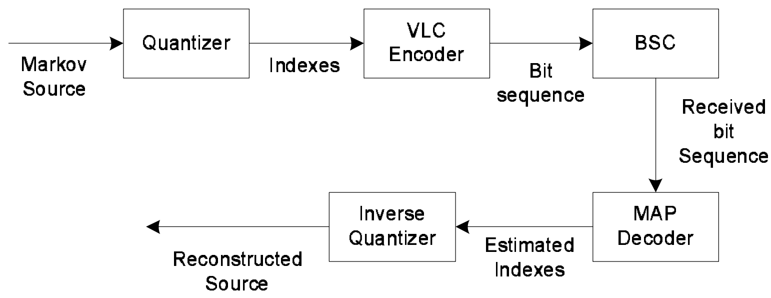

Figure 1: Schematic diagram of transmission system.

$\mathbf{I} = i_0 i_1 \cdots i_{K-1}$, we have

$$P(\mathbf{I}) = p(i_0) \cdot \prod_{t=1}^{K-1} p(i_t | i_{t-1}), \tag{1}$$

and

$$P(\mathbf{y}|I) = \prod_{t=0}^{K-1} P_e(i_t, \mathbf{y}[m_t, m_{t+1})), \tag{2}$$

where

$$P_e(i_t, \mathbf{y}[m_t, m_{t+1})) = p_c^{H_d(C(i_t), \mathbf{y}[m_t, m_{t+1}))} \cdot (1 - p_c)^{|C(i_t)| - H_d(C(i_t), \mathbf{y}[m_t, m_{t+1}))}, \tag{3}$$

with $\mathbf{y}[m_t, m_{t+1})$ being the t^{th} subsequence in the parsing with respect to $\mathbf{I}$, p_c being the crossover probability of the BSC, and $H_d(u, v)$ being the Hamming distance between two binary sequences u and v of the same length. Then the objective of length-constrained MAP decoding is to find

$$\hat{\mathbf{I}} = \arg \max_{\mathbf{I} \in S_K(\mathbf{y})} \log(P(\mathbf{y}|\mathbf{I}) P(\mathbf{I})), \tag{4}$$

where

$$S_K(\mathbf{y}) = \{\mathbf{I} = i_0 i_1 \cdots i_{K-1} \mid |C(\mathbf{I})| = |\mathbf{y}|\} \tag{5}$$

is the set of all possible binary input sequences that consists of K source symbols and have $|\mathbf{y}|$ bits in length. The set of all possible binary input sequences that can possibly produce the output sequence $\mathbf{y}$ via a BSC channel is denoted by

$$S(\mathbf{y}) = \bigcup_{t=K_{min}}^{K_{max}} S_t(\mathbf{y}),$$

where K_{min} and K_{max} are the minimum and maximum possible number of VLC source symbols in an input binary sequence of $|\mathbf{y}|$ bits respectively, i.e., $K_{min} = \lceil \frac{|\mathbf{y}|}{l_{max}} \rceil$ and $K_{max} = \lfloor \frac{|\mathbf{y}|}{l_{min}} \rfloor$, with l_{min} and l_{max} being the length of the shortest and longest codeword of C in bits respectively. The objective function of MAP decoding without length constraint is therefore

$$\hat{\mathbf{I}}' = \arg \max_{\mathbf{I} \in S(\mathbf{y})} \log(P(\mathbf{y}|\mathbf{I})P(\mathbf{I})). \tag{6}$$

3 Graph Representation and Algorithms

3.1 Graph Construction

By expanding the product terms inside the logarithm operator of the objective function (4), we have

$$\hat{\mathbf{I}} = \arg \max_{\mathbf{I} \in S_K(\mathbf{y})} \left\{ \log p(i_0) + \log P_e(i_0, \mathbf{y}[m_0, m_1)) + \sum_{t=1}^{K-1} [\log P_e(i_t, \mathbf{y}[m_t, m_{t+1})) + \log p(i_t|i_{t-1})] \right\}. \tag{7}$$

To facilitate the development of efficient algorithms for the above optimization problem, let us construct a weighted directed acyclic graph (DAG) G in which we can embed all possible parsings of $\mathbf{y}$ with respect to any $\mathbf{I} \in S(\mathbf{y})$. This DAG G has $NM + 1$ vertices, where $M = |\mathbf{y}|$ is the length of the channel output sequence in bits, and N is the size of the VLC codebook. The graph has a unique starting node s, and all other nodes are grouped into M stages. Each stage has N nodes, indexed from 0 to $N-1$, as shown in Fig. 2. Each stage corresponds to a bit location in the received sequence $\mathbf{y}$. Node s is at the 0-th stage. The nodes at the M-th (last) stage are so-called final nodes. Denote by F the set of all final nodes, marked by double circles in Fig. 2. We use n_i^m to label the i-th node at stage m. Node n_i^m corresponds to codeword c_i that is parsed out of $\mathbf{y}[m - |c_i|, m)$ at the bit location m in $\mathbf{y}$. From n_j^m to $n_i^{m+|c_i|}$, there is an edge corresponding to the probability event that the subsequence $\mathbf{y}[m, m + |c_i|)$ is decoded as c_i, given that the previously decoded codeword is c_j. From the starting node s there is an edge to each node $n_i^{|c_i|}$, $0 \le i \le N-1$, corresponding to the probability event that the subsequence $\mathbf{y}[0, |c_i|)$ is decoded as c_i. Generally, for an n_i^m, there are N incoming edges, one from each of the nodes on stage $m - |c_i|$; there are N outgoing edges emitted from n_i^m, one to each node $n_j^{m+|c_j|}$, $0 \leq j \leq N - 1$.

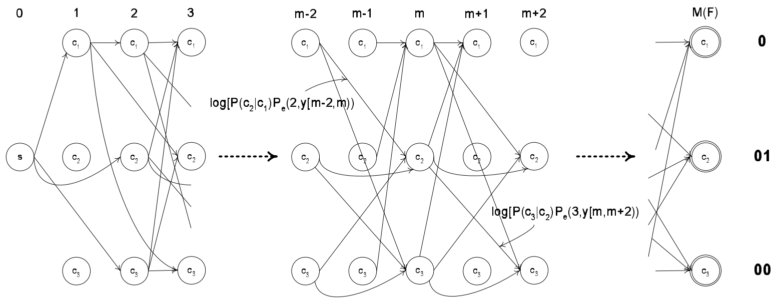

Figure 2: Weighted directed acyclic graph for MAP decoding of a variable length code of a Markov sequence. In this example, the codebook has one 1-bit codeword (c_1), and two 2-bit codewords (c_2 and c_3). For clarity, only directed edges to and from the nodes on stage m are drawn.

With the definition of G, any possible input in $S(\mathbf{y})$ can be mapped to a path from s to F. The weights of outgoing edges emitted from s are $\log p(c_i) + \log P_e(i, \mathbf{y}[0, |c_i|))$, if $\mathbf{y}[0, |c_i|)$ is decoded as c_i, $0 \leq i \leq N - 1$. The weight of the edge from n_j^m to $n_i^{m+|c_i|}$ is assigned to be $\log p(c_i|c_j) + \log P_e(i, \mathbf{y}[m, m + |c_i|))$. Then it follows from (6), (1), (2), and (7) that $\hat{\mathbf{I}}'$ is determined by the maximum-weight path from s to F, and $\hat{\mathbf{I}}$ is determined by the maximum-weight K-link path (i.e., the path of maximal weight over all paths with K edges) from s to F. The problem of MAP decoding without or with length constraint is thus converted to finding the maximum-weight path or the maximum-weight k-link path respectively, in the weighted DAG G, from s to F.

3.2 Solution for MAP Decoding without Length Constraint

If without length constraint, the MAP decoding problem becomes the one of finding the maximum-weight path in the induced weighted DAG. This problem can be solved by dynamic programming. Let $\omega(m, i)$ be the weight of the maximum-weight path from s to the node

n_i^m, then the following recursion holds

$$\omega(m,i) = \max_{0 \le j \le N-1} \{\omega(m - |c_i|, j) + \log(p(c_i|c_j) + \log P_e(i, \mathbf{y}[m - |c_i|, m))\}, \tag{8}$$

for all $0 \le i \le N-1$ and $1 \le m \le M$, with initial values

$$\omega(m,i) = \log p(c_i) + \log P_e(i, \mathbf{y}[0, |c_i|)), \tag{9}$$

for all $m = |c_i|, 0 \le i \le N-1$. Define $\omega(m,i) = -\infty$ if $m < |c_i|$, $0 \le i \le N-1$. Finally, $\omega(M,i)$, $0 \le i \le N-1$, are the weights of the maximum-weight paths from s to individual nodes of the set F. The MAP decoding is determined by

$$\tilde{\omega}(M) = \underset{0 \le i \le N-1}{max} \omega(M,i), \tag{10}$$

which is the maximum-weight path in G. We can reconstruct the MAP decoded sequence $\hat{I}'$ by tracing the path back step by step to s.

Now we analyze the complexity of the dynamic programming algorithm. The cost of (8) is $O(N)$ for fixed m and i, and it amounts to $O(N^2M)$ over the range of $0 \le i \le N-1$ and $1 \le m \le M$. The step of (9) and respectively (10), clearly takes $O(N)$ time. Therefore, the complexity of this algorithm is $O(N^2M)$. Although the graph G has $O(NM)$ nodes and $O(N^2M)$ edges, we do not need to store the entire graph during the process of dynamic programming. Note that the algorithm considers any edge only once. There is no need to store all the $O(N^2M)$ edges. For each node n_i^m at each stage m, we only need to record the edge from the immediate predecessor of n_i^m on the maximum-weight path from s to n_i^m, resulting in a space complexity of $O(NM)$.

3.3 Solution for MAP Decoding with Length Constraint

For the problem of MAP decoding with length constraint, the exact solution corresponds to the maximum-weight path from s to F, among all paths of K edges. Dynamic programming technique can also be applied to find the maximum-weight K-link path [7]. But the time complexity jumps to $O(N^2M^2)$. This complexity is too high to be practical except for very short sequences.

Since the length-constrained MAP decoding problem is equivalent to the maximum-weight k-link path problem in the underlying graph, we can use a parameterized search technique of Aggarwal *et. al* [1] to construct the required path. To this end let us restate below two lemmas proved in [1]. For any real number τ, define a new weighted DAG $G(\tau)$ that is derived from the same sets of nodes and edges as G. The weight of an edge e in $G(\tau)$ is the sum of the weight of e in G, and τ. Then we have:

Lemma 1: If for some real τ, the maximum-weight path from s to F in $G(\tau)$ has k links, then this path is the maximum-weight k-link path from s to F in G.

Thus, if there exists a real number τ such that the maximum-weight path from s to F in $G(\tau)$ has exactly k edges, then this path has maximal weight over all paths from s to F in G, with exactly k edges, which in return solves the length-constrained MAP decoding problem exactly.

Lemma 2: Suppose the maximum-weight path from s to F in $G(\tau)$ has k links. Then for every $\beta < \tau$, the maximum-weight path from s to F in $G(\beta)$ has at most k links.

Lemma 2 implies that binary search can be used to search for the optimal parameter τ. For a general weighted directed acyclic graph, like the DAG G induced by MAP decoding, and a length constraint K, there is no guarantee that such a real value τ exists. In other words, the binary search in τ may not find the specified K-link path. Fortunately, if the algorithm fails to find the optimal parameter, it will converge very quickly to such a τ that the maximum-weight path in $G(\tau)$ is a $(K + \alpha)$-link path, where α is an integer whose absolute value is very small. To reduce the computational complexity, we limit the number of iterations in the binary search to be L, then the overall time complexity is $O(LMN^2)$.

This algorithm does not increase the space complexity.

3.4 Complexity Reduction by Matrix Search

As we saw in the preceding subsections, solving the MAP decoding problem by straightforward dynamic programming has a time complexity $O(N^2M)$ (resp. $O(LN^2M)$). In this section we strive to reduce the complexity to $O(NM)$ (resp. $O(LNM)$) by exploiting some monotonicity of the objective function (8) underlying MAP decoding.

In order to find the maximum-weight path from s to each final node, in the graph G, the sequence of computations in the dynamic programming process can be organized as follows: for each $m, 0 \leq m \leq M-1$, in increasing order, evaluate $\omega(m+|c_i|, i)$ for all $i, 0 \leq i \leq N-1$, with $m + |c_i| \leq M$.

For each $1 \leq m \leq M - l_{max}$, consider the matrix G_m of dimension $N \times N$, with elements $G_m(i,j)$ defined as

$$G_m(i,j) = \omega(m,j) + \log p(c_i|c_j) + \log P_e(i, \mathbf{y}[m, m+|c_i|]), \tag{11}$$

$0 \leq i, j \leq N-1$. Computing all $\omega(m+|c_i|, i)$ for given m and all $0 \leq i \leq N-1$, is equivalent to finding all row maxima of the matrix G_m. A straightforward algorithm can find all row maxima in $O(N^2)$ time, which leads to the solution for MAP decoding without length constraint in $O(N^2M)$ time. Very interestingly, if the matrix G_m satisfies the so-called Monge condition, we can find its row maxima in $O(N)$ time by using the elegant matrix search algorithm [2], and subsequently, we can reduce the total time complexity to $O(NM)$.

In order to apply the fast matrix search technique to speed up the MAP decoding algorithm, we need

$$G_m(i,j) \leq G_m(i,j') \Rightarrow G_m(i',j) \leq G_m(i',j') \tag{12}$$

for $0 \leq i < i' \leq N-1$, and $0 \leq j < j' \leq N-1$ to hold, which is also known as the total monotonicity condition [2]. A sufficient condition for (12) is

$$G_m(i,j') + G_m(i',j) \leq G_m(i',j') + G_m(i,j) \tag{13}$$

for all $0 \leq i < i' \leq N-1$, and $0 \leq j < j' \leq N-1$, which is also known as the Monge condition. By replacing $G_m(i,j)$ according to (11) and doing the cancellations, relation (13) becomes equivalent to

$$\log p(c_i|c_{j'}) + \log p(c_{i'}|c_j) \leq \log p(c_{i'}|c_{j'}) + \log p(c_i|c_j), \tag{14}$$

for $0 \leq i < i' \leq N-1$, and $0 \leq j < j' \leq N-1$. Interestingly, this condition does not depend on the channel statistics, but only on the source statistics.

Moreover, if (14) holds, then the fast matrix search technique can also be applied to solve the maximum-weight path problem in the graph $G(\tau)$. Indeed, the matrix corresponding to G_m is now the matrix $G_{m,\tau}$ with elements:

$$G_{m,\tau}(i,j) = \omega_\tau(m,j) + \log p(c_i|c_j) + \log P_e(i, \mathbf{y}[m, m+|c_i|)) + \tau, \tag{15}$$

$0 \leq i, j \leq N-1$, where $\omega_\tau(m,j)$ denotes the weight of the maximum-weight path from s to the node n_j^m in $G(\tau)$. By a similar argument as above it follows that the maximum-weight path in the graph $G(\tau)$ can be found in $O(NM)$ time if the matrix $G_{m,\tau}$ satisfies the Monge property, i.e.,

$$G_{m,\tau}(i,j') + G_{m,\tau}(i',j) \leq G_{m,\tau}(i',j') + G_{m,\tau}(i,j) \tag{16}$$

for all $0 \leq i < i' \leq N-1$, and $0 \leq j < j' \leq N-1$. Again, by replacing $G_{m,\tau}(i,j)$ according to (15) and doing the cancellations, relation (16) becomes equivalent to (14). Hence, if the latter holds, then the time complexity of MAP decoding with length constraint can be reduced from $O(LMN^2)$ to $O(LMN)$.

Now our attention is turned to the type of Markov source that lends itself to fast MAP decoding by matrix search. We will try to derive sufficient conditions on the joint pdf $f(\cdot,\cdot)$ such that relation (14) holds.

Theorem Relation (14) is valid for all $0 \leq i < i' \leq N-1$, $0 \leq j < j' \leq N-1$, when the joint pdf $f(\cdot,\cdot)$ is Gaussian.

The proof of the theorem is given in [12]. In summary, the MAP decoding problem without or with length constraint for Markov sequences can be solved in $O(NM)$ time or $O(LNM)$ time respectively, for BSC, if the joint pdf $f(\cdot,\cdot)$ satisfies the condition (14). An instance of this class is scalar quantized Gaussian Markov sequence.

4 Experimental Results

The simplest performance measure of a MAP decoder is a symbol-by-symbol difference (inner product) such as PSNR in the case of signal compression. However, for most applications a meaningful distance between two sequences is not as simple as pairwise distortion, even if they have the same length. For instance, a MAP-decoded sequence with errors may contain subsequences that are in the original input sequence only with some shifts. But a symbol-by-symbol distortion measure may mistakenly quantify these subsequences as complete loss. Instead, a string edit distance is more appropriate to measure the performance of MAP decoding. In the following evaluation of MAP decoding performance, a decoded sequence is first aligned to the original sequence by minimizing the Levenshtein distance between them. This is to find an alignment scheme with the minimum number of insertions, deletions and substitutions of symbols to transform the decoded sequence to the original one.

Suppose after the minimum edit distance alignment, the estimated MAP sequence $\hat{\mathbf{I}}$ is adjusted to $\tilde{\mathbf{I}} = \cdots s_i s_d s_j \cdots$, where the subsequences s_i and s_j agree with the original input symbol by symbol, but s_d differs from the original in all of its symbols. Then we count s_d

as one error propagation of length $e_l = |s_d|$. We use the mean error propagation length $\bar{e}_l$, and the number of error propagations, e_n as the measures of MAP decoding performance. The former measures the length of burst errors due to loss of synchronization of the VLC decoding, while the latter measures the frequency of desynchronization. Of course, $\bar{e}_l e_n$ equals to the total number of decoding errors after the alignment of minimum edit distance.

If our MAP decoding algorithm with length constraint fails to find the optimal solution within L iterations, it will stop with an approximate solution that has $K + \alpha$ codewords. Nevertheless, we can still compare its performance with the approximation algorithm in [7], and our own MAP decoding algorithm without length constraint in [11].

In our experiments MAP decoding is applied to a scalar-quantized (uniform with nine codecells) zero-mean, unit-variance, first-order Gaussian-Markov process of correlation coefficient 0.9. The performance results of the MAP decoding algorithms being evaluated were averages of 1000 simulation runs on test sequences of different lengths generated by the above source model. The test sequences were encoded at an average rate of about 3 bits/sample and were transmitted through a BSC of various crossover probabilities p_c. In Fig. 3, we plot the mean error propagation length $\bar{e}_l$ of different MAP decoding algorithms versus crossover probability p_c. In order to evaluate the effects of side information on MAP decoding performance three groups of curves for different sequence lengths $K = 50, 100, 500$ are plotted. The experiments show that the maximum-weight k-link path algorithm (i.e., our algorithm described in subsection 3.3) has the lowest $\bar{e}_l$ among the three MAP decoding algorithms. The MAP decoding algorithm based on the maximum-weight k-link path also outperforms the other two MAP decoding algorithms when measured by the number of error propagations. In Fig. 4 we plot the number of error propagations e_n as a function of p_c for two different sequence lengths $K = 50, 100$. For sequences of 500 and 1000 symbols, the relative ranking of the three different algorithms remains the same.

In Fig. 5 we plot the probability that the maximum-weight k-link path algorithm solves the length-constrained MAP decoding problem exactly. It can be observed that the proposed MAP decoding algorithm has higher than 0.9 probability to obtain the globally optimal solution when $p_c \leq 10^{-2}$ and the sequence length $K \leq 100$.

Another observation to be made from our experiments is that the length-constrained MAP decoding algorithms perform better for smaller sequence lengths K. This should be expected since short sequence length means stronger side information on per-symbol basis.

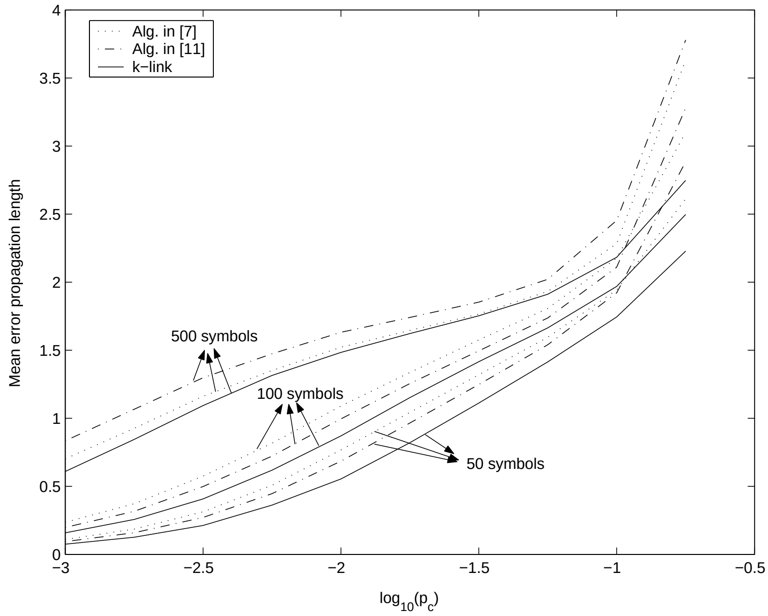

Figure 3: Performances of different algorithms measured by mean error propagation length $\bar{e}_l$ versus crossover probability p_c and for different sequence lengths.

References

[1] A. Aggarwal, B. Schieber, and T. Tokuyama, "Finding a minimum-weight k-link path in graphs with the concave monge property and applications", *Discrete & Computational Geometry* , vol. 12, pp. 263-280, 1994.

[2] A. Aggarval, M. Klave, S. Moran, P. Shor, and R. Wilber, "Geometric applications of a matrix-searching algorithm", *Algorithmica*, 2(1987), pp. 195-208.

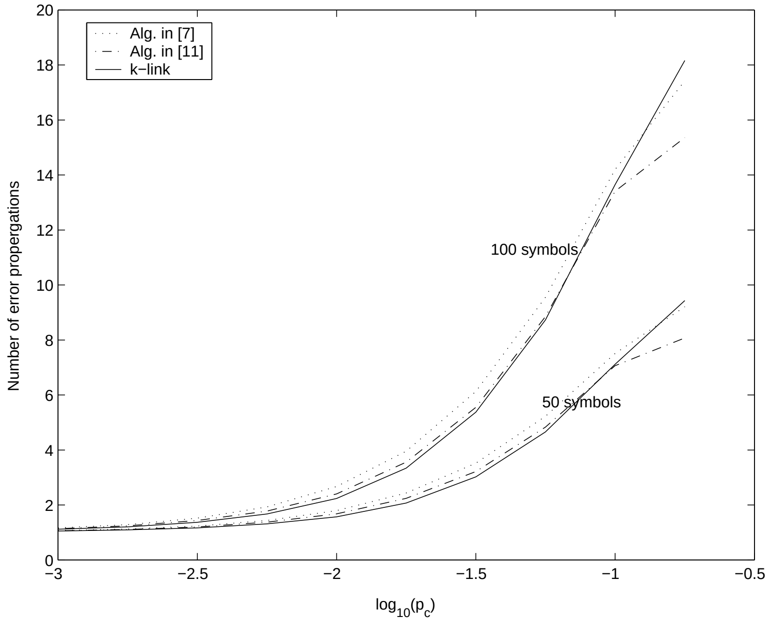

Figure 4: Performances of different algorithms measured by the number of error propagations versus crossover probability and for different sequence lengths.

[3] L. B. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. Infor. Theory*, vol. 20, pp. 762-764, Nov. 1974.

[4] N. Demir and K. Sayood, "Joint source/channel coding for variable length codes," in *Proc. of IEEE Data Compression Conf.* , pp. 139-148, 1998.

[5] K. Sayood and J. C. Borkenhagen, "Use of Residual reddundancy in the design of joint source/channel coders", *IEEE Trans. Commun.* , vol. 39, pp. 838-846, Mar. 1994.

[6] C. E. Shannon, "A mathematical theory of communication", *Bell Systems Technical Journal*, vol. 27, pp. 379-423 and pp. 623-656, 1948.

[7] M. Park and D. J. Miller, "Decoding entropy-coded symbols over noisy channels by MAP sequence estimate for asynchronous HMMs", in *Proc. of Conf. Information Sciences and Systems*, Princeton, NJ, pp. 477-482, Mar. 1998.

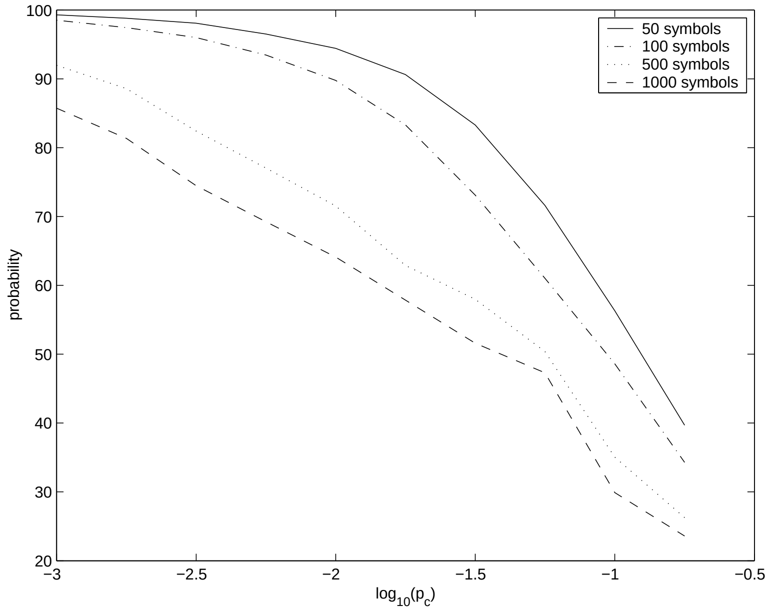

Figure 5: The probability that the maximum-weight k-link path algorithm obtains the globally optimal solution of the length-constrained MAP decoding problem for different sequence lengths.

[8] M. Park and D. J. Miller, "A joint source-channel decoding for variable-length encoded data by exact and approximate map sequence estimation", *IEEE Trans. Commun.* , vol. 48, Jan. 2000.

[9] N. Phamdo and N. Farvardin, "Optimal detection of discrete Markov sources over discrete memoryless channels - application to combined source channel coding", *IEEE Trans. Inform. Theory*, vol. 40, pp. 186-193, Jan. , 1994.

[10] K. P. Subbalakshmi and J. Vaisey, "On the joint source-channel decoding of variable-length encoded sources: the BSC case", *IEEE Trans. Commun.* , vol. 49, NO. 12, Dec. 2001.

[11] Z. Wang and X. Wu, "MAP decoding of variable length length code with substitution, insertion and deletion", in *Proc. of IEEE Workshop on Information Theory*, pp. 287-290, Paris, France, April, 2003.

[12] X. Wu, S. Dumitrescu and Z. Wang, "Monotonicity-based Fast Algorithms for MAP Estimation of Markov Sequences over Noisy Channels", submitted to *IEEE Trans. Inform. Theory.*