Maximum likelihood decoding of trellis codes in fading channels with no receiver CSI is a polynomial-complexity problem

Chun-Hao Hsu and Achilleas Anastasopoulos
Electrical Engineering and Computer Science Department
University of Michigan, Ann Arbor, MI

I. Introduction

The problem of optimal decoding of a trellis coded sequence transmitted over a frequency non-selective, time-selective fading channel is considered in this paper. It is a well-known fact that when the channel state information (CSI) is known to the receiver, the receiver may use Viterbi's algorithm (VA) to £nd the maximum a posteriori probability sequence detection (MAPSqD) solution with linear complexity in sequence length, N. However, when CSI is not available at the receiver, the MAPSqD solution cannot be obtained using such a simple dynamic programming technique, due to memory imposed on the observation by the channel. One approach for solving the problem is to transmit regularly spaced pilot symbols. The receiver can estimate the channel using the pilots and then use VA to decode the sequence based on the estimated CSI. Although this might be a desirable approach for high SNR applications, the unreliable estimated CSI provided by pilots may substantially deteriorate the performance when the operating SNR is low, e.g., when high-performance codes are used. In this case, joint sequence decoding and channel estimation (i.e., true MAPSqD in the presence of unknown CSI) appears to be the desirable policy.

There is an extensive literature on approximate algorithms for solving the joint decoding/estimation problem. The expectation-maximization (EM) algorithm [1, 2] performs a two-step statistical iteration between channel-conditioned sequence decoding and data-conditioned channel estimation. A family of algorithms can be constructed by viewing this problem as a hypothesis testing problem with each hypothesis (sequence) being a path in a tree of depth N. Since testing all hypotheses amounts to exponential complexity, a tree-pruning algorithm, such as the T-algorithm [3], the M-algorithm [4], or the per-survivor processing (PSP) [5] algorithm can be employed to trade off complexity for performance. **In all these works, the underlying assumption was that the optimal (exact) MAPSqD solution can only be found with an exponential complexity in the sequence length N due to the exponential growth of the sequence tree.**

It is our intention to prove that the exact MAPSqD solution can be obtained with only polynomial complexity in the sequence length, N. The authors have addressed and solved this problem in the case of uncoded sequences for a class of channel models in [6–8]. The basic idea behind this work is that although the sequence tree (and thus the number of hypotheses) grows exponentially in N, there is only a certain number of sequences that are potential candidates for the MAPSqD solution. This suf£cient set of sequences is not known a-priori, but once the noisy observation is obtained at the receiver, there is a polynomial-complexity algorithm to obtain it [6–8]. This algorithm is derived by de£ning a new kind of "decision regions" that partition the channel parameter space, as opposed to the traditionally de£ned decision regions that partition the observation space. By studying the structure of these new "decision regions" the authors showed that their number grows only polynomially with N and that there is a polynomial complexity algorithm that constructs them. Unfortunately, all arguments used in [6–8] rely heavily on the assumption of uncoded sequence. In this paper, in order to solve the trellis coded MAPSqD problem, we adopt the concept of "decision regions" de£ned in the parameter space as in [6–8]. Contrary to the previous works, however, we de£ne a set of suf£cient survivor sequences and study their evolution in time. In particular, we show that this set can be updated in a forward recursive fashion and that the resulting set grows only polynomially, thus establishing the polynomial-complexity result for the coded case.

The same ideas can be used to de£ne both forward and backward suf£cient survivor sets. This essentially means that *exact* symbol-by-symbol soft decisions (more speci£cally, the messages corresponding to the min-sum algorithm [9]) can also by generated with polynomial complexity. Applications that can potentially bene£t from this development include serially concatenated convolutional codes through ¤at-fading channels, where now the entire system consisting of the inner trellis code and the channel has an exact "soft inverse" [10] [11, p. 85] that is computationally feasible.

II. System Model and Problem Formulation

Consider the transmission of a sequence of information symbols $\mathbf{a} = [a_1, a_2, \ldots, a_N]^T$, with $a_k \in \mathcal{A} \triangleq \{0, 1, \ldots, K-1\}$. The sequence is encoded by a £nite state machine (FSM) de£ned by its state $s_k \in \mathcal{S} \triangleq \{1, \ldots, I\}$ at time k. For a given state s_{k-1} and a current input a_k, de£ne $e_k \triangleq (s_{k-1}, a_k) \in \mathcal{E} \triangleq \{1, \ldots, IK\}$ to be a trellis edge. The FSM is de£ned by the "next-state" function $ns : \mathcal{E} \rightarrow \mathcal{S}$

$$s_k = ns(e_k), \quad \text{initial state } s_0 \text{ is known} \tag{1}$$

and the "output function" $out : \mathcal{E} \rightarrow \mathcal{O} \triangleq \{0, 1, \ldots, M-1\}$, such that the transmitted M-ary phase shift keying (M-PSK) signal is

$$y_k = \sqrt{E_s} e^{j \frac{2\pi}{M} out(e_k)}, \tag{2}$$

with E_s being the symbol energy. It will also be useful to de£ne the "previous-state function" $ps : \mathcal{E} \to \mathcal{S}$, and the "input function" $in : \mathcal{E} \to \mathcal{A}$ as follows

$$s_{k-1} = ps(e_k) \tag{3}$$

$$a_k = in(e_k). \tag{4}$$

The sequence $\mathbf{y} = [y_1, y_2, \ldots, y_N]^T$ is transmitted through a frequency-non-selective/time-selective fading channel. Assuming that the channel remains constant for the entire sequence transmission[1], the observation model can be expressed as

$$\mathbf{z} = c\mathbf{y} + \mathbf{n} \tag{5}$$

where $\mathbf{z} = [z_1, z_2, \ldots, z_N]^T$ is the received signal, $c \sim \mathcal{CN}(0, 1)$ is a complex Gaussian random variable, $\mathbf{n} = [n_1, n_2, \ldots, n_N]^T$ is zero-mean circularly symmetric complex additive white Gaussian noise with covariance matrix $\mathbf{K}_N = N_0 \mathbf{I}_N$.

When no CSI is available at the receiver, i.e., when the realization of c is unknown, the MAPSqD solution to this problem is

$$\hat{\mathbf{a}}_{MAPSqD} = \arg\max_{\mathbf{a} \in \mathcal{A}^N} p(\mathbf{z}|\mathbf{a})p(\mathbf{a}) = \arg\max_{\mathbf{a} \in \mathcal{A}^N} \left\{ \ln p(\mathbf{a}) + \frac{1}{N_0(N_0 + NE_s)} |\mathbf{z}^H \mathbf{y}|^2 \right\} \tag{6}$$

It is fairly well known fact (see for instance [12, p.]) that, due to the linear and Gaussian nature of the observation model, this problem can be expressed in a double maximization form as

$$\hat{\mathbf{a}}_{MAPSqD} = \arg\max_{\mathbf{a} \in \mathcal{A}^N} \max_{c \in \mathcal{C}} \left\{ \ln p(\mathbf{a}) - \frac{1}{N_0} |\mathbf{z} - c\mathbf{y}|^2 - |c|^2 \right\}. \tag{7}$$

Since there is an one-to-one correspondence between $\mathbf{e}$ and $\mathbf{a}$, £nding $\hat{\mathbf{a}}_{MAPSqD}$ is equivalent to £nding $\hat{\mathbf{e}}_{MAPSqD}$ as follows

$$\hat{\mathbf{e}}_{MAPSqD} = \arg\max_{\mathbf{e} \in \tilde{\mathcal{E}}^N} \max_{c \in \mathcal{C}} \sum_{k=1}^{N} L_k(e_k, c) = L^N(\mathbf{e}, c) \tag{8}$$

where

$$L_k(e_k, c) \triangleq \ln p(in(e_k)) - \frac{1}{N_0} |z_k - c\sqrt{E_s} e^{j\frac{2\pi}{M} out(e_k)}|^2 - \frac{|c|^2}{N}, \tag{9}$$

[1]The polynomial complexity result that we will establish is valid even if the channel is time varying, as long as the degrees of freedom in the channel, i.e., the non-zero eigenvalues of the channel covariance matrix, do not increase linearly with N.

$$L^k(\mathbf{e}^k, c) \triangleq \sum_{i=1}^{k} L_i(e_i, c) \tag{10}$$

and

$$\tilde{\mathcal{E}}^k \triangleq \{\mathbf{e}^k : ns(e_i) = ps(e_{i+1}),\ \forall i = 1, 2, \ldots, k-1\} \tag{11}$$

is the set of valid paths up to time k.

III. SUFFICIENT SURVIVOR MATRICES

For a given c, if we de£ne the survivor that ends in state i at time k obtained by the VA as

$$\hat{\mathbf{V}}^k(i|c) = \arg \max_{\mathbf{e}^k \in \tilde{\mathcal{E}}^k : ns(e_k) = i} L^k(\mathbf{e}^k, c),\ \forall i \in \mathcal{S} \tag{12}$$

we have the following lemma.

Lemma *1:*

$$\hat{\mathbf{e}}_{MAPSqD} = \mathbf{V}^N(i|c) \text{ for some } i \in \mathcal{S} \text{ and } c \in \mathcal{C} \tag{13}$$

Proof: See Appendix I. ■

Therefore, if we de£ne $\hat{\mathbf{V}}^k(c) \triangleq [\hat{\mathbf{V}}^k(1|c), \hat{\mathbf{V}}^k(2|c), \ldots, \hat{\mathbf{V}}^k(I|c)]^T$ to be the survivor matrix consisting of all survivors that end in different states with parameter c at time k and collect all such survivor matrices in a set

$$\hat{\mathcal{D}}^k \triangleq \{\hat{\mathbf{V}}^k(c) : c \in \mathcal{C}\} \tag{14}$$

then **instead of searching $\hat{\mathbf{e}}_{MAPSqD}$ in $\tilde{\mathcal{E}}^N$, we need only search through all rows of all survivor matrices in a potentially smaller suf£cient set $\hat{\mathcal{D}}^N$.** However, since $\mathcal{C}$ is an in£nite set, constructing $\hat{\mathcal{D}}^N$ by searching through all $c \in \mathcal{C}$ will require in£nite complexity. We make the following observation: it is sensible to expect that the function $\hat{\mathbf{V}}^k(c)$ remains constant for a range of c's. More rigorously, we can partition $\mathcal{C}$ in such a way that for all c's in each set of the partition, $\hat{\mathbf{V}}^k(c)$ is the same. This leads us to the de£nition of the "parameter-space decision regions"

$$T^k(\mathbf{V}^k) \triangleq \{c \in \mathcal{C} : \hat{\mathbf{V}}^k(c) = \mathbf{V}^k\} \tag{15}$$

de£ned for any valid survivor matrix $\mathbf{V}^k \in \mathcal{D}^k$, where

$$\mathcal{D}^k \triangleq \{[\mathbf{e}^{(1)k}, \mathbf{e}^{(2)k}, \ldots, \mathbf{e}^{(I)k}]^T : \mathbf{e}^{(i)k} \in \tilde{\mathcal{E}}^k,\ ps(e_1^{(i)}) = s_o\ \forall i \in \mathcal{S},\ \text{and}$$
$$e_l^{(i)} \neq e_l^{(j)} \Rightarrow ns(e_l^{(i)}) \neq ns(e_l^{(j)})\ \forall i, j \in \mathcal{S}, 1 \leq l \leq k\} \tag{16}$$

is the set of all valid survivor matrices at time k. The constraints in (16) imply that once survivors merge, they have to stay merged for their entire past. With the introduction of "parameter-space decision regions" it is now clear that the suf£cient set can be constructed as

$$\hat{\mathcal{D}}^k = \{\mathbf{V}^k \in \mathcal{D}^k : T^k(\mathbf{V}^k) \neq \phi\}. \tag{17}$$

In the next section we will show that $\hat{\mathcal{D}}^N$ and $T^N(\mathbf{V}^N)$, $\forall \mathbf{V}^N \in \hat{\mathcal{D}}^N$ can be generated with polynomial complexity, and furthermore, that the size of the resulting suf£cient set $\hat{\mathcal{D}}^N$ is also polynomial in N.

IV. Recursive construction of $\hat{\mathcal{D}}^N$ and $T^N(\mathbf{V}^N)$

De£ne the set of all possible extensions of $\mathbf{V}^k$ as

$$ext(\mathbf{V}^k) \triangleq \{\mathbf{W}^{k+1} \in \mathcal{D}^{k+1} : \mathbf{W}^{k+1}(i, k+1) = e \Rightarrow \mathbf{W}^{k+1}(i) = [\mathbf{V}^k(ps(e)), e]\} \tag{18}$$

where $\mathbf{W}^k(i,j)$ is the j-th element of the i-th survivor in $\mathbf{W}^k$. Also de£ne the follwoing set of channel parameters

$$P^k(\mathbf{V}^k, e) \triangleq \{c \in \mathcal{C} : L^{k+1}([\mathbf{V}^k(ps(e)), e], c) = \max_{e' \in \mathcal{E}: ns(e') = ns(e)} L^{k+1}([\mathbf{V}^k(ps(e')), e'], c)\}. \tag{19}$$

Observe that the set $P^k(\mathbf{V}^k, e)$ is a convex polytope since its boundaries are straight lines in $\mathcal{C}$. It should be clear from this de£nition that for any $c \in P^k(\mathbf{V}^k, e)$, and if the survivor matrix at time k is $\mathbf{V}^k$ the VA (conditioned on c) will choose $[\mathbf{V}^k(ps(e)), e]$ as the survivor extension. Then as a consequence of the VA, we have the following lemma.

***Lemma** 2:*

$$T^{k+1}(\mathbf{V}^{k+1}) = \bigcup_{\mathbf{W}^k : \mathbf{V}^{k+1} \in ext(\mathbf{W}^k)} \left\{ T^k(\mathbf{W}^k) \bigcap_{i \in \mathcal{S}} P^k(\mathbf{W}^k, \mathbf{V}^{k+1}(i, k+1)) \right\} \tag{20}$$

Proof: See Appendix II. ■

Lemma 2 essentially suggests a recursive algorithm for constructing $T^{k+1}(\mathbf{V}^{k+1})$. In addition, the set $\hat{\mathcal{D}}^{k+1}$ can be easily obtained by collecting the $\mathbf{V}^{k+1}$'s with nonempty $T^{k+1}(\mathbf{V}^{k+1})$. However, since the sets $P^k(\mathbf{W}^k, e)$ need to be generated for all $\mathbf{W}^k \in \hat{\mathcal{D}}^k$, the size of $\hat{\mathcal{D}}^{k+1}$ can grow (in the worst case) as $\beta|\hat{\mathcal{D}}^k|$ for some $\beta > 1$, and thus the size of the suf£cient set $\hat{\mathcal{D}}^N$ will be exponential in N.

To overcome this problem, we modify the above algorithm by observing that several survivor matrices $\mathbf{W}^k$ result in the same set $P^k(\mathbf{W}^k, e)$. In particular, if $\mathbf{W}^k$ and $\mathbf{U}^k$ are such that

$$\mathbf{W}^k(i,j) \neq \mathbf{U}^k(i,j) \Rightarrow \mathbf{W}^k(i,j) = \mathbf{W}^k(l,j) \text{ and } \mathbf{U}^k(i,j) = \mathbf{U}^k(l,j)\ \forall i, l \in \mathcal{S}, \tag{21}$$

i.e., if they only differ in positions at which all the survivors merge together, then $P^k(\mathbf{W}^k, e) = P^k(\mathbf{U}^k, e)$, $\forall e \in \mathcal{E}$. Therefore, we can partition $\hat{\mathcal{D}}^k$ into groups $G_1^k, G_2^k, \ldots, G_{\alpha_k}^k$ such that any two survivor matrices that satisfy (21) belong to the same group. Fig. 1 shows an example of this partitioning. With this modi£cation,

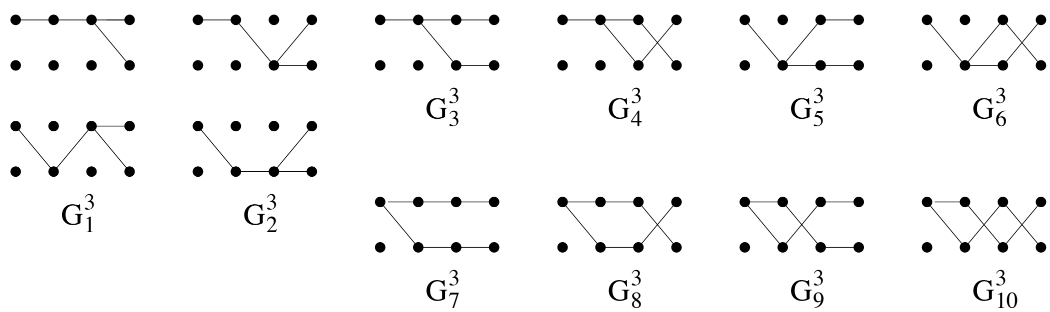

Fig. 1. An example of groups for $I = K = M = 2$ and $k = 3$. Although in this example only G_1^3 and G_2^3 have more than one elements, for larger k the sets G_i^k have a large number of elements.

instead of constructing $P^k(\mathbf{W}^k, e)$ for all $\mathbf{W}^k \in \hat{\mathcal{D}}^k$, we can construct $P^k(\mathbf{W}^k, e)$ for only one $\mathbf{W}^k \in G_i^k$ for each group $i = 1, 2, \ldots, \alpha_k$, thus reducing complexity.

This leads to the following modi£ed algorithm:

1) Construct $T^1(\mathbf{V}^1)$ for all $\mathbf{V}^1 \in \mathcal{D}^1$. If $T^1(\mathbf{V}^1) \neq \phi$, put $\mathbf{V}^1$ into $\hat{\mathcal{D}}^1$.
2) Given $\hat{\mathcal{D}}^k$ and $T^k(\mathbf{V}^k)$ for all $\mathbf{V}^k \in \hat{\mathcal{D}}^k$, construct groups $G_1^k, G_2^k, \ldots, G_{\alpha_k}^k$.
3) For each $i = 1, 2, \ldots, \alpha_k$, do the following.
   a) Choose an arbitrary matrix $\mathbf{W}^k \in G_i^k$, and construct $P_e^k(\mathbf{W}^k), \forall e \in \mathcal{E}$.
   b) For all $\mathbf{V}^{k+1} = ext(\mathbf{U}^k), \mathbf{U}^k \in G_i^k$, if $T^k(\mathbf{U}^k) \bigcap_{i \in \mathcal{S}} P^k_{\mathbf{V}^{k+1}(i,k+1)}(\mathbf{W}^k) \neq \phi$,
      i) Union this set with $T^{k+1}(V^{k+1})$, where $T^{k+1}(V^{k+1})$ is initially set to be ϕ. (Note that $T^{k+1}(V^{k+1})$ is exactly what was de£ned in Lemma 2)
      ii) Put V^{k+1} into $\hat{\mathcal{D}}^{k+1}$.
4) Iteratively do steps 2) and 3) until we get $\hat{\mathcal{D}}^N$ and $T^N(\mathbf{V}^N)$.

At this stage, the proof of the polynomial complexity of the algorithm is only available for the case of $I = 2$. Although this is the simplest case, it has important applications, such as a differentially encoded BPSK system. The authors are currently working on the generalization to the case $I > 2$.

***Lemma** 3:* The number of groups α_k is at most polynomial in k.

Proof: See Appendix III. ■

We summarize the main result of this paper in the following theorem.

***Theorem** 1:* For $I = 2$, the algorithm stated in Lemma 2 and modi£ed using the groups de£ned in (21) can £nd the exact $\hat{\mathrm{e}}_{MAPSqD}$ solution with worst-case polynomial complexity in N for any signal-to-noise ratio.

Proof: A sketch of the proof is now presented. The proof hinges on Lemma 3, which implies that all sets $T^{k+1}(\mathbf{V}^{k+1})$ are polytopes de£ned by a number of equations which is polynomial in N. Since each equation represents a line in the complex plane, the problem becomes equivalent to £nding all partitions generated by a polynomial number of lines in $\mathcal{C}$. This problem has been studied before in [13] where it has been shown that there exists a polynomial-complexity algorithm that can £nd the at-most-polynomial number of such polytopes. The nature of the algorithm is not important here; its existence completes the proof. ■

V. Discussion and Conclusion

In this paper, the problem of optimal MAPSqD of a trellis coded data sequence transmitted over a frequency-nonselective, time-selective channel is considered. The case when the receiver does not have CSI is addressed. It is shown that, contrary to the traditional belief, the exact solution can be obtained with polynomial complexity in the sequence length (the proof for a two-state trellis is only presented). The novel approach we used here to establish these results is to view this detection problem from the channel parameter space as opposed to the observation space and de£ne appropriate decision regions.

We would like to point out that with a small modi£cation one can also solve the more interesting problem of obtaining symbol-by-symbol soft decisions with polynomial complexity. In particular, the metric

$$SbS_k(a) = \max_{\mathbf{a}:a_k=a} \ln\{p(\mathbf{z}|\mathbf{a})p(\mathbf{a})\} = \max_{\mathbf{a}:a_k=a} \left\{ \ln p(\mathbf{a}) + \frac{1}{N_0(N_0+NE_s)}|\mathbf{z}^H\mathbf{y}|^2 \right\} = \max_{\mathbf{e}:in(e_k)=a} \left\{ \max_c L^N(\mathbf{e},c) \right\}, \tag{22}$$

which is exactly the metric implied by the min-sum algorithm can be obtained using the following idea. Recall that in the proposed algorithm, we have the suf£cient sets $\hat{\mathcal{D}}^k$ of forward survivor matrices for all time instants k. Similarly, we can construct the suf£cient sets $\hat{\mathcal{B}}^k$ of backward survivor matrices at all time instants. Therefore for any given edge e_k we can always £nd its best past and future evolution and the corresponding soft metric by comparing all the possible combinations of all the past and future survivors in the forward $\hat{\mathcal{D}}^{k-1}$ and backward $\hat{\mathcal{B}}^{k+1}$ suf£cient sets respectively.

Currently we are working on generalizing Theorem 1 for $I > 2$. For this we might need a more precise description of the evolution of groups in order to £nd α_k exactly, or even a different grouping technique.

Another topic of interest is the design of practical approximate algorithms for performing MAPSqD or generating symbol-by-symbol soft decisions. The authors have derived such algorithms for the uncoded case in [6–8] by introducing a suboptimal, but sensible, partitioning of the parameter space. A similar approach can be followed for the coded case, thus resulting in performance complexity tradeoffs that are more attractive than the ones obtained by the existing approximate algorithms.

APPENDIX I
PROOF OF LEMMA 1

De£ne

$$\hat{i}(c) \triangleq \arg\max_{j\in\mathcal{S}} L(\hat{\mathbf{V}}^N(j|c), c) \tag{23}$$

to be the best survivor given c, we have

$$\hat{\mathbf{V}}^N(\hat{i}(c)|c) = \arg\max_{\mathbf{e}^k\in\tilde{\mathcal{E}}^k} L(\mathbf{e}, c) \tag{24}$$

De£ne further

$$\hat{c}(\mathbf{e}) \triangleq \arg\max_{c\in\mathcal{C}} L^N(\mathbf{e}, c) \tag{25}$$

Since $\hat{c}(\mathbf{e}) \in \mathcal{C}\ \forall \mathbf{e} \in \tilde{\mathcal{E}}^N$, (8) becomes

$$\begin{aligned}\hat{\mathbf{e}}_{MAPSqD} &= \arg\max_{\mathbf{e}\in\tilde{\mathcal{E}}^N} L^N(\mathbf{e}, \hat{c}(\mathbf{e})) \\ &= \hat{\mathbf{V}}^N(\hat{i}(c)|c) \text{ for some } c \in \mathcal{C} \end{aligned} \tag{26}$$

which proves lemma 1.

APPENDIX II
PROOF OF LEMMA 2

By the Viterbi's algorithm we know

$$\hat{\mathbf{V}}^{k+1}(c) \in ext(\hat{\mathbf{V}}^k(c)) \quad \text{and} \tag{27}$$

$$\begin{aligned}&\hat{\mathbf{V}}^{k+1}(i, k+1|c) = e \\ \Rightarrow &\hat{\mathbf{V}}^{k+1}(i|c) = (\hat{\mathbf{V}}^k(ps(e)|c), e) \\ \Rightarrow &L^{k+1}((\hat{\mathbf{V}}^k(ps(e)|c), e), c) \\ = &\max_{e'\in\mathcal{E}:ns(e')=i} L^{k+1}((\hat{\mathbf{V}}^k(ps(e')|c), e'), c),\ \forall i \in \mathcal{S}\end{aligned} \tag{28}$$

Therefore

$$
\begin{aligned}
T^{k+1}(\mathbf{V}^{k+1}) \triangleq &\{c \in \mathcal{C} : \hat{\mathbf{V}}^{k+1}(c) = \mathbf{V}^{k+1}\} \\
=&\{c \in \mathcal{C} : \mathbf{V}^{k+1} \in ext(\mathbf{W}^k), \mathbf{W}^k = \hat{\mathbf{V}}^k(c),\ \mathbf{V}^{k+1}(i, k+1) = e \\
&\Rightarrow L^{k+1}((\mathbf{W}^k(ps(e)), e), c) = \max_{e' \in \mathcal{E} : ns(e') = i} L^{k+1}((\mathbf{W}^k(ps(e')), e'), c),\ \forall i \in \mathcal{S}\} \\
=& \bigcup_{\mathbf{W}^k : \mathbf{V}^{k+1} \in ext(\mathbf{W}^k)} T^k(\mathbf{W}^k) \cap \{c \in \mathcal{C} : \mathbf{V}^{k+1}(i, k+1) = e \\
&\Rightarrow L^{k+1}((\mathbf{W}^k(ps(e)), e), c) = \max_{e' \in \mathcal{E} : ns(e') = i} L^{k+1}((\mathbf{W}^k(ps(e')), e'), c),\ \forall i \in \mathcal{S}\} \\
=& \bigcup_{\mathbf{W}^k : \mathbf{V}^{k+1} \in ext(\mathbf{W}^k)} \left\{ T^k(\mathbf{W}^k) \bigcap_{i \in \mathcal{S}} P^k_{\mathbf{V}^{k+1}(i,k+1)}(\mathbf{W}^k) \right\}
\end{aligned}
$$

which proves lemma 2.

APPENDIX III
PROOF OF LEMMA 3

Suppose $I = 2$.

Let $i \to j$ denote a transition from state i to state j, $\mathcal{E}^{i \to j} \subset \mathcal{E}$ be the set of edges going from state i to state j and

$$\hat{e}_l^{i \to j}(c) \triangleq \arg \max_{e \in \mathcal{E}^{i \to j}} L_l(e, c) \tag{29}$$

be the best edge going from state i to state j for a given c at time l. For any give i, j and l, consider the partition $J_l^{i \to j}$ of $\mathcal{C}$ created by the following set partitioning lines

$$L_l(e, c) = L_l(f, c),\ \forall e \neq f,\ e, f \in \mathcal{E}^{i \to j} \tag{30}$$

Since each line in (30) de£nes a boundary of the two possible results of a pairwise comparison in set $\mathcal{E}^{i \to j}$, $\hat{e}_l^{i \to j}(c)$, as a result of all possible pairwise comparisons in set $\mathcal{E}^{i \to j}$, is a constant for all c in the same element of the partition $J_l^{i \to j}$. Now consider the £ner partition $\hat{J}^k$ created by all the lines in (30) $\forall i, j \in \mathcal{S}$, $1 \leq l \leq k$. Then it follows directly from the above discussion that $\hat{e}_l^{i \to j}(c)$ is a constant for all c in the same element of the partition $\hat{J}^k$, $\forall i, j \in \mathcal{S}$, $1 \leq l \leq k$.

Let another set of partitioning lines

$$L_l(e^{1 \to 2}, c) + L_l(e^{2 \to 1}, c) = L_l(e^{1 \to 1}, c) + L_l(e^{2 \to 2}, c)\ \forall e^{i \to j} \in \mathcal{E}^{i \to j},\ \forall i, j \in \mathcal{S} \tag{31}$$

de£ne the partition $\tilde{J}_l$. Then for all c and c' in the same element of $\tilde{J}_l$, $k \geq l$, $\hat{\mathbf{V}}^k(c)$ and $\hat{\mathbf{V}}^k(c')$, can not be such that

$$\hat{\mathbf{V}}^k(1,l|c) = e^{1\to 2} \in \mathcal{E}^{1\to 2}, \hat{\mathbf{V}}^k(2,l|c) = e^{2\to 1} \in \mathcal{E}^{2\to 1} \quad \text{and} \tag{32}$$

$$\hat{\mathbf{V}}^k(1,l|c') = e^{1\to 1} \in \mathcal{E}^{1\to 1}, \hat{\mathbf{V}}^k(2,l|c') = e^{2\to 2} \in \mathcal{E}^{2\to 2} \tag{33}$$

since

$$\begin{aligned}
&\hat{\mathbf{V}}^k(1,l|c) = e^{1\to 2} \in \mathcal{E}^{1\to 2}, \hat{\mathbf{V}}^k(2,l|c) = e^{2\to 1} \in \mathcal{E}^{2\to 1} \\
\Rightarrow &\sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(1,i|c),c) + L_l(e^{1\to 2},c) \geq \sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(2,i|c),c) + L_l(e^{2\to 2},c), \quad \text{and} \\
&\sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(2,i|c),c) + L_l(e^{2\to 1},c) \geq \sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(1,i|c),c) + L_l(e^{1\to 1},c) \\
\Rightarrow &L_l(e^{1\to 2},c) + L_l(e^{2\to 1},c) = L_l(e^{2\to 2},c) + L_l(e^{1\to 1},c)
\end{aligned} \tag{34}$$

and

$$\begin{aligned}
&\hat{\mathbf{V}}^k(1,l|c') = e^{1\to 1} \in \mathcal{E}^{1\to 1}, \hat{\mathbf{V}}^k(2,l|c') = e^{2\to 2} \in \mathcal{E}^{2\to 2} \\
\Rightarrow &\sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(1,i|c'),c') + L_l(e^{1\to 1},c') \geq \sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(2,i|c'),c') + L_l(e^{2\to 1},c'), \quad \text{and} \\
&\sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(2,i|c'),c') + L_l(e^{2\to 2},c') \geq \sum_{i=1}^{l-1} L_i(\hat{\mathbf{V}}^k(1,i|c'),c') + L_l(e^{1\to 2},c') \\
\Rightarrow &L_l(e^{1\to 1},c') + L_l(e^{2\to 2},c') = L_l(e^{2\to 1},c') + L_l(e^{1\to 2},c')
\end{aligned} \tag{35}$$

contradict with the fact that c and c' is in the same element in $\tilde{J}_l$. Therefore for all c in the same element of $\tilde{J}_l$ the two-to-two transition at time $l \leq k$ for all $\hat{\mathbf{V}}^k(c)$ is uniquely de£ned. If we construct a partition J^k by intersecting all the lines in (31) $\forall 1 \leq l \leq k$ and all the lines partitioning $\hat{J}^k$, then for all c in the same element of J^k, we have the following two facts.

1) $\hat{e}_l^{i\to j}(c)$ is a constant $\forall i,j \in \{1,2\}$, $1 \leq l \leq k$.
2) The two-to-two transitions ($1 \to 2$, $2 \to 1$ or $1 \to 1$, $2 \to 2$) of survivor matrices $\hat{\mathbf{V}}^k(c)$ at all time instants l, $1 \leq l \leq k$ is uniquely de£ned.

As de£ned in section IV, a group at time k is uniquely de£ned by the following information.

1) The merging time $m < k$ before which all the survivors merge together into one tail.
2) The state $m_s \in \mathcal{S} = \{1, 2\}$ to which the only tail connects at the merging time m.
3) The one-to-two transition (trivially $m_s \rightarrow 1$, $m_s \rightarrow 2$) of the survivor matrices at time $m + 1$.
4) The two-to-two transitions ($1 \rightarrow 2, 2 \rightarrow 1$ or $1 \rightarrow 1, 2 \rightarrow 2$) of survivor matrices for all time instants l, $m + 2 \leq l \leq k$.
5) The edges corresponding to all the transitions in 3) and 4).

Since for all c in the same element of J^k, information 4 and 5 are uniquely de£ned, the number of different groups for all $\hat{\mathbf{V}}^k(c)$'s is at most equal to the number of all possible combinations of information 1 and 2 (since 3 is a trivial information), which is $2k + 1$. Therefore the number of groups α_k at time k is at most $(2k + 1)|J^k|$.

To enumerate the size of $|J^k|$, £rst observe that $J_l^{i \rightarrow j}$ is a partition created by $\frac{K(K-1)}{2}$ lines $\forall i, j \in \{1, 2\}$ $\forall 1 \leq l \leq k$. Therefore $\hat{J}^k$ is a partition created by $2^2 k \frac{K(K-1)}{2}$ lines. Moreover, since $\tilde{J}_l$ is created by at most $(\frac{K}{2})^4$ lines $\forall 1 \leq l \leq k$, we conclude that J^k is a partition created by at most $(2^2 k \frac{K(K-1)}{2})((\frac{K}{2})^4 k) = k^2 \frac{K^2(K-1)}{8}$ lines, which is polynomial in k. This completes the proof of Lemma 3.

References

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.

[2] C. N. Georghiades and J. C. Han, "Sequence estimation in the presence of random parameters via the EM algorithm," *IEEE Trans. Communications*, vol. 45, no. 3, pp. 300–308, Mar. 1997.

[3] S. Simmons, "Breadth-£rst trellis decoding with adaptive effort," *IEEE Trans. Communications*, vol. 38, pp. 3–12, 1990.

[4] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey cost analysis," *IEEE Trans. Communications*, vol. 32, pp. 169–176, Feb. 1984.

[5] R. Raheli, A. Polydoros, and C.-K. Tzou, "Per-survivor processing: A general approach to MLSE in uncertain environments," *IEEE Trans. Communications*, vol. 43, no. 2/3/4, pp. 354–364, Feb/Mar/Apr. 1995.

[6] I. Motedayen and A. Anastasopoulos, "Polynomial-complexity noncoherent symbol-by-symbol detection with application to adaptive iterative decoding of turbo-like codes," *IEEE Trans. Communications*, vol. 51, no. 2, pp. 197–207, Feb. 2003.

[7] I. Motedayen and A. Anastasopoulos, "Polynomial complexity ML sequence and symbol-by-symbol detection in fading channels," in *Proc. International Conf. Communications*, Anchorage, Alaska, May 2003, pp. 2718–2722.

[8] I. Motedayen and A. Anastasopoulos, "Optimal joint detection/estimation in fading channels with polynomial complexity," *IEEE Trans. Information Theory*, Sept. 2003, (Submitted; can be downloaded from http://www-personal.engin.umich.edu/∼anastas/preprints.html).

[9] B. J. Frey, *Graphical models for machine learning and digital communications*, MIT Press, Cambridge, MA, 1998.

[10] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Soft-input soft-output modules for the construction and distributed iterative decoding of code networks," *European Trans. Telecommun.*, vol. 9, no. 2, pp. 155–172, March/April 1998.

[11] K. M. Chugg, A. Anastasopoulos, and X. Chen, *Iterative Detection: Adaptivity, Complexity Reduction, and Applications*, Kluwer Academic Publishers, 2001.

[12] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital Communication Receivers: Synchronization, Channel Estimation, and Signal Processing*, John Wiley & Sons, New York, NY, 1998.

[13] H. Edelsbrunner, J. O'Rourke, and R. Seidel, "Constructing arrangements of lines and hyperplanes with applications," *Society for Industrial and Applied Mathematics Journal on Computing*, vol. 15, pp. 341–363, 1986.