Two-Dimensional Maximum-Likelihood Sequence Detection is NP Hard

Erik Ordentlich, *Fellow, IEEE,* and Ron M. Roth, *Fellow, IEEE*

***Abstract*—A two-dimensional version of the classical maximum-likelihood sequence detection (MLSD) problem is considered for a binary antipodal signal that is corrupted by linear intersymbol interference (ISI) and then passed through a memoryless channel. For one-dimensional signals and fixed ISI, this detection problem is well-known to be solved using the Viterbi algorithm in time complexity that is linear in the sequence length. It is shown here that, in contrast, the two-dimensional MLSD problem is NP hard. Specifically, a decision formulation of the problem is shown to be NP complete for a particular two-dimensional ISI cascaded with either of two memoryless channels: one involving errors and erasures and the other corresponding to additive white Gaussian noise. The proof for the latter case is obtained through a reduction from a still NP complete restricted version of the former. These results are applied to proving the NP completeness of multiuser detection under a Toeplitz constraint—a problem known to be equivalent to a variant of one-dimensional MLSD with growing ISI. This proves a conjecture posed by Verdú in 1989.**

***Index Terms*—Intersymbol interference, maximum-likelihood, multiuser detection, NP completeness, sequence detection, two-dimensional, Viterbi algorithm.**

I. Introduction

For positive integers m and n, let $\mathcal{Q} = \mathcal{Q}_{m,n}$ denote the subset $\{1,2,\ldots,m\} \times \{1,2,\ldots,n\}$ of the integer plane $\mathbb{Z}^2$. Consider a two-dimensional (2D) communication system in which an $m \times n$ real-valued input signal array $X = (x_{i,j})_{(i,j)\in\mathcal{Q}}$ over the bipolar alphabet $\Phi = \{-1,1\}$ is corrupted by 2D linear intersymbol interference (ISI) cascaded with a memoryless channel. Specifically, the ISI is modeled by a 2D linear filter with finite support $\mathcal{S} \subseteq \mathbb{Z}^2$ and real-valued coefficients $(h_{r,s})_{(r,s)\in\mathcal{S}}$; and the memoryless channel is defined by a conditional probability distribution $W(y|u)$, where y is any symbol of some channel output alphabet $\mathcal{Y}$, and u is any real value in the following range of ISI output values obtained for input arrays over Φ:

$$\mathcal{U} = \Big\{ v \in \mathbb{R} \,:\, v = \textstyle\sum_{(r,s)\in\mathcal{S}} h_{r,s}\chi_{r,s} \text{ for some } \chi_{r,s} \in \Phi \Big\}$$

(see Figure 1). The conditional distribution of the channel output $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}}$ of the 2D communication system under consideration given the input $X = (x_{i,j})_{(i,j)\in\mathcal{Q}}$ is then

$$P_{m,n}(Y|X) = \prod_{(i,j)\in\mathcal{Q}} W\left(y_{i,j} \,\middle|\, \textstyle\sum_{(r,s)\in\mathcal{S}} h_{r,s}x_{i+r,j+s}\right),$$

where $x_{i,j}$ is set to 1 for $(i,j) \in \mathbb{Z}^2 \setminus \mathcal{Q}$.

Given a channel output array $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}}$ over $\mathcal{Y}$, we are interested in performing 2D maximum-likelihood sequence detection (MLSD), or, formally, in computing

$$\arg\max_X P_{m,n}(Y|X) \tag{1}$$

(for applications of this problem see [1]–[7]). We are specifically interested in how the complexity of this computation behaves in the parameters m and n, for fixed $(h_{r,s})$. In the case that, say, m is also fixed, or in the case where the elements of $\mathcal{S}$ are collinear points in $\mathbb{Z}^2$, the resulting 2D MLSD problem collapses to a 1D problem, which is well-known to be efficiently solved by the Viterbi algorithm in time complexity that is polynomial (in fact, linear) in the size of the input array [8]. The case of general filter $(h_{r,s})$, conditional distribution W, and both m and n increasing has been conjectured to be intractable [2], though, to our knowledge, not formally proved as such for *any* choice of $(h_{r,s})$ and W.

In this work, we formally establish the intractability of MLSD for an "L–shaped" 2D ISI filter with support $\mathcal{S} = \{(0,0),(-1,0),(0,-1)\}$ and coefficients

$$h_{r,s} = \begin{cases} 1 & \text{if } (r,s) \in \mathcal{S} \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

and two channels, W_1 and W_2, having inputs from the set $\mathcal{U} = \{-3,-1,1,3\}$. The channel W_1 has output alphabet $\mathcal{Y} = \{-3,-1,1,3,\mathrm{e}\}$ and the following conditional probabilities for every $u \in \mathcal{U}$:

$$W_1(y|u) = \begin{cases} \delta & \text{if } y = \mathrm{e} \\ \epsilon & \text{if } y \in \mathcal{Y} \setminus \{u,\mathrm{e}\} \\ 1-3\epsilon-\delta & \text{if } y = u \end{cases}.$$

Thus, the channel W_1 induces erasures (represented by the output symbol e) and errors in a symmetrically-distributed fashion. The channel W_2 corresponds to an additive Gaussian noise channel: in this case, we will restrict attention to a finite set of integer-valued channel outputs.

It is easy to see that when $\epsilon < (1-\delta)/4$, the MLSD in (1) for the ISI (2) and channel model W_1 is equivalent to

$$\arg\min_X \sum_{(i,j)\in\mathcal{Q}_{m,n}} \mathsf{d}(y_{i,j}, x_{i,j} + x_{i-1,j} + x_{i,j-1}), \tag{3}$$

This work was supported in part by Grant No. 2008058 from the US–Israel Binational Science Foundation (BSF), Jerusalem, Israel.

Erik Ordentlich is with Hewlett–Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304, USA.

Email: eord@hpl.hp.com

Ron M. Roth is with the Computer Science Department, Technion, Haifa 32000, Israel. This work was done in part while visiting Hewlett–Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304, USA.

Email: ronny@cs.technion.ac.il

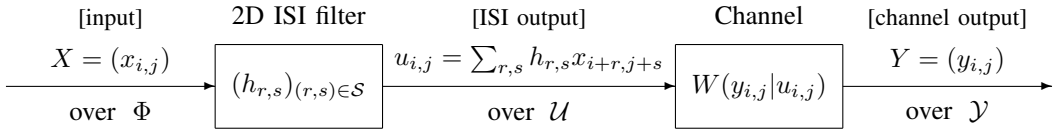


Fig. 1. 2D ISI cascaded with a memoryless channel.

where the minimization in (3) is over input arrays $X = (x_{i,j})_{(i,j)\in\mathcal{Q}_{m,n}}$ over Φ, the boundary values $x_{i,0}$ and $x_{0,j}$ are set to 1, and $\mathsf{d}(\cdot,\cdot)$ is a distance function defined as

$$\mathsf{d}(y,u) = \begin{cases} 0 & \text{if } y \in \{u, \mathrm{e}\} \\ 1 & \text{otherwise} \end{cases}, \quad u \in \mathcal{U}\ ,\ y \in \mathcal{Y}\ .$$

Erasures, in particular, have zero distance with respect to all ISI output values (and in this respect, $\mathsf{d}(\cdot,\cdot)$ deviates from the common definition of a metric). Similarly, MLSD for the Gaussian channel W_2 is well known to be equivalent to

$$\arg\min_{X} \sum_{(i,j)\in\mathcal{Q}_{m,n}} \big(y_{i,j} - (x_{i,j} + x_{i-1,j} + x_{i,j-1})\big)^2\ , \quad (4)$$

where again the minimization is over input arrays $X = (x_{i,j})$ over Φ and $x_{i,0} = x_{0,j} = 1$.

Clearly, if (3) and (4) were tractable (with the latter restricted to rational-valued output sequences), so would the following decision versions (where we set $m = n$ and let $\mathcal{Q}_n$ stand for $\mathcal{Q}_{m,n}$):

Problem 1.1: Given an $n \times n$ channel output array $Y = (y_{i,j})$ over $\{1, 3, \mathrm{e}\}$, does

$$\min_{X} \sum_{(i,j)\in\mathcal{Q}_n} \mathsf{d}(y_{i,j}, x_{i,j} + x_{i-1,j} + x_{i,j-1}) = 0 \quad (5)$$

hold, where the minimization is over input arrays $X = (x_{i,j})$ over Φ and $x_{i,0} = x_{0,j} = 1$?

Problem 1.2: Given an $n \times n$ channel output array $Y = (y_{i,j})$ over $\{0, 1, 2, 3\}$ and integer $T > 0$, does

$$\min_{X} \sum_{(i,j)\in\mathcal{Q}_n} \big(y_{i,j} - (x_{i,j} + x_{i-1,j} + x_{i,j-1})\big)^2 \le T \quad (6)$$

hold, where the minimization is over input arrays $X = (x_{i,j})$ over Φ and $x_{i,0} = x_{0,j} = 1$?

Note that in both problems, the filter $(h_{r,s})$ is fixed to be (2) and is not part of the problem instance. Indeed, it suffices to prove intractability for this particular filter in order to conclude that efficient algorithms for 1D MLSD, such as the Viterbi algorithm, do not have an efficient 2D counterpart that works for all bounded-support filters (unless the complexity class NP collapses to P).

Since the summations in (5) and (6) can be computed in $O(n^2)$ operations for any input array $(x_{i,j})$, it follows that Problems 1.1 and 1.2 are both in the complexity class NP. Our main result is to show that both problems are, in fact, NP complete.

Theorem 1.3: Problems 1.1 and 1.2 are both NP complete.

We prove Theorem 1.3 by establishing a polynomial reduction to Problem 1.1 from the classical 3–SAT problem [9], and then establishing a reduction from a restricted (though still NP complete) version of Problem 1.1 to Problem 1.2. Specifically, for the first step, given any Boolean expression F in n' variables $\boldsymbol{v} = (v_1, v_2, \ldots, v_{n'})$ and in conjunctive normal form (product-of-sums) where each clause contains three variables (possibly negated), we show how to construct, in time complexity that is polynomial in n', arrays $(y_{i,j})_{(i,j)\in\mathcal{Q}_n}$ over $\{1, 3, \mathrm{e}\}$ (where n is polynomially large in n') so that (5) holds for $(y_{i,j})$ if and only if $F(\boldsymbol{v})$ is satisfiable.

The reduction from 3–SAT is based on implementing the Boolean expression $F(\boldsymbol{v})$ as a circuit of the form illustrated in Figure 2, with the corresponding "logic gates" and "wires" formed by the values assigned to the entries of $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}_n}$ (in the figure, we have used conventional logic gate symbols for the Boolean AND, OR, and INVERTER operations, where the latter are depicted as bubbles at the inputs of the OR gates). The underlying input array $X = (x_{i,j})_{(i,j)\in\mathcal{Q}_n}$ (over Φ) can be interpreted as "signals" passing through the wires and gates. A subset $V \subset \mathcal{Q}_n$ of $|V| = n'$ indices will correspond to the n' variables $\boldsymbol{v} = (v_1, v_2, \ldots, v_{n'})$, and one index, (i_0, j_0), will correspond to the Boolean value obtained when $F(\boldsymbol{v})$ is evaluated at the entries of X that are indexed by V, namely, when $(X)_V = (x_{i,j})_{(i,j)\in V}$ is substituted for $\boldsymbol{v}$; here, Boolean "1" (true) and "0" (false) are mapped, respectively, to the real values 1 and -1. In the reduction to Problem 1.1, the array Y is constructed from $F(\boldsymbol{v})$ in such a way that any input array $X = (x_{i,j})$ for which the summation in (5) evaluates to zero must satisfy $x_{i_0,j_0} = F\big((X)_V\big)$. Conversely, each of the $2^{|V|}$ possible assignments (over Φ) to entries that are indexed by V, can be extended into an array X over Φ such that $x_{i_0,j_0} = F\big((X)_V\big)$ and the summation in (5) for that X evaluates to zero. The reduction is completed by setting $y_{i_0+1,j_0} = 3$, which forces x_{i_0,j_0} to equal 1. Most of the remainder of the paper specifies the reduction to Problem 1.1, which is then slightly modified to obtain a reduction to a more restricted form of Problem 1.1, that, in turn, is reduced to Problem 1.2.

We note that the MLSD problems (3) and (4) can be cast as a maximum *a posteriori* probability (MAP) estimation problem in certain Bayesian belief networks (BBNs). The MAP problem for BBNs was shown to be NP hard in [10] for a variety of network constraints, all of which are less constrained than the linear ISI networks and joint distributions considered here. The present results are, therefore, not implied by [10]. Another related work is [11], which proves that multiuser detection, when the intra-user correlation matrix is unconstrained and (along with the channel outputs) is also part of the problem instance, is NP complete. In contrast, we show NP completeness when only the channel outputs comprise the problem instance for a specific 3-tap 2D ISI.

As shown in Section V, NP completeness for the 2D case can also be used to prove an open conjecture in [11] regarding

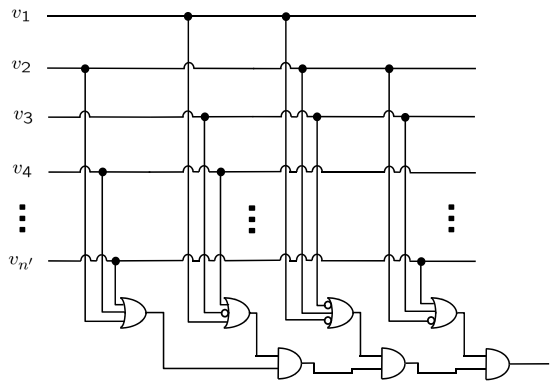


Fig. 2. Schematic of a circuit implementation of an instance of a 3–SAT Boolean expression.

the NP completeness of a constrained version of multiuser detection (shown to be equivalent in [11] to single-user, one-dimensional detection with growing ISI).

In the next section, we describe the various components underlying the reduction from 3–SAT to Problem 1.1, such as vertical and horizontal wires, various gates, and inverters. These correspond to building blocks that will serve to implement circuits following Figure 2. In Section III, we describe how these components are combined to obtain the overall reduction to Problem 1.1. Then, in Section IV, we show how to modify this reduction so that it applies to a restricted version of Problem 1.1, which, in turn, is reduced to Problem 1.2. Thus, Sections II through IV collectively give the proof of Theorem 1.3. In Section V, we show that the NP completeness of a restricted version of Problem 1.2 can be used to prove the previously open Conjecture 1 in [11], stating that multiuser detection with a Toeplitz constraint is NP complete.

II. Components of the reduction

Figures 3 through 9 illustrate the building blocks of the reduction. The figures depict channel output subarrays and respective input subarrays that can be replicated, extrapolated, and tiled to obtain larger output and input arrays. Each cell in each figure corresponds to an entry in a channel output and input subarray as follows. In cells containing a pair of values, the upper-left and lower-right values of the pair, respectively, specify channel output and input array entries. A cell labeled with 1 at the upper-left corner and x at the lower-right, for example, specifies that the corresponding channel output array entry is set to 1 while the input entry has the undetermined value $x \in \Phi$. Only channel output values of 1, 3, and e are used. Empty cells are assumed to specify corresponding channel output array values of e. Cells will be identified by their position in the integer plane $\mathbb{Z}^2$, except that we have elected to rotate the plane ninety degrees clockwise, thus making the coordinates of a cell into standard matrix indexing, where cell (i, j) is the one in row i and column j.

For each figure, the specified set of channel output values induces a constraint on input arrays for which the summation in (5) evaluates to zero, except that the summation is restricted to indices (i, j) for which the channel output entries are specified. In the figures and elsewhere, we use standard notation from Boolean logic to denote such constraints: the values 1 and -1, are interpreted as logical "1" and "0," respectively, a bar over an expression denotes NOT (negation), and "$\wedge$," "$\vee$," and "$\oplus$" denote AND, OR, and XOR (exclusive OR), respectively. In Figure 3, for example, the input in cell $(2, 3)$ is constrained to be 1 by its corresponding channel output of 3. This constrained input value, in turn, along with the output value of 1 in cell $(3, 3)$, constrain the inputs in cells $(3, 2)$ and $(3, 3)$ (labeled x and $\overline{x}$, respectively) to be negations of each other: any choice of equal input values in these cells would force a violation of zero ISI distance at $(3, 3)$. Similar reasoning applies to all of the other adjacent cells along the third row in Figure 3. Collectively, these adjacent constraints impose a global, propagating constraint on the entries of the input array that are indexed by some connected subset of $\mathbb{Z}^2$ formed by the cells with specified channel output values.

The figures depict certain patterns of output values that induce constraints which will be needed in our circuit-based reduction when mapping Boolean expressions to output arrays, essentially along the lines depicted in Figure 2. Figures 3 and 4 represent "wires" which serve to constrain entries of the input array that are positioned in $\mathbb{Z}^2$ arbitrarily far from each other. The figures illustrate how "wires" can bend and branch in a variety of directions, as will be needed for implementing Figure 2. Figure 3, for example, will be used to implement the connections between the horizontal and vertical wires appearing in Figure 2. Non-branching (just bending) wires can be obtained by replacing the output values along any undesired branch with erasures. More specifically, we shall construct bending wires as follows: (a) in Figure 3, setting to empty rows 2 and 3 in columns 1 through 3 (thereby implicitly setting

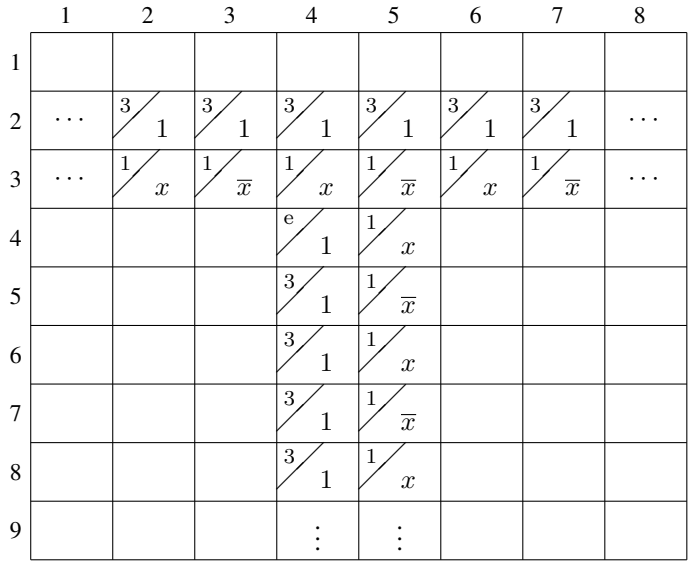


Fig. 3. Wires: downward T–connection.

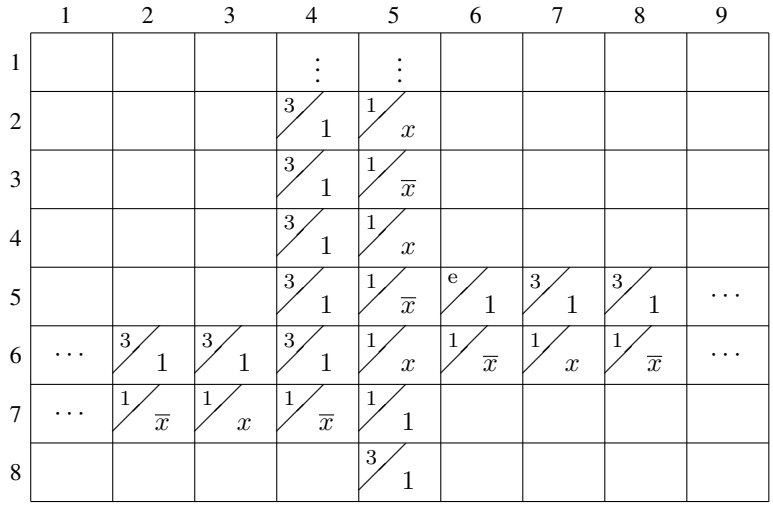


Fig. 4. Wires: upward T–connection.

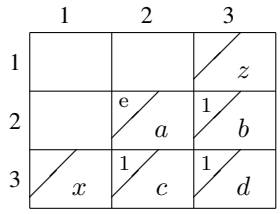
these cells to e) and setting cell $(3,4)$ to e;[1] (b) in Figure 3 setting to empty rows 2 and 3 in columns 6 through 8; (c) in Figure 4, setting to empty row 6 in columns 1 through 3, row 7, and row 8; (d) in Figure 4, setting to empty rows 5 and 6 in columns 6 through 9. To avoid cluttering the figures we did not specify input values that are adjacent to the main pattern and are obviously constrained to be 1 by adjacent cells with output values set to 3 (e.g., top row of empty cells in Figure 3).

Figure 5 can be used to implement a variety of logic gates computing Boolean operations on the input values at cells $(3,1)$ and $(1,3)$, when these values are regarded as assignments to Boolean variables denoted in the figure by x and z, respectively. The truth table in the bottom half of the figure specifies the constraints imposed on the input entries in other cells, as a function of the values assigned to x and z. The input b at cell $(2,3)$, for example, is constrained to be $x \vee \overline{z}$ while the input d at cell $(3,3)$ is constrained to be $x \oplus z$. The key to this table, which we leave to the reader to check, is that under the zero ISI distance requirement, the values of x and z fully determine the input values a, b, c, and d.[2]

Figures 6 and 7 illustrate how "wires" can be attached to the input and output terminals of the "universal gate" of Figure 5 to implement wired binary AND and XOR gates, respectively; in these figures, we have highlighted the position of the six labeled cells of Figure 5, and the full circles mark the terminals of the wired gates. (As will be explained in the next section,

[1]We set cell $(3,4)$ *explicitly* to e (as opposed to setting it to empty) for compatibility with Remark 2.1 below. In Section IV, a distinction is made between cells that are empty and cells that are explicitly set to e.

[2]For example, if $x = z = 1$, then setting a to be 1 would require setting b and c to be -1 (i.e., logical "0") in order satisfy zero ISI distance; this, in turn, would prevent zero ISI distance at $(3,3)$ for any choice of d. Therefore, we must have $a = -1$, and the remaining variables are then fully determined.



x	z	a	b	c	d
-1	-1	1	1	1	-1
-1	1	1	-1	1	1
1	-1	1	1	-1	1
1	1	-1	1	1	-1
		$\overline{x \wedge z}$	$x \vee \overline{z}$	$\overline{x} \vee z$	$x \oplus z$

Fig. 5. Universal gate and truth table.

the XOR gate construction will be used to implement the wire crossings in Figure 2.) Finally, Figures 8 and 9 depict INVERTERs that can be used to replace small portions of sufficiently long wires in the horizontal and vertical directions to impose negation between input values in the cells at the two ends of the wire. Observe that without INVERTERS (e.g., with just wires as in Figures 3 and 4) the negation–equality constraint feasible for any pair of input array entries would depend on the parity of the Manhattan distance between their coordinates in $\mathbb{Z}^2$: odd distance would force negation and even distance would force equality. And it might be impossible to lay out certain circuits if the parity of the Manhattan distance between different gate inputs and outputs were the only way to control Boolean negation of the associated variables.

Note that the input value in cell $(3, 4)$ in the horizontal INVERTER of Figure 8 is constrained to be 1 even though the output value in that cell is a 1 (and not the obvious case of a 3): this follows from the fact that cell $(3, 4)$ is part of a universal gate pattern and functions as cell $(3, 2)$ in Figure 5, given that cell $(1, 3)$ in the latter figure is forced to have input value $z = 1$. Under these circumstances, the value of c in Figure 5 is constrained to be 1 too. Similar reasoning applies to the input value in cell $(5, 4)$ in the vertical INVERTER of Figure 9.

Remark 2.1: It can be readily checked that for any empty cell in the figures of this section, if the adjacent cell to the right or below has its output value specified, then this value is either 3 or e. This, in turn, implies that the input values of the empty cells can always be set to 1 while satisfying the constraint that the summation in (5) evaluates to zero. Note that this property also holds for the bending wire constructions obtained from Figures 3 and 4, as described above. □

III. The full reduction

Having at hand the building blocks presented in Section II, we next combine them into a proof of the NP completeness of Problem 1.1.

Proof of Theorem 1.3 for Problem 1.1: Given any Boolean expression $F(\boldsymbol{v})$ (in conjunctive normal form satisfying the 3–SAT restriction), we first represent it using a logic circuit with wires and logic gates following the layout in Figure 2. Specifically, that layout consists of a set of horizontal wires that propagate the values of the circuit variables $\boldsymbol{v} = (v_1, v_2, \ldots, v_{n'})$ and a set of vertical wires that tap into these values and collect them into the desired sums (ORs) along the bottom, the outputs of which, in turn, are collected into a product (AND), one output at a time.

The next step is to map such a logic circuit into an $n \times n$ channel output array $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}_n}$ (for n to be determined that is polynomially large in the size of the Boolean expression) by implementing each portion of the circuit using the corresponding building blocks from Section II. We start with one particular column in Y and designate a set $V \subset \mathcal{Q}_n$ of equally spaced positions (minimum spacing to be clarified below) along this column for the n' circuit variables. The output values in the cells positioned at V will be set (explicitly) to e (thus, Remark 2.1 does not apply to these cells, as they are non-empty). Starting to the right of each circuit variable, the respective horizontal wire in Figure 2 will be mapped into Y through the pattern given by rows 2 and 3 in Figure 3.

A certain number of columns in Y will then be required to implement each OR clause. Specifically, for each such clause, there are three vertical wires in Figure 2 which tap into the horizontal wires corresponding to the circuit variables that participate in the clause. These vertical wires and the tapping to the horizontal wires are mapped into Y using the downward–T connection of Figure 3. Following the layout in Figure 2, each such vertical wire is continued downward until (possibly) reaching horizontal wires which it must cross. As illustrated in Figure 10, the crossing can be implemented by replacing portions of the horizontal and vertical wires in Figure 2 using a small circuit consisting of three XOR gates. These XOR gates, in turn, are mapped into Y through the pattern in Figure 7 (in this mapping, portions of the wires in the crossing circuit may require the use of the INVERTERs of Figures 8 and 9 to ensure that signals arriving at the inputs to the XOR gates do so without negation for any given placement of the XOR gates).

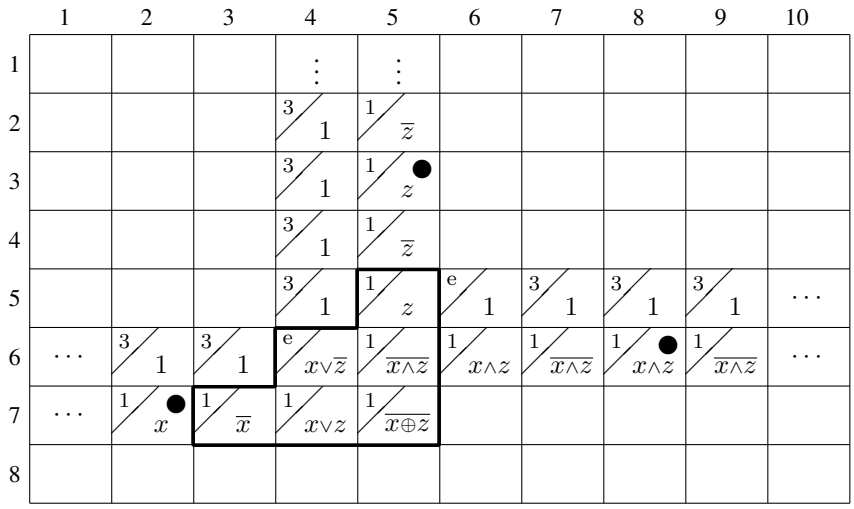


Fig. 6. AND gate with leads.

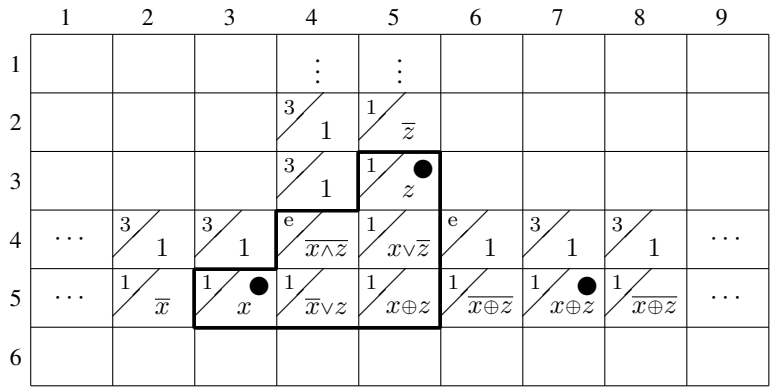


Fig. 7. XOR gate with leads.

In the mapping into Y, the vertical wires are continued until they extend sufficiently below all of the horizontal wires. At this point, and as shown in Figure 2, the three vertical wires that correspond to each clause are fed into an OR gate; this OR gate, in turn, can be implemented using two AND gates with appropriately negated inputs and outputs. The connection of the vertical wires to the OR gate is thus mapped into Y using AND gates as in Figure 6 and the INVERTERs of Figures 8 and 9.

The subcircuits for the OR clauses are positioned with uniform column spacing across the array Y. At the bottom of the array, the values of the OR clauses can be successively connected, as shown along the bottom of Figure 2, into a cumulative AND of each OR clause (again, INVERTERs can be used to ensure that the parity of each clause appears correctly in the cumulative output of each successive AND operation). We let (i_0, j_0) denote the coordinates of the termination point of the horizontal wire at the output of the last (rightmost) accumulating AND gate along the bottom of the array. All the entries of Y that have not been set by the mapping so far are assumed to be set to erasures e.

The spacing of the horizontal wires corresponding to the circuit variables and of the vertical tapping wires must be large enough to accommodate the wire-crossing circuit of Figure 10, any required INVERTERs, the OR subcircuits, and the cumulative AND operation along the bottom of the array. It is clear that there exists some constant spacing that will suffice, independently of the number of variables or clauses in F. Thus, the number of entries, n^2, in Y will be polynomially large in the number of variables and clauses in F, and Y can therefore be constructed in polynomial time.

The channel output array $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}_n}$ that is constructed from $F(\boldsymbol{v})$ is such that any input array $X = (x_{i,j})_{(i,j)\in\mathcal{Q}_n}$ over Φ for which the ISI filter (2) produces output that is at zero distance from Y must satisfy $x_{i_0,j_0} = F\big((X)_V\big)$. Conversely, any assignment of values from Φ to the $n' = |V|$ entries that are indexed by V can be extended into an array $X = (x_{i,j})_{(i,j)\in\mathcal{Q}_n}$ over Φ where $x_{i_0,j_0} = F\big((X)_V\big)$, and the filter (2) produces for the input X an array that is at zero distance from Y. The reduction is completed by setting $y_{i_0+1,j_0} = 3$ which, in turn, constrains x_{i_0,j_0} to be 1 to maintain the zero distance. Thus, $F(\boldsymbol{v})$ is satisfiable (i.e., there is an assignment to $\boldsymbol{v}$ for which $F(\boldsymbol{v})$ evaluates to 1) if and only if there exists an input array X that induces ISI at zero

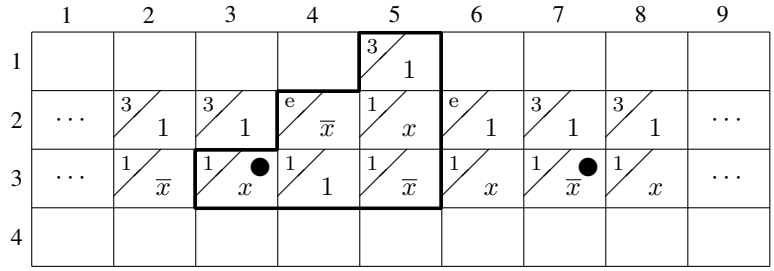


Fig. 8. Horizontal INVERTER.

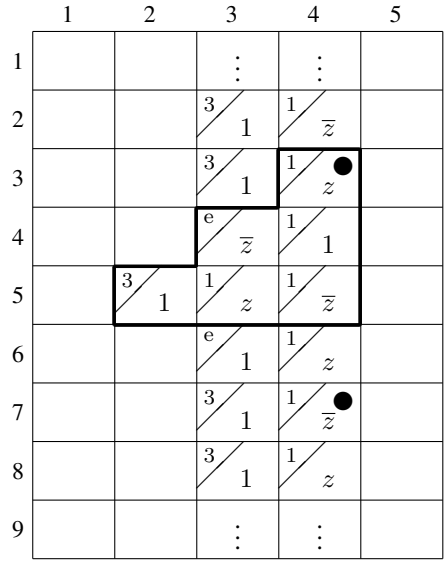


Fig. 9. Vertical INVERTER.

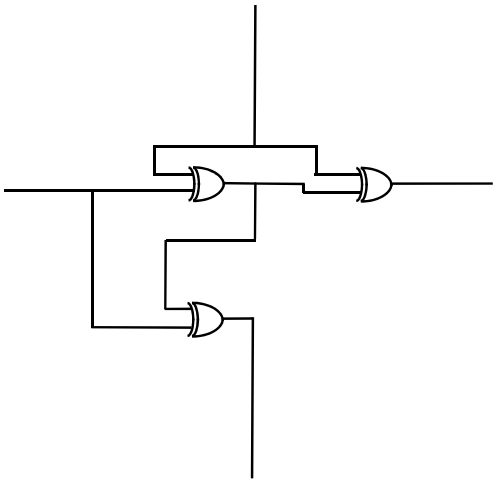


Fig. 10. Crossing wires using XOR gates.

distance from the constructed channel output array Y. □

IV. Gaussian noise

In this section, we prove Theorem 1.3 for Problem 1.2, the Gaussian (squared error distance) case. Our approach will be to reduce a constrained, but still NP complete, version of Problem 1.1 to Problem 1.2. The reduction will involve replacing erasures with Gaussian channel output values that behave partially like erasures, namely, the output values -2, 0, or 2: each of these values is at a squared error distance of 1 from two elements in the range of ISI output values $\mathcal{U} = \{-3, -1, 1, 3\}$. For $y = -2, 0, 2$, we denote the subset which consists of these two closest elements in $\mathcal{U}$ by $\mathcal{U}(y)$; thus, $\mathcal{U}(0) = \{-1, 1\}$ and $\mathcal{U}(2) = \{1, 3\}$ (in our forthcoming reduction, we will not actually use -2 to replace erasures).

Consider Problem 1.1 with instances restricted to those that can arise in the reduction from 3–SAT to this problem in the proof of Section III. Clearly, Problem 1.1 under this restricted set of instances is still NP complete. Given such an instance, namely, an $n \times n$ channel output array $Y = (y_{i,j})$ over $\{1, 3, \mathrm{e}\}$, we can attempt to replace erasures in Y by either 0 or 2, thereby forming an array $Y' = (y'_{i,j})$ over $\{0, 1, 2, 3\}$ with the following property: for any input array X achieving (5), we can modify entries in X to generate an array $X' = (x'_{i,j})$ over Φ which will still satisfy (5) (with respect to the original array Y) and, in addition, for every $(i, j) \in \mathcal{Q}_n$,

$$y'_{i,j} \in \{0, 2\} \quad \Longrightarrow \quad x'_{i,j} + x'_{i-1,j} + x'_{i,j-1} \in \mathcal{U}(y'_{i,j}) \; .$$

Thus, the squared error distance between Y' and the ISI resulting from X' will be precisely the number of erasures in Y. For example, one such attempt could be replacing every erasure in Y by a 2 and then modifying the achieving input array X so that every entry that corresponds to an empty cell in the figures of Section III becomes a 1 (see Remark 2.1).

It turns out that two types of erasures in Y pose problems for this approach: those that correspond to the assumed erasure in cell $(4, 6)$ of Figure 3 and those that correspond to cell $(7, 6)$ of Figure 6. Now, for these cases, we could try to replace the erasures in Y by 0's (instead of 2's). While it is possible to adjust locally the entries of X to ensure that the ISI output values at the positions of these erasures are in $\mathcal{U}(0) = \{-1, 1\}$, such an adjustment may need to propagate further in X, possibly leading to conflicts with other entries of X and thus to a violation of (5). We work around this difficulty by replacing the reduction components of Figure 3 and Figure 6 with the somewhat more complicated alternatives in Figures 11 and 12. We leave it to the reader to check that the specified channel outputs for these modified components do indeed impose the displayed constraints on the input values, and that Remark 2.1 applies to the new figures as well (including the bending wires obtained from Figure 11 similarly as above). It should also be evident that the modified components can be used to implement the same functionality in the reduction of 3–SAT to Problem 1.1 as their original counterparts. The modified components address the above erasure replacement issue, as explained in the proof below.

Proof of Theorem 1.3 for Problem 1.2: We first note that Problem 1.1 remains NP complete when instances of the problem are restricted to channel output arrays Y obtained from instances of 3–SAT via the reduction outlined in the proof of the first part of Theorem 1.3 in Section III, but with the modified components of Figure 11 (downward-T connection) and Figure 12 (AND gate) used in place of their counterparts in Figures 3 and 6. The following reduction of such a restricted form of Problem 1.1 to Problem 1.2 will complete the proof. Let Y be an instance of the restricted Problem 1.1, and let $\mathcal{J} = \mathcal{J}(Y)$ (respectively, $\mathcal{K} = \mathcal{K}(Y)$) be the set of positions of all the erasures in Y that correspond to cell $(7, 6)$ (respectively, $(7, 7)$) of Figure 12. Let $Y' = (y'_{i,j})$ be obtained from Y by

$$y'_{i,j} = \begin{cases} y_{i,j} & \text{if } y_{i,j} \in \{1, 3\} \\ 0 & \text{if } (i, j) \in \mathcal{J} \cup \mathcal{K} \\ 2 & \text{otherwise} \end{cases} \quad , \quad (i, j) \in \mathcal{Q}_n \; .$$

Also, let $N = N(Y)$ be the total number of erasures in Y (i.e. N is the number of 0's and 2's in Y'). We claim that (5) holds for the instance $Y = (y_{i,j})$ if and only if (6) holds for the instance $(Y', T{=}N)$, thereby establishing the reduction to Problem 1.2. The rest of the proof verifies this claim.

Suppose that Y satisfies (5) and let $X = (x_{i,j})$ be an input array achieving (5) for this Y. Denote by $\mathcal{P} = \mathcal{P}(Y) \subseteq \mathcal{Q}_n$ the set of positions of all the entries of Y that correspond to the *non-empty* cells in Figure 4, Figures 7–9, Figures 11–12, or the bending wires obtained from Figures 4 and 11. Note that $\mathcal{P}$ contains the set $\mathcal{J} \cup \mathcal{K}$, since the respective cells are now explicitly assigned the output value e; for the same reason, $\mathcal{P}$ also contains the set V of positions of the n' circuit variables. Let $X' = (x'_{i,j})$ be obtained from X through the following four steps:

(a) For all positions $(i, j) \in \mathcal{P} \setminus (\mathcal{J} \cup \mathcal{K})$, let $x'_{i,j} = x_{i,j}$.
(b) For all positions $(i, j) \in \mathcal{J}$, let $x'_{i,j}$ be such that $x'_{i,j} + x_{i-1,j} + x_{i,j-1} \in \{-1, 1\}$ $(= \mathcal{U}(0))$.
(c) For all positions $(i, j) \in \mathcal{K}$, let $x'_{i,j}$ be such that $x'_{i,j} + x_{i-1,j} + x'_{i,j-1} \in \{-1, 1\}$.
(d) For all positions $(i, j) \in \mathcal{Q}_n \setminus \mathcal{P}$, let $x'_{i,j} = 1$.

By Remark 2.1 it follows that X' achieves (5) for the instance Y.

We now argue that X' also achieves (6) for the instance $(Y', T{=}N)$. First, we argue that for every erasure position (i, j) in Y, the respective ISI output value induced by X', namely, $x'_{i,j} + x'_{i-1,j} + x'_{i,j-1}$, belongs to $\mathcal{U}(y'_{i,j})$. For the erasures that are indexed by $(i, j) \in \mathcal{J} \cup \mathcal{K}$ (in which case $y'_{i,j} = 0$) this follows immediately from steps (b) and (c). We leave it to the reader to verify that steps (a) and (d) guarantee that for all other erasure positions (i, j) (namely, when $y'_{i,j} = 2$), at least two of the three values—$x'_{i,j}$, $x'_{i-1,j}$, and $x'_{i,j-1}$—equal 1. Thus, the squared error distance between Y' and the ISI resulting from X' is exactly $T = N$, with an increment of 1 incurred for each channel output equaling 0 or 2, and an increment of 0 incurred everywhere else (the latter, again, by virtue of X achieving (5)), thereby implying (6). This verifies one direction of the above claim concerning the reduction.

Conversely, if (6) holds for $(Y', T{=}N)$, it must hold with equality, since a distance increment of 1 must be incurred for

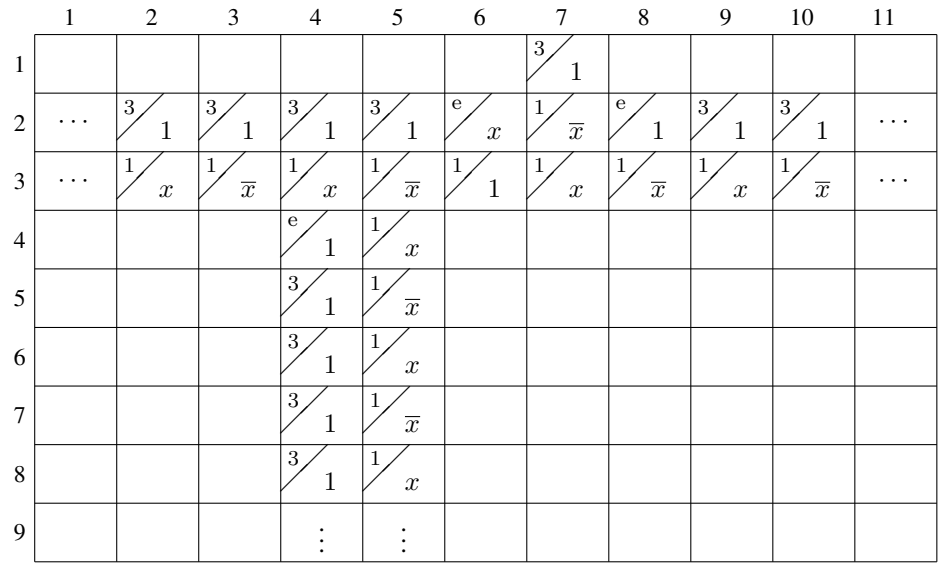


Fig. 11. Modified downward T–connection.

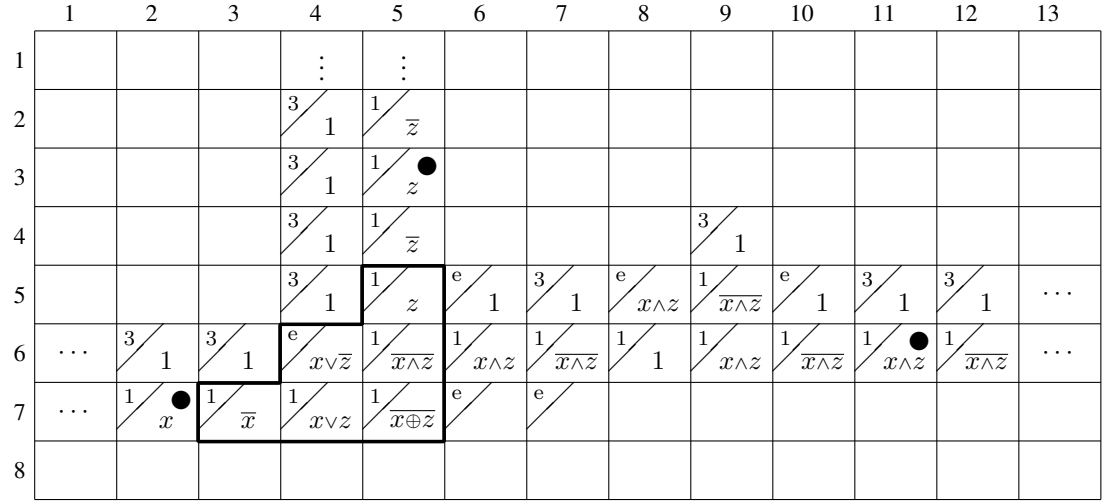


Fig. 12. Modified AND gate with leads.

each channel output equaling 0 or 2, of which there are $T = N$. This implies that the ISI output values for a minimizing input array X' achieving (6) must be at zero distance from all the respective entries of Y' differing from 0 and 2, which are precisely the non-erased entries in Y. The input array X' thus also achieves (5). □

V. Multiuser detection and 1D MLSD

In [11] the (binary input) synchronous multiuser detection problem is reduced to the following problem.

Problem 5.1: Given a $k \times k$ nonnegative-definite matrix H and a column vector $\boldsymbol{z} \in \mathbb{R}^k$, determine

$$\arg\max_{\boldsymbol{b}\in\Phi^k} \left(2\boldsymbol{b}^t\boldsymbol{z} - \boldsymbol{b}^t H\boldsymbol{b}\right) \ .$$

It is shown in [11] that this problem is NP hard when a problem instance consists of $\boldsymbol{z}$ and H over the rational field $\mathbb{Q}$. In addition, the following single-user intersymbol interference problem was considered in [11]:

Problem 5.2: Determine the maximum *a posteriori* (MAP) estimate of the sequence $(b(i))_{i=-M}^{M} \in \Phi^{2M+1}$ upon observing the continuous real time signal

$$r(t) = \sum_{i=-M}^{M} b(i)s(t - i\Delta) + \nu(t)$$

for $t \in (-\infty, \infty)$, with $s(t)$ having finite support, $\nu(t)$ being white Gaussian noise, and $(b(i))_i$ being i.i.d. uniform Bernoulli over Φ.

Problem 5.2 was shown in [11] to coincide with Problem 5.1 with the additional restriction that H is Toeplitz. Conjecture 1 on p. 311 in [11] states that Problem 5.1 remains NP hard even when H is restricted to be Toeplitz, implying the intractability of Problem 5.2 with a growing number, $2M$, of signals that

interfere at a given time (the case of a bounded number of interfering signals is readily handled by the Viterbi algorithm). We prove this conjecture by showing a reduction from a slightly constrained version of Problem 1.2, which remains NP complete, to a decision version of Problem 5.1 in which H is a nonnegative-definite matrix with Toeplitz structure.

We begin with some definitions and claims. Given an $n \times n$ real array $Y = (y_{i,j})_{(i,j)\in\mathcal{Q}_n}$, let $\mathsf{v}(Y)$ denote the column vector in $\mathbb{R}^{n^2}$ whose mth entry is $y_{i(m),j(m)}$, where $i(m) = 1 + \lfloor (m-1)/n \rfloor$ and $j(m) = m - n\lfloor (m-1)/n \rfloor$. The vector $\mathsf{v}(Y)$ thus consists of a concatenation of the transposed rows of Y. Let A_n be an $(n^2+n) \times n^2$ lower-triangular Toeplitz matrix defined as follows:

$$(A_n)_{\ell,m} = \begin{cases} 1 & \text{if } \ell - m \in \{0,1,n\} \\ 0 & \text{otherwise} \end{cases} ,$$
$$1 \le \ell \le n^2+n \ , \quad 1 \le m \le n^2 \ .$$

Thus, A_n takes the form:

$$A_n = \begin{pmatrix} 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 \\ 1 & 1 & 0 & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & 1 & 1 & & & & & 0 \\ 0 & 0 & 1 & & & & \cdot & \\ \vdots & 0 & 0 & \cdot & & & \cdot & \cdot \\ 0 & \vdots & 0 & \cdot & & & \cdot & \cdot \\ 1 & 0 & \vdots & & \cdot & & & \cdot \\ 0 & 1 & 0 & & \cdot & & 0 & \\ 0 & 0 & 1 & & & \cdot & 0 & 0 \\ & 0 & 0 & & & \cdot & 0 & 0 \\ \cdot & & 0 & \cdot & & & 1 & 0 \\ \cdot & \cdot & & & & & 1 & 1 \\ \cdot & \cdot & \cdot & & \cdot & & 0 & 1 \\ & \cdot & \cdot & & & & 0 & 0 \\ 0 & & \cdot & & & \cdot & \vdots & 0 \\ 0 & 0 & & & & & 0 & \vdots \\ 0 & 0 & 0 & & & & 1 & 0 \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & 0 & 1 \end{pmatrix} \begin{matrix} \leftarrow 1 \\ \leftarrow 2 \\ \\ \\ \\ \\ \leftarrow n{+}1 \\ \\ \\ \\ \leftarrow n^2 \\ \\ \\ \\ \\ \\ \leftarrow n^2{+}n \end{matrix}$$

Given an input array $X = (x_{i,j})_{(i,j)\in\mathcal{Q}_n}$ over Φ, let $U = (u_{i,j})$ be the array at the output of the ISI filter (2), namely, $u_{i,j} = x_{i,j} + x_{i-1,j} + x_{i,j-1}$ for $(i,j) \in \mathcal{Q}_n$ where, as before, $x_{i,0} = x_{0,j} = 1$. It can be checked that the corresponding entries of $\mathsf{v}(U)$ and $A_n\mathsf{v}(X)$ are equal for the set of indices

$$\mathcal{M} = \{m = (i-1)n + j \ : \ 1 < i,j \le n\} \ .$$

For the remaining indices, which correspond to borders with either $i = 1$ or $j = 1$ or the extra n indices of $A_n\mathsf{v}(X)$, the entries of $A_n\mathsf{v}(X)$ are sums of boundary entries of X (i.e., entries with indices (i,j) where either $i \in \{1,n\}$ or $j \in \{1,n\}$).

We then have the following lemma.

Lemma 5.3: Let $Y = (y_{i,j})$ be an $n \times n$ array with entries in $\{0,1,2,3\}$ such that $y_{i,j} = 3$ if either $i \in \{1,n\}$ or $j \in \{1,n\}$. Let T be the number of 0's and 2's in Y. Let $\boldsymbol{y}$ be the column vector in $\mathbb{R}^{n^2+n}$ whose ℓth entry equals the ℓth entry of $\mathsf{v}(Y)$ for $\ell \in \mathcal{M}$ and equals the ℓth entry of $A_n\mathbf{1}$ for $\ell \notin \mathcal{M}$, where $\mathbf{1}$ is a vector of all 1's. Then

$$\min_X \sum_{1\le i,j\le n} \big(y_{i,j} - (x_{i,j} + x_{i-1,j} + x_{i,j-1})\big)^2 \le T \qquad (7)$$

(where the minimization is over all arrays $X = (x_{i,j})$ over Φ and $x_{i,0} = x_{0,j} = 1$) if and only if

$$\min_{\boldsymbol{x}\in\Phi^{n^2}} \|\boldsymbol{y} - A_n\boldsymbol{x}\|_2^2 \le T \ , \qquad (8)$$

where $\|\cdot\|_2$ denotes the L_2 norm on vectors.

Proof: Suppose that (7) holds and let $X^* = (x^*_{i,j})$ be the array that achieves the minimum. Then, necessarily, for every $(i,j) \in \mathcal{Q}_n$,

$$|y_{i,j} - (x^*_{i,j} + x^*_{i-1,j} + x^*_{i,j-1})| = \begin{cases} 1 & \text{if } y_{i,j} \in \{0,2\} \\ 0 & \text{if } y_{i,j} \in \{1,3\} \end{cases} ,$$

namely, the inequality in (7) must hold with equality. It then follows from the choice of boundary values of Y that all boundary values of X^* (those with indices $i \in \{1,n\}$ or $j \in \{1,n\}$) must be 1. Above, it was noted that $A_n\mathsf{v}(X^*)$ coincides with $\mathsf{v}(U^*)$ in all indices in $\mathcal{M}$, where U^* is the 2D ISI signal induced by X^*. It then follows from the definition of $\boldsymbol{y}$, and the fact that all 0's and 2's in Y get mapped to locations in $\boldsymbol{y}$ with indices in $\mathcal{M}$, that the squared distance between $\boldsymbol{y}$ and $A_n\mathsf{v}(X^*)$ incurred on $\mathcal{M}$ is T. It remains to show that the squared distance incurred on the complement set $\{1,2,\ldots,n^2{+}n\} \setminus \mathcal{M}$ is 0. Recall that for these indices, the entries of $\boldsymbol{y}$ are set to the corresponding entries of $A_n\mathbf{1}$. The claim follows from the fact (noted above) that the corresponding entries of $A_n\mathsf{v}(X^*)$ are sums of boundary values of X^*, which, in turn, are constrained to be all 1's.

The converse follows similarly. In this case, suppose (8) holds and that $\boldsymbol{x}^*$ achieves the minimum. Let U^* be the 2D ISI induced by the the $n \times n$ array $\mathsf{v}^{-1}(\boldsymbol{x}^*)$ (where v^{-1} is the inverse of v). Again, because T equals the number of 0's and 2's in Y, and because these appear in $\boldsymbol{y}$ in indices belonging to $\mathcal{M}$ where entries of $A_n\boldsymbol{x}^*$ equal respective entries of $\mathsf{v}(U^*)$, equality is achieved in (8). This implies that any entry in $\boldsymbol{y}$ which is either 1 or 3 or has an index belonging to $\{1,2,\ldots,n^2{+}n\} \setminus \mathcal{M}$, must be equal to the respective entry in $A_n\boldsymbol{x}^*$. Thus, as before, confined to the inverses of indices in $\mathcal{M}$ under the mapping v, the ISI output array U^* induced by $\mathsf{v}^{-1}(\boldsymbol{x}^*)$ incurs a distance T with respect to Y. For the remaining indices (corresponding to those with $i = 1$ or $j = 1$), the relevant boundary entries of $\mathsf{v}^{-1}(\boldsymbol{x}^*)$ are forced to be 1 by the boundary of 3's (for $j = 1$) and by the definition of $\boldsymbol{y}$ (for $i = 1$). These 1's then sum with the assumed 1's for $i = 0$ or $j = 0$ to form ISI output values of 3 which incur zero distance from the corresponding boundary 3's in Y. □

We are now ready to prove the NP completeness of the following problem, which, as shown in [11], is the decision version of Problem 5.2 (and is also the decision version of Problem 5.1 with the added Toeplitz restriction).

Problem 5.4: Given a $k \times k$ nonnegative-definite Toeplitz matrix H over $\mathbb{Q}$ and a column vector $\boldsymbol{z} \in \mathbb{Q}^k$, along with a scalar $T \in \mathbb{Q}$, does

$$\max_{\boldsymbol{b}\in\Phi^k} \left(2\boldsymbol{b}^t\boldsymbol{z} - \boldsymbol{b}^t H \boldsymbol{b}\right) \ge T$$

hold?

Theorem 5.5: Problem 5.4 is NP complete.

Proof: The reduction from Problem 1.1 to Problem 1.2, as presented in Section IV, involves only instances of the latter problem that satisfy the following two restrictions:

(i) The channel output array Y is over (the restricted alphabet) $\{0,1,2,3\}$.

(ii) The value of T is precisely the number of 0's and 2's in Y.

Additionally, the initial reduction from 3–SAT to Problem 1.1 can readily incorporate the following restriction:

(iii) The boundary entries of Y are equal to 3.

(In the reduction described in the proof in Section III, simply map the logic circuit sufficiently away from the boundaries of Y.) Since our reduction from Problem 1.1 to Problem 1.2 preserves restriction (iii), it follows that Problem 1.2 with the added restrictions (i)–(iii) remains NP complete. Lemma 5.3 shows that this restricted problem, in turn, can be reduced to the following problem:

Given a column vector $\boldsymbol{y} \in \mathbb{Q}^{n^2+n}$ and a scalar $T \in \mathbb{Q}$, does

$$\min_{\boldsymbol{x} \in \Phi^{n^2}} \|\boldsymbol{y} - A_n \boldsymbol{x}\|_2^2 \leq T \tag{9}$$

hold?

The reduction to Problem 5.4 is completed by noting that

$$\|\boldsymbol{y} - A_n \boldsymbol{x}\|_2^2 = \boldsymbol{y}^t \boldsymbol{y} - 2\boldsymbol{x}^t A_n^t \boldsymbol{y} + \boldsymbol{x}^t A_n^t A_n \boldsymbol{x} \ ,$$

so that (9) is equivalent to

$$\max_{\boldsymbol{b} \in \Phi^{n^2}} \left(2b^t \boldsymbol{z} - \boldsymbol{b}^t H \boldsymbol{b} \right) \leq T' \ ,$$

where

$$H = A_n^t A_n \ , \quad \boldsymbol{z} = A_n^t \boldsymbol{y} \ , \quad \text{and} \quad T' = \boldsymbol{y}^t \boldsymbol{y} - T.$$

It can be checked that $A_n^t A_n$ is Toeplitz and nonnegative-definite. This completes the (polynomial-time) reduction from the version of Problem 1.2 with the added restrictions (i)–(iii) (which is still NP complete) to Problem 5.4. □

ACKNOWLEDGMENTS

We thank Sergio Verdú for helpful discussions about [11].

REFERENCES

[1] P. Kumar and S. Roy, "Two-dimensional equalization: Theory and applications to high density magnetic recording," *IEEE Trans. Commun.,* vol. 42, no. 2/3/4 (Feb.–Apr. 1994), 386–395.

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

[3] M.A. Neifeld, K. M. Chugg, and B. M. King, "Parallel data detection in page oriented optical memory," *Optical Letters,* vol. 21 (1996), 1481–1483.

[4] X. Chen, K. M. Chugg, and M. A. Neifeld, "Near-optimal parallel distributed data detection for page-oriented optical memories," *IEEE J. Select. Topics Quantum Electron.,* vol. 4, no. 4 (Sep.–Oct. 1998), 866–879.

[5] N. Singla, J. A. O'Sullivan, R. S. Indeck, and Y. Wu, "Iterative decoding and equalization for 2-D recording channels," *IEEE Trans. Magn.,* vol. 38, no. 5 (Sep. 2002), 2328–2330.

[6] M. Marrow and J. K. Wolf, "Iterative detection of 2-dimensional ISI channels," *Proc. IEEE Inform. Theory Workshop (ITW),* Paris, France (Mar. 2003), 131–134.

[7] Y. Wu, J. A. O'Sullivan, N. Singla, and R. S. Indeck, "Iterative detection and decoding for separable two-dimensional intersymbol interference," *IEEE Trans. Magn.,* vol. 39, no. 4 (July 2003), 2115–2120.

[8] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory,* vol. 18, no. 3 (May 1972), 363–378.

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

[10] S. E. Shimonyi, "Finding MAPs for belief networks is NP hard," *Artificial Intelligence*, 68 (1994), 399–410.

[11] S. Verdú, "Computational complexity of optimum multiuser detection," *Algorithmica*, 4 (1989), 303–312.

Erik Ordentlich (S'92–M'96–SM'06–F'11) received the S.B. and S.M. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, in 1990, and the Ph.D. degree, also in electrical engineering, from Stanford University, Stanford, CA, in 1996.

He is a Senior Research Scientist in the Information Theory Research Group at Hewlett-Packard Laboratories, Palo Alto, CA. He has been with Hewlett-Packard Laboratories since 1996, with the exception of a period in 1999-2002, when he was with iCompression, Inc., Santa Clara, CA. His work has addressed multiple topics in signal processing and information theory. He is a coinventor on numerous U.S. Patents and contributed technology to the ISO JPEG 2000 image compression standard.

Dr. Ordentlich was a corecipient of the 2006 IEEE Joint Communications/ Information Theory Paper Award and is a member of Phi Beta Kappa and Tau Beta Pi. He served as Associate Editor for Source Coding for the IEEE TRANSACTIONS ON INFORMATION THEORY from 2007-2010.

Ron M. Roth (M'88–SM'97–F'03) was born in Ramat Gan, Israel, in 1958. He received the B.Sc. degree in computer engineering, the M.Sc. in electrical engineering and the D.Sc. in computer science from Technion—Israel Institute of Technology, Haifa, Israel, in 1980, 1984 and 1988, respectively. Since 1988 he has been with the Computer Science Department at Technion, where he now holds the General Yaakov Dori Chair in Engineering. During the academic years 1989–91 he was a Visiting Scientist at IBM Research Division, Almaden Research Center, San Jose, California, and during 1996–97 and 2004–05 he was on sabbatical leave at Hewlett–Packard Laboratories, Palo Alto, California. He is the author of the book *Introduction to Coding Theory*, published by Cambridge University Press in 2006. Dr. Roth was an associate editor for coding theory in IEEE TRANSACTIONS ON INFORMATION THEORY from 1998 till 2001, and he is now serving as an associate editor in *SIAM Journal on Discrete Mathematics*. His research interests include coding theory, information theory, and their application to the theory of complexity.