On the Computational Complexity of 2D Maximum-Likelihood Sequence Detection

Erik Ordentlich
Hewlett-Packard Laboratories
Palo Alto, CA 94304, U.S.A.
eord@hpl.hp.com

Ron M. Roth*
Computer Science Department
Technion
Haifa 32000, Israel
ronny@cs.technion.edu.il

***Abstract*— We consider a two-dimensional version of the classical maximum-likelihood sequence detection problem for a binary antipodal signal that is corrupted by linear intersymbol interference (ISI) and then passed through a memoryless channel. For one-dimensional signals, this detection problem is well-known to be efficiently solved using the Viterbi algorithm. We show that the two-dimensional version is, in general, intractable, in the sense that a decision formulation of the problem is NP–complete for a certain fixed two-dimensional ISI and memoryless channel with errors and erasures.**

I. Introduction

Consider a two-dimensional (2D) communication system in which an $n \times m$ input signal array $\{x_{i,j} \in \{-1,1\} : 1 \le i \le n, 1 \le j \le m\}$ is corrupted by 2D linear intersymbol interference (ISI) cascaded with a memoryless channel. Specifically, let $\{h_{i,j} \in \mathbb{R} : -N \le i,j \le N\}$ denote the real-valued coefficients of a finite-extent 2D linear filter, and let $W(y|u)$ denote a conditional probability distribution defining the memoryless channel, where $u \in \mathcal{U} \stackrel{\triangle}{=} \{\sum_{-N \le i,j \le N} h_{i,j} z_{i,j} : z_{i,j} \in \{-1,1\}\}$ and $y \in \mathcal{Y}$ where $\mathcal{Y}$ is some channel output alphabet. The conditional distribution of the output $\{y_{i,j}\}$ given the input $\{x_{i,j}\}$ of the 2D communication system under consideration is then

$$P^{(n,m)}(\{y_{i,j}\}|\{x_{i,j}\}) = \prod_{\substack{i,j: 1 \le i \le n, \\ 1 \le j \le m}} W(y_{i,j}| \sum_{r,s: -N \le r,s \le N} h_{r,s} x_{i+r,j+s})$$

where $x_{i,j}$ is set to 1 for $(i,j) \notin \{1,\ldots,n\} \times \{1,\ldots,m\}$.

Given an output $\{y_{i,j}\}$ we are interested in performing 2D maximum-likelihood sequence detection, or, formally, in computing

$$\arg \max_{\{x_{i,j}\}} P^{(n,m)}(\{y_{i,j}\}|\{x_{i,j}\}).$$

We are specifically interested in how the complexity of this computation behaves in the parameters n and m, for fixed $\{h_{i,j}\}$. In the case that, say, m is also fixed, or in the case that there is an i_0 (or j_0) such that $h_{i,j} = 0$ for $i \ne i_0$ (or for $j \ne j_0$), the resulting 2D maximum-likelihood sequence detection problem collapses to a 1D problem, which is well-known to be efficiently solved in polynomial time (actually linear time) by the Viterbi algorithm. In all other cases, it has generally been thought, though, to our knowledge, not explicitly established, that the problem is intractable [1].

In this work, we formally establish this intractability for the specific "L–shaped" 2D ISI filter with

$$h_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in \{(0,0),(-1,0),(0,-1)\}, \\ 0 & \text{otherwise,} \end{cases}$$

and channel W having inputs $\mathcal{U} = \{-3,-1,1,3\}$, outputs $\mathcal{Y} = \{-3,-1,1,3,e\}$, and transition probabilities

$$W(y|u) = \begin{cases} \delta & \text{if } y = e, \\ \epsilon & \text{if } y \ne u \text{ and } y \ne e, \\ 1-3\epsilon-\delta & \text{if } y = u. \end{cases} \quad (1)$$

Thus, the channel W induces erasures (output symbol e) and errors in a symmetrically-distributed fashion.

It is easy to see that when $\epsilon < (1-\delta)/4$, maximum-likelihood sequence detection for the above ISI and channel model is equivalent to

$$\arg \min_{\{x_{i,j}\}} \sum_{\substack{1 \le i \le m \\ 1 \le j \le n}} d(y_{i,j}, x_{i,j} + x_{i-1,j} + x_{i,j-1}), \quad (2)$$

where $d(\cdot,\cdot)$ is a distance metric defined as

$$d(y,u) = \begin{cases} 0 & \text{if } y = e \text{ or } y = u, \\ 1 & \text{otherwise.} \end{cases} \quad (3)$$

Erasures, in particular, have zero distance with respect to all ISI outputs. Clearly, if (2) were tractable so would the following decision version (where we set $n = m$):

Problem 1.1: Given a channel output array $\{y_{i,j}\}$, is there a system input array $\{x_{i,j}\}$ for which

$$\sum_{1 \le i,j \le n} d(y_{i,j}, x_{i,j} + x_{i-1,j} + x_{i,j-1}) = 0 \quad ? \quad (4)$$

Since the condition (4) can be checked in $O(n^2)$ operations for any input array $\{x_{i,j}\}$, it follows that Problem 1.1 is in the complexity class NP. Our main result is to show that Problem 1.1 is, in fact, NP–complete, by establishing a reduction from the classical 3–SAT problem [2]. Specifically, given any Boolean expression involving n' variables, and in conjunctive normal form (product-of-sums), such that each clause contains three variables (possibly negated), we show how to construct, in time polynomial in n', an $n(n') \times n(n')$

*Work done while visiting Hewlett–Packard Laboratories, Palo Alto, CA.

...	$3/1$	$3/1$	$3/1$	$3/1$	$3/1$	$3/1$	...
...	$1/x$	$1/\overline{x}$	$1/x$	$1/\overline{x}$	$1/x$	$1/\overline{x}$	...
			$e/1$	$1/x$	$e/$		
			$3/1$	$1/\overline{x}$			
			$3/1$	$1/x$			
			$3/1$	$1/\overline{x}$			
			$3/1$	$1/x$			
			$\vdots$	$\vdots$			

Fig. 1. Wires: downward T–connection.

			$\vdots$	$\vdots$				
			$3/1$	$1/x$				
			$3/1$	$1/\overline{x}$				
			$3/1$	$1/x$				
			$3/1$	$1/\overline{x}$	$e/1$	$3/1$	$3/1$	...
...	$3/1$	$3/1$	$3/1$	$1/x$	$1/\overline{x}$	$1/x$	$1/\overline{x}$	...
...	$1/\overline{x}$	$1/x$	$1/\overline{x}$	$1/1$				
				$3/1$				

Fig. 2. Wires: upward T–connection.

		e/y
	e/w	$1/y'$
e/x	$1/x'$	$1/z$

x	y	z	x'	y'	w
-1	-1	-1	1	1	1
-1	1	1	1	-1	1
1	-1	1	-1	1	1
1	1	-1	1	1	-1

Fig. 3. Universal gate and truth table.

output array for which (4) holds for some input array if and only if the given Boolean expression is satisfiable.

Our reduction from 3–SAT is based on implementing a Boolean expression as a circuit, involving "logic gates" and "wires," constructed from the $\{y_{i,j}\}$. The underlying $\{x_{i,j}\}$ can be interpreted as "signals" passing through the wires and gates. A subset of indices $V \subset \{(i,j)\}$ will correspond to the variables in the Boolean expression, and one index, (i_0, j_0), will correspond to the output of the Boolean expression. In our reduction, the $y_{i,j}$ for $(i,j) \notin V \cup \{(i_0,j_0)\}$ are chosen so that if $y_{i,j} = e$ for $(i,j) \in V \cup \{(i_0,j_0)\}$, the only input arrays $\{x_{i,j}\}$ for which the distance (4) evaluates to zero are precisely those for which $x_{i_0,j_0} = F(\{x_{i,j} : (i,j) \in V\})$, where F denotes the Boolean expression, and the Boolean 1 (true) and 0 (false) values are mapped, respectively, to $+1$ and -1. The reduction is completed by setting $y_{i_0+1,j_0} = 3$, which forces x_{i_0,j_0} to equal 1.

We note that the above maximum-likelihood sequence detection problem can be cast as a maximum *a posteriori* probability estimation problem in a certain Bayesian belief network. This problem was shown to be NP–complete in [3] for a variety of network constraints, all of which are less constrained than the specific network and joint distribution considered here. The present results are, therefore, not implied by [3].

In the next section, we describe the various components underlying our reduction, such as vertical and horizontal wires, various gates, and inverters. In Section III, we describe how these components are combined to obtain the overall reduction.

II. Reduction components

Figures 1 through 7 illustrate the building blocks of our reduction. The figures depict subarrays of inputs and outputs that can be replicated, extrapolated, and tiled to obtain larger input and output arrays. Most cells of the depicted subarrays contain a pair of values separated by a forward slash, such as $1/x$. The left and right values of the pair correspond respectively to channel output and input arrays. A cell labeled with $1/x$ for example, specifies that the corresponding channel output array location is set to 1 while the input location has the undetermined value $x \in \{-1, 1\}$. In the figures and the sequel we use standard notation from Boolean logic, such as a bar over a variable to denote inversion, and $\wedge$ and $\oplus$ to denote AND and XOR (exclusive OR), where $+1$ and -1, are interpreted as logical 1 and 0, respectively.

For each figure, the specified set of channel output values induces a constraint on input arrays for which the corresponding ISI output satisfies (4). The constrained input values for each figure are denoted by variables and the constraints are denoted using Boolean operations. Thus, in Figure 1 the input cells labeled x and $\overline{x}$ are constrained to be negations of one another if the resulting ISI is to achieve a distance metric of zero relative to the specified output values. The figures depict certain choices of output values that induce constraints which are useful for our Boolean reduction. Figures 1 and 2 represent "wires" which serve to constrain the input values of arbitrarily distant groups of cells. The figures illustrate how "wires" can bend and branch in a variety of directions. Figure 3 implements a variety of logic gates computing Boolean operations on the variables x and y. The truth table in the bottom half of the figure specifies the constraints imposed on the corresponding input variables appearing in the cells. The variable z, for example, is constrained to be $\overline{x \oplus y}$ while the variable y' is constrained to be $\overline{\overline{x} \wedge y}$. Figures 4 and 5 illustrate how "wires" can be attached to the input and output terminals of the "universal gate" of Figure 3 to implement wired binary AND and XOR gates, respectively. Finally, Figures 6 and 7 depict "inverters" that can be used to replace small portions of sufficiently long wires in the horizontal and vertical directions to invert the constraint between input variables at the two ends of the wire.

In all of the figures, an input value of ϕ corresponds to a value that is constrained by the specified output values, but which is not relevant to the reduction. An unspecified input

			$\vdots$	$\vdots$				
			$3/1$	$1/\,y$				
			$3/1$	$1/\,\overline{y}$				
			$3/1$	$1/\,y$	$e/1$	$3/1$	$3/1$	$\ldots$
$\ldots$	$3/1$	$3/1$	e/ϕ	$1/\,\overline{x \wedge y}$	$1/\,x \wedge y$	$1/\,\overline{x \wedge y}$	$1/\,x \wedge y$	$\ldots$
$\ldots$	$1/\,x$	$1/\,\overline{x}$	$1/\phi$	$1/\phi$				

Fig. 4. AND gate with leads.

			$\vdots$	$\vdots$				
			$3/1$	$1/\,y$				
			$3/1$	$1/\,\overline{y}$				
			$3/1$	$1/\,y$				
$\ldots$	$3/1$	$3/1$	e/ϕ	$1/\phi$	$e/1$	$3/1$	$3/1$	$\ldots$
$\ldots$	$1/\,\overline{x}$	$1/\,x$	$1/\phi$	$1/\,x \oplus y$	$1/\,\overline{x \oplus y}$	$1/\,x \oplus y$	$1/\,\overline{x \oplus y}$	$\ldots$

Fig. 5. XOR gate with leads.

				$3/1$				
$\ldots$	$3/1$	$3/1$	$e/\,\overline{x}$	$1/\,x$	$e/\,1$	$3/1$	$3/1$	$\ldots$
$\ldots$	$1/\,\overline{x}$	$1/\,x$	$1/\,1$	$1/\,\overline{x}$	$1/\,x$	$1/\,\overline{x}$	$1/\,x$	$\ldots$

Fig. 6. Horizontal inverter.

		$\vdots$	$\vdots$	
		$3/1$	$1/\,y$	
		$3/1$	$1/\,\overline{y}$	
		$3/1$	$1/\,y$	
		$e/\,\overline{y}$	$1/\,1$	
	$3/1$	$1/\,y$	$1/\,\overline{y}$	
		$e/1$	$1/\,y$	
		$3/1$	$1/\,\overline{y}$	
		$3/1$	$1/\,y$	
		$\vdots$	$\vdots$	

Fig. 7. Vertical inverter.

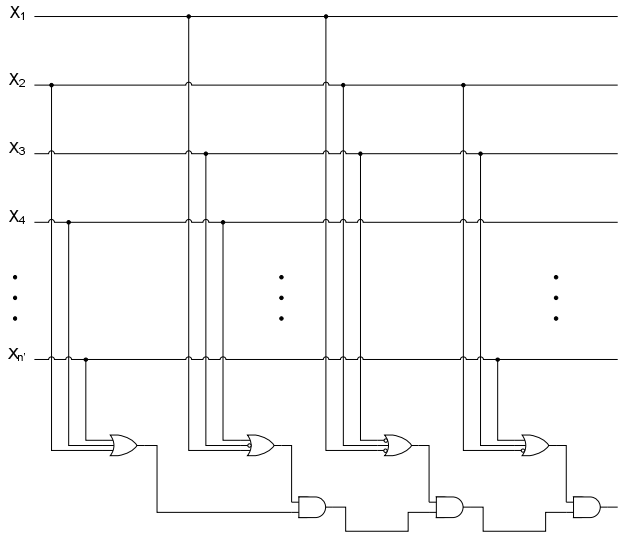

Fig. 8. Schematic for full reduction of 3–SAT Boolean equation.

value for a cell is unconstrained by the specified outputs, and can be assumed to be 1 (e.g. cell containing second e in third row of Figure 1). To avoid cluttering the figures we also avoid specifying input values that are adjacent to the main structures and are obviously constrained to be 1 by neighboring outputs set to 3 (e.g. top row of empty cells in Figure 1). All unspecified output values are assumed to be e.

III. The full reduction

It can be shown that for any Boolean expression (in conjunctive normal form satisfying the 3–SAT restriction), an output array can be constructed from the basic components of the previous section, in time polynomial in the size of the Boolean expression, such that the corresponding set of constrained input array variables constitute a circuit (with "wires," "gates," and "inverters") implementing the Boolean expression. More precisely, the input variable corresponding to the circuit output is constrained to be the Boolean expression evaluated on the input variables corresponding to the circuit inputs, if and only if the associated ISI is within a distance metric of zero from the specified output array. The reduction is completed by constraining the circuit output variable to be 1 by setting to 3 a channel output value whose filter input set contains the circuit output variable.

The basic architecture of the full reduction is depicted in

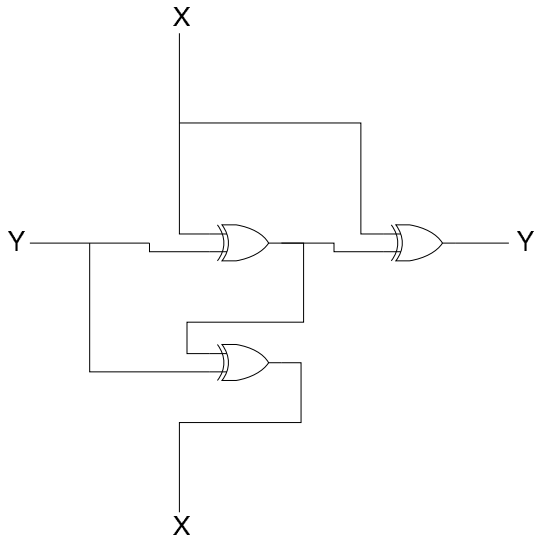

Fig. 9. Crossing wires using XOR gates.

Figure 8. It consists of a set of horizontal wires that propagate the circuit input values and a set of vertical wires that tap into these input values and collect them into sums along the bottom, the outputs of which, in turn, are collected into a product, one output at a time. The OR gate can be implemented using the 2–input AND gate construction of Figure 4 and inversion (Figures 6 and 7). We note that there are places in Figure 8 where horizontal and vertical wires cross. Such a crossing point can be implemented using the XOR gate of Figure 5, as illustrated in Figure 9.

REFERENCES

[1] K. M. Chugg, "Performance of optimal digital page detection in a two-dimensional ISI/AWGN channel," *Proceedings of the 30th Asilomar Conference on Signals, Systems, and Computers*, pp. 958–962, 1996.

[2] M. R. Garey and D. S. Johnson, *Computers and Intractability, A Guide to the Theory of NP–Completeness*, W. H. Freeman and Co. 1979.

[3] S. E. Shimonyi, "Finding MAPs for belief networks is NP–hard," *Artificial Intelligence*, Vol. 68, No. 2, pp. 399-410, August, 1994.