

# The Binary Linearization Complexity of Pseudo-Boolean Functions

Matthias Walter\*

July 11, 2023

## Abstract

We consider the problem of linearizing a pseudo-Boolean function  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  by means of  $k$  Boolean functions. Such a linearization yields an integer linear programming formulation with only  $k$  auxiliary variables. This motivates the definition of the linearization complexity of  $f$  as the minimum such  $k$ . Our theoretical contributions are the proof that random polynomials almost surely have a high linearization complexity and characterizations of its value in case we do or do not restrict the set of admissible Boolean functions. The practical relevance is shown by devising and evaluating integer linear programming models of two such linearizations for the low auto-correlation binary sequences problem. Still, many problems around this new concept remain open.

## 1 Introduction

We consider *pseudo-Boolean functions*, i.e.,  $f : \{0, 1\}^n \rightarrow \mathbb{R}$ . Such functions arise at the intersection of *Constraint Programming* and *Operations Research*. The corresponding *pseudo-Boolean optimization problem* of minimizing  $f$  over  $\{0, 1\}^n$  is known to be NP-hard since it subsumes the maximum cut problem [8, 26]. Besides being subject to minimization, pseudo-Boolean functions appear frequently in problem constraints, e.g., in satisfiability problems. It is known well that every pseudo-Boolean function has a unique representation by means of a multilinear polynomial. Above-mentioned optimization problems can be reduced to integer *linear* programming problems by introducing auxiliary variables, e.g., for every monomial of its multilinear representation. Since there may be a huge number of monomials, the natural question for alternative ways of linearizing using fewer auxiliary variables arises. The main research question addressed by this paper is that for the minimum number of such auxiliary variables under the additional restriction that these are binary variables.

To this end, a *linearization* of  $f$  is defined by functions  $g_1, g_2, \dots, g_k : \{0, 1\}^n \rightarrow \mathbb{R}$  and parameters  $\beta \in \mathbb{R}$  and  $b \in \mathbb{R}^n$  such that

$$f(x) = a^\top x + \beta + \sum_{i=1}^k b_i g_i(x) \quad (1)$$

holds for all  $x \in \{0, 1\}^n$ . Its *size* is the number  $k$  of such functions. The *linearization complexity* of  $f$  with respect to a family  $\mathcal{G}$  of functions is defined as the minimum size of a linearization with  $g_i \in \mathcal{G}$  for all  $i$ , and is denoted by  $lc_{\mathcal{G}}(f)$ .

In this paper we focus on *binary* linearizations, which are those where each function  $g \in \mathcal{G}$  is Boolean, i.e.,  $g : \{0, 1\}^n \rightarrow \{0, 1\}$  holds. For a binary linearization of size  $k$  there exists an integer linear programming (IP) formulation with  $n + k$  variables which works as follows. In addition to the variables  $x \in \{0, 1\}^n$  we introduce the binary variables  $y \in \{0, 1\}^k$  that shall represent the values  $g_i(x)$ . By construction,  $f(x)$  is affine in  $(x, y)$ . The following constraints ensure that  $y_i = g_i(x)$  holds:

$$\sum_{j:\bar{x}_j=0} x_j + \sum_{j:\bar{x}_j=1} (1 - x_j) + y_i \geq 1 \quad \forall \bar{x} \in \{0, 1\}^n : g_i(\bar{x}) = 1, \quad (2a)$$

$$\sum_{j:\bar{x}_j=0} x_j + \sum_{j:\bar{x}_j=1} (1 - x_j) + (1 - y_i) \geq 1 \quad \forall \bar{x} \in \{0, 1\}^n : g_i(\bar{x}) = 0. \quad (2b)$$

---

\*Department of Applied Mathematics, University of Twente, The Netherlands. E-mail: m.walter@utwente.nl

Note that (2) consists of  $2^n$  inequalities. The associated separation problem can be solved with linearly many evaluations of  $g_i$  plus linear computation time by computing, for given  $\hat{x} \in [0, 1]^n$  the closest vertex  $\bar{x}^0 \in \{0, 1\}^n$  of the cube and all vectors  $\bar{x}^j \in \{0, 1\}^n$  that can be obtained from  $\bar{x}^0$  by flipping coordinate  $j$ , computing  $g_i(\bar{x}^j)$  and testing (2) for  $j = 0, 1, 2, \dots, n$ . However, it is worth noting that the LP bounds obtained from such a formulation are typically very weak. A special case in which (2) yields a perfect formulation is that of a parity indicator variable, that is, if  $g_1(x) = 1$  holds if and only if  $\sum_{i=1}^n x_i$  is odd (resp. even) [25]. Here, perfect means that the convex hull of  $\{(x, g_1(x)) : x \in \{0, 1\}^n\}$  is described by (2) and  $0 \leq x_i \leq 1$  for  $i = 1, 2, \dots, n$ . Such variables have applications in integer programming approaches for decoding of binary codes [33]. For now, the main purpose of (2) is to guarantee the *existence* of integer programming formulations. Once a more specific family of functions is identified to yield small linearizations for a certain application, improved integer programming relaxations can be developed for this family. We will later see an example for this proposed approach. We denote by  $\mathcal{B}$  the set of all Boolean functions. An interesting constrained subclass is the family  $\mathcal{C} \subseteq \mathcal{B}$  of functions of the form

$$g_{I,J}(x) := \prod_{i \in I} x_i \cdot \prod_{j \in J} (1 - x_j), \quad (3)$$

i.e., products of potentially complemented variables. Even more restricted is the family  $\mathcal{M} \subseteq \mathcal{C}$  of *monomials*, i.e., functions  $g_{I,\emptyset}$  for all  $I$ .

There often exist smaller and stronger formulations than (2), e.g., if  $g_i(x)$  does not depend on all  $x$ -variables. For instance, for each function  $g_{I,J} \in \mathcal{C}$ , a perfect IP formulation is known, which is due to Fortet [18, 19], namely

$$y_{I,J} \leq x_i \quad \forall i \in I, \quad (4a)$$

$$y_{I,J} \leq 1 - x_j \quad \forall j \in J, \quad (4b)$$

$$1 - y_{I,J} \leq \sum_{i \in I} (1 - x_i) + \sum_{j \in J} x_j, \quad (4c)$$

$$y_{I,J} \geq 0. \quad (4d)$$

On the one hand, the interplay of multiple simultaneous linearizations has been investigated by many researchers. This includes mainly results about the quadratic case, e.g., for products of binary and continuous variables or products of binary variables and linear combinations of binary variables [21]. Worth mentioning is also the relaxation-linearization technique due to Adams and Sherali [1, 2, 3], which was also applied to polynomial optimization [32]. The strength of such alternative formulations for quadratic polynomials is compared theoretically and in practice in [20]. However, we are not aware of any research about their size.

On the other hand, the minimum number of additional *quadratzation variables* was investigated by Anthony et al. [4]. These are variables  $y \in \{0, 1\}^k$  such that

$$f(x) = \min\{g(x, y) : \exists y \in \{0, 1\}^k\} \quad (5)$$

holds for a quadratic polynomial  $g : \{0, 1\}^{n+k} \rightarrow \mathbb{R}$ . This approach is not directly related to the linearization complexity, but similar properties of pseudo-Boolean functions are exploited. Besides establishing first results on the linearization complexity with respect to different function families we showcase the use of a new alternative linearization technique for an application. Most importantly, we present open problems and interesting research questions to stimulate further research in this direction. Before presenting the outline of the paper, the potential of using various linearizations is exemplified.

**Example.** Consider  $f(x) = x_1x_2 + x_1x_3 + x_2x_3 - x_1x_2x_3$ . Since it is the sum of four binary (non-affine) terms it has  $lc_{\mathcal{C}}(f) \leq lc_{\mathcal{M}} = 4$ . However, it turns out that  $lc_{\mathcal{C}}(f) \leq 1$  holds: consider  $g_1(x) = (1 - x_1)(1 - x_2)(1 - x_3)$  and observe that  $f(x) = x_1 + x_2 + x_3 - 1 + g_1(x)$ , which yields  $lc_{\mathcal{C}}(f) \leq 1$ . It is also not hard to see that  $lc_{\mathcal{C}}(f) \geq 1$  holds (see Proposition 2), which establishes  $lc_{\mathcal{C}}(f) = 1$ .

Clearly, this example can be extended to an arbitrary degree, showing that the  $lc_{\mathcal{M}}(f)$  can be exponentially larger (in  $n$ ) than  $lc_{\mathcal{C}}(f)$ .

**Outline.** The paper is structured as follows. In Section 2 we present theoretical results about the linearization complexity for various families  $\mathcal{G}$  of Boolean functions. Section 3 is about the low auto-correlation binary sequences problem, which is an optimization problem that arises in theoretical physics, and for which we present a new linearization and evaluate it computationally. We conclude the paper with several open problems in Section 4 and hope that some of them will be addressed by researchers in the future.

## 2 Basic results

**Nonlinear part of a function.** We first introduce an auxiliary function with the purpose of removing the linear part of a given function. For a function  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  we denote by  $\tilde{f}$  the function defined by

$$\tilde{f}(x) := f(x) - f(\mathbb{0}) - \sum_{i=1}^n (f(\mathbf{e}^i) - f(\mathbb{0}))x_i,$$

which we call *the nonlinear part of  $f$* . Here,  $\mathbb{0}$  and  $\mathbf{e}^i$  denote the zero vector and the  $i$ -th standard unit vector, respectively. The following proposition makes clear why the nonlinear portion is useful.

**Proposition 1.** *Let  $f : \{0, 1\}^n \rightarrow \mathbb{R}$ . Then its nonlinear part satisfies the following properties.*

1.  $\tilde{f}(\mathbb{0}) = 0$  and  $\tilde{f}(\mathbf{e}^i) = 0$  holds for  $i = 1, 2, \dots, n$ .
2.  $lc_{\mathcal{G}}(\tilde{f}) = lc_{\mathcal{G}}(f)$  holds for any family  $\mathcal{G}$  of functions.

*Proof.* To verify the first property it suffices to plug in the zero vector and the unit vectors. Now observe that  $\tilde{f}(x) = a^T x + \beta + f(x)$  where  $a_i = f(\mathbb{0}) - f(\mathbf{e}^i)$  and  $\beta = -f(\mathbb{0})$  holds. With this in mind, the second property follows readily from the definition of linearization complexity.  $\square$

The result implies that we only need to analyze linearization complexities of functions  $f$  with  $f(\mathbb{0}) = f(\mathbf{e}^i) = 0$  for  $i = 1, 2, \dots, n$ .

**Linearization complexity.** We continue with simple properties of the linearization complexity. Our first observation is that for arbitrary  $\mathcal{G} \subseteq \mathcal{B}$ ,  $lc_{\mathcal{G}}(f) > 0$  indicates that  $f$  is actually nonlinear over  $\{0, 1\}^n$ .

**Proposition 2.** *For  $\mathcal{G} \subseteq \mathcal{B}$ , a function  $f$  has  $lc_{\mathcal{G}}(f) = 0$  if and only if  $f$  is affine.*

We can rephrase the result in terms of the nonlinear part of  $f$  as follows. It holds  $lc_{\mathcal{G}}(\tilde{f}) = 0$  if and only if  $\tilde{f}$  is the zero function.

The second property is the monotonicity of  $lc_{\mathcal{G}}(f)$  with respect to the family  $\mathcal{G}$ , which also follows from the definition.

**Proposition 3.** *Let  $\mathcal{G}' \subseteq \mathcal{G}$  and consider  $f : \{0, 1\}^n \rightarrow \mathbb{R}$ . Then  $lc_{\mathcal{G}'}(f) \geq lc_{\mathcal{G}}(f)$ .*

The third property is an easy upper bound.

**Proposition 4.** *Every function  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  satisfies  $lc_{\mathcal{M}}(f) \leq 2^n - n - 1$ .*

*Proof.* Let  $f : \{0, 1\}^n \rightarrow \mathbb{R}$  and consider the polynomial

$$g(x) := \sum_{X \subseteq [n]} f(\chi(X)) \cdot \prod_{i \in X} x_i \cdot \prod_{i \notin X} (1 - x_i),$$

where  $\chi(X) \in \{0, 1\}^n$  denotes the characteristic vector of  $X$ , defined via  $\chi(X)_i = 1 \iff i \in X$ . Note that  $g$  has degree at most  $n$  and hence is the sum of  $a^T x + \beta$  (for suitable  $a \in \mathbb{R}^n$  and  $\beta \in \mathbb{R}$ ) and at most  $2^n - n - 1$  monomials of degree greater than 1. By construction, we have  $f(x) = g(x)$  for each  $x \in \{0, 1\}^n$ .  $\square$

**Random polynomials.** Our first larger result essentially states that having a small linearization complexity is an exceptional property in a probabilistic sense.

**Theorem 5.** *Consider a family  $\mathcal{G}$  of functions with  $\mathcal{M} \subseteq \mathcal{G} \subseteq \mathcal{B}$ . Then the subset of functions  $f : \{0,1\}^n \rightarrow \mathbb{R}$  with  $lc_{\mathcal{G}}(f) < 2^n - n - 1$  is a null set (in the set of pseudo-Boolean functions  $f : \{0,1\}^n \rightarrow \mathbb{R}$ ).*

The statement and its proof are similar to Theorem 1 in [4], basically exploiting that pseudo-Boolean functions form a vector space of dimension  $2^n$  and that any subspace spanned by fewer functions constitutes a null set.

*Proof.* Consider a fixed number  $n$  of variables and the subset  $\mathcal{G}_n \subseteq \mathcal{G}$  of functions that map from  $\{0,1\}^n$ . We consider the system of equations

$$\bar{x}^\top a + \beta + \sum_{g \in \mathcal{G}_n : g(\bar{x})=1} c_g = f(\bar{x}) \quad \forall \bar{x} \in \{0,1\}^n \quad (6)$$

in variables  $a_i \in \mathbb{R}$  (for  $i = 1, 2, \dots, n$ ),  $\beta \in \mathbb{R}$  and  $c_g \in \mathbb{R}$  for all  $g \in \mathcal{G}_n$ . On the one hand, every solution  $(a, \beta, c)$  yields a linearization by letting  $g_1, g_2, \dots, g_k$  be those  $g \in \mathcal{G}_n$  for which  $c_g \neq 0$  and letting  $b_i := c_{g_i}$  for  $i = 1, 2, \dots, k$ . Note that its size  $k$  is equal to the number of nonzero components of the  $c$ -vector. On the other hand, every linearization, say with  $a \in \mathbb{R}^n$ ,  $\beta \in \mathbb{R}$ ,  $b \in \mathbb{R}^k$  and  $g_1, g_2, \dots, g_k \in \mathcal{G}_n$ , corresponds to a solution  $(a, \beta, c)$ , where  $c_{g_i} := b_i$  for  $i = 1, 2, \dots, k$  and  $c_g := 0$  for all  $g \in \mathcal{G}_n \setminus \{g_1, g_2, \dots, g_k\}$ .

Note that by Proposition 4, every function  $f$  has a solution to (6), which implies that the system has full row rank  $2^n$ . Hence, the solutions  $(a, \beta, c)$  with the smallest number of nonzeros in the  $c$ -vector arise as solutions of a  $2^n$ -by- $2^n$  subsystem  $D(a^\top, \beta, c^\top)^\top = e$  of (6), where  $e$  contains as entries the values  $f(\bar{x})$  for all  $\bar{x} \in \{0,1\}^n$  and  $D$  is some invertible submatrix of the coefficient matrix of (6).

Now note that the linearization has size less than  $2^n - n - 1$  if and only if  $D^{-1}e$  has a 0-entry. However, the set of right-hand side vectors  $e$  having such a property is a null set (i.e., it has Lebesgue-measure 0), which concludes the proof.  $\square$

**Corollary 6.** *Consider a family  $\mathcal{G}$  of functions with  $\mathcal{M} \subseteq \mathcal{G} \subseteq \mathcal{B}$ . A polynomial  $p : \{0,1\}^n \rightarrow \mathbb{R}$  whose coefficients are chosen randomly according to any absolutely continuous probability distribution has  $lc_{\mathcal{G}}(f) = 2^n - n - 1$  with probability 1.*

**Products of variables.** Proposition 3 implies that the worst results for  $lc_{\mathcal{G}}(f)$  will be obtained if the family  $\mathcal{G}$  of functions  $g$  is small. A minimal choice that still guarantees finiteness of  $lc_{\mathcal{G}}$  is  $\mathcal{G} = \mathcal{M}$ . For this choice we characterize the linearization complexity in case a polynomial representation of  $f$  is known. In fact, the following result follows readily from the well-known fact that every pseudo-Boolean function has a unique multilinear polynomial representation [23, 24] (see also Proposition 2 in [8]).

**Proposition 7.** *Let  $f : \{0,1\}^n \rightarrow \mathbb{R}$ . Then  $lc_{\mathcal{M}}(f)$  is equal to the number of monomials (with a nonzero coefficient) of degree at least 2 of the (unique) multilinear polynomial  $p$  with  $p(x) = f(x)$  for all  $x \in \{0,1\}^n$ .*

**Arbitrary functions.** We already observed that  $lc_{\mathcal{B}}(f) \leq 2^n - n - 1$  holds for any function  $f : \{0,1\}^n \rightarrow \mathbb{R}$ . It turns out that the family  $\mathcal{B}$  is so general that the corresponding linearization complexity only depends on the range of  $\tilde{f}$ . To make this precise, we define for a vector  $w \in \mathbb{R}^k$  its *partial sum set*  $pss(w)$  as the set of partial sums  $\sum_{i \in I} w_i$  over all subsets  $I \subseteq \{1, 2, \dots, k\}$ . We also allow for  $k = 0$  and define  $pss(w) = \{0\}$  in this case.

**Theorem 8.** *Let  $f : \{0,1\}^n \rightarrow \mathbb{R}$  and let  $Y := \{\tilde{f}(x) : x \in \{0,1\}^n\}$  be the range of its nonlinear part. Then  $lc_{\mathcal{B}}(f)$  is equal to the smallest dimension  $k$  of a vector  $w \in \mathbb{R}^k$  with  $pss(w) \supseteq Y$ .*

*Proof.* Let  $f$  and  $Y$  be as in the theorem and consider a vector  $w \in \mathbb{R}^k$  (for some  $k$ ) with  $pss(w) \supseteq Y$ . We now construct a linearization of  $f$  of size  $k$ . For each  $x \in \{0,1\}^n$  we have  $\tilde{f}(x) \in Y \subseteq pss(w)$ , which implies that there must be a subset  $I_x \subseteq \{1, 2, \dots, k\}$  with  $\sum_{i \in I_x} w_i = \tilde{f}(x)$ . We now define, for  $i = 1, 2, \dots, k$ , the function  $g_i : \{0,1\}^n \rightarrow \{0,1\}$  such that

$$g_i(x) = 1 \iff i \in I_x$$

holds. Moreover, we use  $b := w$  and observe  $\sum_{i=1}^k b_i g_i(x) = \sum_{i \in I_x}^k w_i = \tilde{f}(x)$  for each  $x \in \{0, 1\}^n$ , which establishes  $lc_{\mathcal{B}}(\tilde{f}) \leq k$  and by Proposition 1 also  $lc_{\mathcal{B}}(f) \leq k$ . For the reverse direction, suppose  $lc_{\mathcal{B}}(f) = k$ . Again by Proposition 1 there exist vectors  $a \in \mathbb{R}^n$ ,  $b \in \mathbb{R}^k$ , scalar  $\beta \in \mathbb{R}$  and functions  $g_1, g_2, \dots, g_k : \{0, 1\}^n \rightarrow \{0, 1\}$  such that

$$f(x) = a^\top x + \beta + \sum_{i=1}^k b_i g_i(x)$$

holds for all  $x \in \{0, 1\}^n$ . For the nonlinear part we obtain

$$\tilde{f}(x) = \sum_{i=1}^k b_i \tilde{g}_i(x).$$

Now consider a  $y \in Y$ , i.e., the function value  $y = \tilde{f}(x)$  for some  $x \in \{0, 1\}^n$ . By using the set  $I := \{i \in \{1, 2, \dots, k\} : \tilde{g}_i(x) = 1\}$  we obtain that  $y \in pss(b)$  holds. This implies  $pss(b) \supseteq Y$ , and hence the choice  $w := b$  concludes the proof.  $\square$

The intuition behind Theorem 8 is the following: since there are no restrictions on the structure of the functions  $g$  used, the fact that the possible arguments  $x$  of  $f$  are actually (binary) vectors is not relevant, i.e., one can think of just  $2^n$  different inputs without any structure. In the next section we will use this viewpoint to construct a small linearization for an example application.

### 3 Application to Low Autocorrelation Binary Sequences

In this section we consider the low auto-correlation binary sequences problem, which arises in theoretical physics when studying ground states of the Bernasconi model [5, 22]. An extension of the problem was considered in [27], which involves two parameters to specify a problem instance. In the Bernasconi model, both parameters are equal, leading to the following formulation.

$$\min \sum_{d=1}^{N-1} \left( \sum_{i=1}^{N-d} s_i s_{i+d} \right)^2 \quad (7a)$$

$$\text{s.t. } s_i \in \{-1, +1\} \quad \forall i \in \{1, 2, \dots, N\} \quad (7b)$$

The state-of-the-art method for solving this problem is a highly parallelized combinatorial branch-and-bound algorithm [30] which builds upon earlier work [28, 29]. The optima for the problem are known for all  $N \leq 66$ . Recently, also methods based on quadratic convex reformulations were applied successfully to the problem [7, 17].

In this section we investigate IP models based on binary linearizations for this problem. To this end, let us first rephrase the problem by substituting  $s_i \in \{-1, +1\}$  with  $2x_i - 1$  for  $x_i \in \{0, 1\}$  for all  $i \in \{1, 2, \dots, N\}$ . We obtain

$$\min f_N^{\text{bern}}(x) := \sum_{d=1}^{N-1} \left( \sum_{i=1}^{N-d} (4x_i x_{i+d} - 2x_i - 2x_{i+d} + 1) \right)^2 \quad (8a)$$

$$\text{s.t. } x_i \in \{0, 1\} \quad \forall i \in \{1, 2, \dots, N\}. \quad (8b)$$

Let  $f^{\text{bern}}(N)(x) = \sum_{m=1}^t a_m g_{I_m, \emptyset}(x)$  be the decomposition of  $f^{\text{bern}}$  into its  $t$  monomials. With this notation, the standard IP reformulation (4) applied to  $f_N^{\text{bern}}$  reads as follows.

$$\min \sum_{m=1}^t a_m z_{I_m} \quad (9a)$$

$$\text{s.t. } z_I \leq x_i \quad \forall i \in I_m, \quad m = 1, 2, \dots, t \quad (9b)$$

$$1 - z_{I_m} \leq \sum_{i \in I_m} (1 - x_i) \quad m = 1, 2, \dots, t \quad (9c)$$

$$z_{I_m} \in \{0, 1\} \quad m = 1, 2, \dots, t \quad (9d)$$

$$x_i \in \{0, 1\} \quad i = 1, 2, \dots, N \quad (9e)$$

The following proposition establishes the asymptotic number of monomials of  $f^{\text{bern}}(x)$ , which implies that (9) has  $\Theta(N^3)$  many variables and constraints.

**Proposition 9.** *For  $N \geq 3$ , the number of monomials of  $f^{\text{bern}}(x)$  is cubic, model (9) has  $\Theta(N^3)$  variables and constraints.*

*Proof.* The polynomial  $f^{\text{bern}}(x)$  has degree 4. Since there are at most  $\mathcal{O}(N^3)$  monomials of degree at most 3, it suffices to prove that the number of nonzero degree 4 monomials of  $f^{\text{bern}}(x)$  is cubic. Since a monomial of degree 4 arises only as a product of  $x_i x_{i+d}$  and  $x_{i'} x_{i'+d}$  and such terms only have positive coefficients, there are no cancellations. Consider such a monomial, i.e.,  $x_p x_q x_r x_s$  for  $p < q < r < s$ . It occurs if and only if either  $q = p + d$  and  $s = r + d$  holds (where  $d$  is the loop index in (8a)) or  $r = p + d$  and  $s = q + d$ . This condition is equivalent to  $p + s = q + r$ . It follows that the number of such tuples  $(p, q, r, s)$  is in  $\mathcal{O}(N^3)$ .

It remains to show that there also exist  $\Omega(N^3)$  degree 4 monomials. To this end, let  $k := \lfloor N/4 \rfloor$ . We consider any  $i \in \{1, 2, \dots, k\}$ , any  $j \in \{2k + 1, 2k + 2, \dots, 3k\}$  and any  $d \in \{1, 2, \dots, k - 1\}$ . Note that for each such  $(i, j, d)$ , the term  $x_i x_{i+d} x_j x_{j+d}$  arises as a monomial of  $f^{\text{bern}}$ . Moreover, all these monomials are distinct. Due to  $k \in \Omega(N)$ , the number of monomials is in  $\Omega(k^3) = \Omega(N^3)$ .  $\square$

**Partial sum sets.** With Proposition 9 in mind a natural follow-up question is whether there exist formulations of smaller asymptotic size. By Theorem 8 we can investigate the partial sum set of the domain  $Y$  of its nonlinear part. The set  $Y$  is nontrivial to characterize, in particular since computing  $\min(Y)$  is equivalent to solving (7). However, its size  $|Y|$  constitutes an upper bound on the smallest dimension  $k$  of a vector  $w \in \mathbb{R}^k$  with  $pss(w) \supseteq Y$ . Now observe that the sum  $\sum_{i=1}^{N-d} s_i s_{i+d}$  can attain  $N + 1 - d$  different values. Consequently there are also only  $N + 1 - d$  different squared such values. Summation over  $d \in \{1, 2, \dots, N - 1\}$  establishes our main theoretical result for the application.

**Theorem 10.** *For  $N \geq 3$  we have  $lc_{\mathcal{B}}(f_N^{\text{bern}}) \leq N^2$ .*

**Model with value indicators.** Theorem 10 only yields the existence of an IP formulation with  $N^2$  auxiliary variables. Instead of following its proof to derive such a formulation we directly exploit the main observation that led to the quadratic size, namely that every element from the domain is a sum of squares and that, for every  $d$ , at most  $N + 1 - d$  such squared numbers can actually arise. This yields the idea of introducing binary variables for indicating (for each  $d$  separately) which number was actually squared. To this end, we define the set  $L_d := \left\{ \sum_{i=1}^{N-d} s_i s_{i+d} : s \in \{-1, 1\}^N \right\}$  of distinct function values of the inner sum of (7a). Since  $s_i \cdot s_{i+d} \in \{-1, +1\}$  holds, we have

$$L_d \subseteq \{-(N - d), -(N - d) + 2, \dots, (N - d) - 2, (N - d)\}.$$

We obtain  $|L_d| \leq N + 1 - d$  for all  $d \in \{1, 2, \dots, N - 1\}$ . Now we consider, for each  $d \in \{1, 2, \dots, N - 1\}$  and each  $\ell \in L_d$ , the function  $g_{d,\ell} : \{0, 1\}^N \rightarrow \{0, 1\}$  that indicates whether the inner sum of (7a) for some  $d$  is equal to  $\ell$ . More precisely, it shall satisfy  $g_{d,\ell}(x) = 1$  if and only if  $\sum_{i=1}^{N-d} (2x_i - 1)(2x_{i+d} - 1) = \ell$ . Since for each  $d$  and each  $x \in \{0, 1\}^n$ , precisely one of the functions  $g_{d,\ell}(x)$  is equal to 1, we obtain

$$f_N^{\text{bern}}(x) = \sum_{d=1}^{N-1} \sum_{\ell \in L_d} \ell^2 g_{d,\ell}(x).$$

This clearly leads to the following formulation with only  $\mathcal{O}(N^2)$  variables by enforcing  $z_{d,\ell} = g_{d,\ell}(x)$  via constraints (2). However, these are exponentially many constraints.

$$\min \sum_{d=1}^{N-1} \sum_{\ell \in L_d} \ell^2 z_{d,\ell} \quad (10a)$$

$$\text{s.t.} \quad \sum_{j:\bar{x}_j=0} x_j + \sum_{j:\bar{x}_j=1} (1 - x_j) + z_{d,\ell} \geq 1 \quad \forall d \in \{1, 2, \dots, N-1\}, \forall \ell \in L_d, \quad (10b)$$

$$\forall \bar{x} \in \{0, 1\}^n : g_{d,\ell}(\bar{x}) = 1$$

$$\sum_{j:\bar{x}_j=0} x_j + \sum_{j:\bar{x}_j=1} (1 - x_j) + (1 - z_{d,\ell}) \geq 1 \quad \forall d \in \{1, 2, \dots, N-1\}, \forall \ell \in L_d, \quad (10c)$$

$$\forall \bar{x} \in \{0, 1\}^n : g_{d,\ell}(\bar{x}) = 0$$

$$x_i \in \{0, 1\} \quad \forall i \in \{1, 2, \dots, N\} \quad (10d)$$

$$z_{d,\ell} \in \{0, 1\} \quad \forall d \in \{1, 2, \dots, N-1\}, \forall \ell \in L_d \quad (10e)$$

This formulation can be improved by adding more auxiliary variables (in addition to  $z_{d,\ell} = g_{d,\ell}(x)$ ). The idea is to introduce variables  $y_{i,j}$  that indicate for all  $i, j$  (with  $i \neq j$ ) whether the product  $s_i s_j$  is +1 or -1.

$$\min \sum_{d=1}^{N-1} \sum_{\ell \in L_d} \ell^2 z_{d,\ell} \quad (11a)$$

$$\text{s.t.} \quad x_i + x_j \geq 1 - y_{i,j} \quad \forall i, j \in \{1, 2, \dots, N\} : i < j \quad (11b)$$

$$x_i - x_j \geq y_{i,j} - 1 \quad \forall i, j \in \{1, 2, \dots, N\} : i < j \quad (11c)$$

$$-x_i + x_j \geq y_{i,j} - 1 \quad \forall i, j \in \{1, 2, \dots, N\} : i < j \quad (11d)$$

$$x_i + x_j \leq y_{i,j} + 1 \quad \forall i, j \in \{1, 2, \dots, N\} : i < j \quad (11e)$$

$$\sum_{\ell \in L_d} z_{d,\ell} = 1 \quad \forall d \in \{1, 2, \dots, N-1\} \quad (11f)$$

$$\sum_{i=1}^{N-d} (2y_{i,i+d} - 1) = \sum_{\ell \in L_d} \ell z_{d,\ell} \quad \forall d \in \{1, 2, \dots, N-1\} \quad (11g)$$

$$x_i \in \{0, 1\} \quad \forall i \in \{1, 2, \dots, N\} \quad (11h)$$

$$y_{i,j} \in \{0, 1\} \quad \forall i, j \in \{1, 2, \dots, N\} : i < j \quad (11i)$$

$$z_{d,\ell} \in \{0, 1\} \quad \forall d \in \{1, 2, \dots, N-1\}, \forall \ell \in L_d \quad (11j)$$

**Corollary 11.** For  $N \geq 3$ , the integer program (11) correctly models the low auto-correlation binary sequences problem (7) and has  $\mathcal{O}(N^2)$  variables and constraints.

*Proof.* Similar to (8) we model  $s_i \in \{-1, +1\}$  by  $s_i = 2x_i - 1$ , i.e.,  $x_i = 1$  if and only if  $s_i = 1$ . Constraints (11b)–(11e) enforce that  $2y_{i,j} - 1 = s_i \cdot s_j = (2x_i - 1)(2x_j - 1)$  holds, i.e.,  $y_{i,j} = 1$  if and only if  $x_i = x_j$  holds. For each  $d \in \{1, 2, \dots, N-1\}$ , equation (11f) implies that  $z_{d,\ell} = 1$  holds for exactly one  $\ell \in L_d$ . Since the left-hand side of (11g) is equal to the inner sum of (7a), the right-hand side implies that  $z_{d,\ell} = 1$  holds exactly for  $\ell$  being the value of that inner sum. Hence, the right-hand side of (11g) is the corresponding  $\ell \in L_d$ . This implies that the contribution of all  $z_{d,\ell}$  for some  $d \in \{1, 2, \dots, N-1\}$  to the objective (11a) is equal to  $\ell^2$ , where  $\ell$  is the value of the inner sum of (7a). We conclude that (7a) is indeed equal to  $f_N^{\text{bern}}(x)$ .  $\square$

**Experimental evaluation.** The numbers of variables and constraints are only one factor that has impact on the solution time of an IP. For instance, the quality of the dual (in our case lower) bounds on the optimum and the ability to find good primal solutions is extremely important. Hence, we compare the new IP model (11) to the standard formulation (9).

We tried to solve the low auto-correlation binary sequences problem with both models using the SCIP solver framework [6]<sup>1</sup>. Note that both approaches are too slow to compete with the most recent state-of-the-art approach that is based on combinatorial branch-and-bound with which the problem could

<sup>1</sup>The code for generating the instances can be found at [github.com/discopt/multilinear-instance-generators](https://github.com/discopt/multilinear-instance-generators).



be solved to optimality up to  $N = 66$  [30]. More precisely, we only report about instances that can be solved by at least one of the models to optimality within 1 h using version 8.0.2 of SCIP using SoPlex as an LP solver. Also note that we ran our experiments only on a single core (with an Intel CPU on 2.1 GHz) while the computations for  $N = 66$  were done using 248 cores and took about 55 days [30]. Moreover, we strengthened none of the two models to keep the comparison fair. Table 1 shows our results. Note that for  $N > 22$ , SCIP could not solve the standard formulation (9) within one hour. In fact, the lower bound obtained after that time was still a negative number. Note that by construction, model (11) only provides nonnegative bounds. We conclude that our new formulation outperforms the standard formulation, not only by means of size, but also by means of quality of the bound obtained from the LP relaxation.

Table 1: Comparison of IP models (9) and (11). The first two columns show the parameter  $N$  and the optimum. Moreover, for each model, the number of variables and constraints, the LP bound of the relaxation (Bnd), the number of solved branch-and-bound nodes, as well as the total solution time are reported. The symbol  $\ominus$  indicates that the computation reached the timeout of 3600 s.

N	OPT	Standard IP (9)					Value indicator IP (11)				
		Vars	Cons	Bnd	Nodes	Time	Vars	Cons	Bnd	Nodes	Time
3	1	5	4	1	0	0.01 s	17	16	1	1	0.0 s
4	2	16	40	-74	1	0.03 s	31	30	2	1	0.01 s
5	2	28	82	-226	1	0.04 s	49	48	2	1	0.01 s
6	7	47	153	-529	5	0.12 s	71	70	3	1	0.04 s
7	3	75	261	-985	23	0.31 s	97	96	3	1	0.06 s
8	8	113	411	-1668	35	0.43 s	127	126	4	1	0.08 s
9	12	160	599	-2580	89	0.81 s	161	160	4	1	0.22 s
10	13	222	850	-3795	191	1.59 s	199	198	5	19	0.27 s
11	5	297	1156	-5315	425	2.79 s	241	240	5	1	0.42 s
12	10	388	1530	-7214	881	5.63 s	287	286	6	70	0.98 s
13	6	496	1976	-9494	1622	14.73 s	337	336	6	4	0.35 s
14	19	623	2503	-12 229	2173	30.88 s	391	390	7	732	5.01 s
15	15	769	3111	-15 421	4049	72.4 s	449	448	7	1113	4.35 s
16	24	939	3821	-19 144	9372	152.62 s	511	510	8	1452	10.45 s
17	32	1130	4621	-23 400	19 265	317.73 s	577	576	8	5551	36.73 s
18	25	1346	5528	-28 263	41 235	565.69 s	647	646	9	12 379	54.87 s
19	29	1589	6550	-33 735	92 657	1042.12 s	721	720	9	14 396	64.73 s
20	26	1860	7692	-39 890	206 383	1964.38 s	799	798	10	9751	58.45 s
21	26	2158	8950	-46 730	-	$\ominus$	881	880	10	22 949	122.93 s
22	39	2489	10 349	-54 329	-	$\ominus$	967	966	11	41 491	202.87 s
23	47	2851	11 881	-62 689	-	$\ominus$	1057	1056	11	106 590	415.89 s
24	36	3247	13 559	-71 884	-	$\ominus$	1151	1150	12	115 220	445.95 s
25	36	3678	15 387	-81 916	-	$\ominus$	1249	1248	12	88 708	414.45 s
26	45	4146	17 374	-92 859	-	$\ominus$	1351	1350	13	386 737	1716.33 s
27	37	4651	19 520	-104 715	-	$\ominus$	1457	1456	13	266 951	1412.36 s
28	50	5198	21 846	-117 558	-	$\ominus$	1567	1566	14	611 390	3078.0 s
29	62	5784	24 340	-131 390	-	$\ominus$	1681	1680	14	-	$\ominus$

We also tried to determine a small formulation for problem (7) for  $\mathcal{G} = \mathcal{C}$ , i.e., for the family of potentially complemented products of variables. To this end, system (6) can be augmented by binary variables to indicate whether a  $g \in \mathcal{G}$  is used (that is, has a nonzero multiplier). The minimization of the sum of these binary variables yields  $lc_{\mathcal{G}}(f)$ . Clearly, this approach only works for small sizes  $|\mathcal{G}|$ . Our results for  $N \in \{3, 4, 5, 6\}$  were quite disappointing – at least when considering only functions  $g \in \mathcal{C}$  of degree at most 5, the linearization complexity is minimized by the linearization that just uses monomials. Hence, we do not expect that one can gain much by allowing complements of variables for the low auto-correlation problem.



## 4 Open Problems

We hope to have convinced the reader that the linearization complexity is a useful concept. Nevertheless, many unsolved problems remain, and we use the rest of the paper to present them.

**More families.** We discussed various families of functions  $g$  to use for linearization, namely the products of variables  $\mathcal{M}$ , the potentially complemented products of variables  $\mathcal{C}$ , arbitrary Boolean functions  $\mathcal{B}$ , and finally functions that indicate whether an expression of a certain type attains a certain value. Another family is induced by any quadratization strategy: after applying such a strategy to obtain a quadratization function  $g$  as in (5) one can linearize the latter, e.g., in a monomial-wise fashion. We believe that there are more such interesting families.

**Formulations.** Perfect formulations are known for the first two considered linearizations when considering every linearization variables separately. Strengthening the joint formulation for multiple linearization variables from  $\mathcal{C}$  is subject of current research, e.g., by means of studying multilinear polytopes [9, 10, 11, 12, 13, 14, 15, 16, 31]. For  $\mathcal{B}$  we cannot hope to identify perfect formulations since this encompasses arbitrary binary sets. However, for other functions it is interesting to investigate which inequalities are best to add in order to apply such a function. For instance, our formulation (11) worked well because we did not model the meaning of each  $z$ -variable individually, but because we considered multiple of them in a combined fashion in constraints (11f) and (11g).

**Bounding techniques.** While we could interpret the linearization complexities for  $\mathcal{M}$  and for  $\mathcal{B}$ , little is known about  $\mathcal{C}$  and  $\mathcal{V}$ . The first natural question is which properties of a polynomial allow to recast it as a sum of only few products of potentially complemented variables.

**Algorithmic questions.** After settling how  $f$  could be encoded (expressing it as a polynomial is only one possibility), there are many algorithmic problems related to linearization complexity. Most importantly, the complexity of the computation or approximation of  $lc_{\mathcal{C}}(f)$  or of  $lc_{\mathcal{B}}(f)$  is open. For practical purposes it would be very interesting to find small linearizations based on  $\mathcal{C}$  because the actual formulations are essentially the same as those for  $\mathcal{M}$  which are reasonably well understood.

**Approximations.** While Theorem 5 and Corollary 6 indicate that linearizations with small linearization complexity are rare, one may still consider approximate linearizations, i.e., small linearizations of a function  $f'$  that is very close to  $f$ . Instead of abandoning exactness one can also try to pursue a related approach by finding a linearization that may not be small but for which only few of the weights  $b_i$  in (1) are large (in absolute value). While the resulting IP formulation would still have many variables, one could apply strengthening techniques only on those few linearizations that are most important for the value of  $f$ . Then, relaxation errors for the remaining variables (with small  $|b_i|$ ) will not have a big impact on the overall objective value, which would yield better bounds.

## References

- [1] Warren P. Adams and Hanif D. Sherali. A Tight Linearization and an Algorithm for Zero-One Quadratic Programming Problems. *Management Science*, 32(10):1274–1290, 1986. [doi:10.1287/mnsc.32.10.1274](https://doi.org/10.1287/mnsc.32.10.1274).
- [2] Warren P. Adams and Hanif D. Sherali. Linearization Strategies for a Class of Zero-One Mixed Integer Programming Problems. *Operations Research*, 38(2):217–226, 1990. [doi:10.1287/ope.38.2.217](https://doi.org/10.1287/ope.38.2.217).
- [3] Warren P. Adams and Hanif D. Sherali. Mixed-integer bilinear programming problems. *Mathematical Programming*, 59(1):279–305, Mar 1993. [doi:10.1007/BF01581249](https://doi.org/10.1007/BF01581249).
- [4] Martin Anthony, Endre Boros, Yves Crama, and Aritanan Gruber. Quadratic reformulations of nonlinear binary optimization problems. *Mathematical Programming*, 162(1):115–144, Mar 2017. [doi:10.1007/s10107-016-1032-4](https://doi.org/10.1007/s10107-016-1032-4).

- [5] Jakob Bernasconi. Low autocorrelation binary sequences: statistical mechanics and configuration space analysis. *Journal de Physique*, 141(48):559–567, 1987. doi:[10.1051/jphys:01987004804055900](https://doi.org/10.1051/jphys:01987004804055900).
- [6] Ksenia Bestuzheva, Mathieu Besançon, Wei-Kun Chen, Antonia Chmiela, Tim Donkiewicz, Jasper van Doornmalen, Leon Eifler, Oliver Gaul, Gerald Gamrath, Ambros Gleixner, et al. The SCIP Optimization Suite 8.0. *arXiv preprint arXiv:2112.08872*, 2021. URL: <https://arxiv.org/abs/2112.08872>.
- [7] Alain Billionnet and Sourour Elloumi. Using a Mixed Integer Quadratic Programming Solver for the Unconstrained Quadratic 0-1 Problem. *Mathematical Programming Computation*, 109(1):55–68, 2007. doi:[10.1007/s10107-005-0637-9](https://doi.org/10.1007/s10107-005-0637-9).
- [8] Endre Boros and Peter L. Hammer. Pseudo-Boolean optimization. *Discrete Applied Mathematics*, 123(1):155–225, 2002. doi:[10.1016/S0166-218X\(01\)00341-9](https://doi.org/10.1016/S0166-218X(01)00341-9).
- [9] Christoph Buchheim, Yves Crama, and Elisabeth Rodríguez-Heck. Berge-acyclic multilinear 0–1 optimization problems. *European Journal of Operational Research*, 273(1):102–107, 2019. doi:[10.1016/j.ejor.2018.07.045](https://doi.org/10.1016/j.ejor.2018.07.045).
- [10] Yves Crama and Elisabeth Rodríguez-Heck. A class of valid inequalities for multilinear 0–1 optimization problems. *Discrete Optimization*, 25:28–47, 2017. doi:[10.1016/j.disopt.2017.02.001](https://doi.org/10.1016/j.disopt.2017.02.001).
- [11] Alberto Del Pia and Silvia Di Gregorio. Chvátal rank in binary polynomial optimization. *INFORMS Journal on Optimization*, 2021. doi:[10.1287/ijoo.2019.0049](https://doi.org/10.1287/ijoo.2019.0049).
- [12] Alberto Del Pia and Aida Khajavirad. A Polyhedral Study of Binary Polynomial Programs. *Mathematics of Operations Research*, 42(2):389–410, 2017. doi:[10.1287/moor.2016.0804](https://doi.org/10.1287/moor.2016.0804).
- [13] Alberto Del Pia and Aida Khajavirad. The multilinear polytope for acyclic hypergraphs. *SIAM Journal on Optimization*, 28(2):1049–1076, 2018. doi:[10.1137/16M1095998](https://doi.org/10.1137/16M1095998).
- [14] Alberto Del Pia and Aida Khajavirad. On decomposability of Multilinear sets. *Mathematical Programming*, 170(2):387–415, 2018. doi:[10.1007/s10107-017-1158-z](https://doi.org/10.1007/s10107-017-1158-z).
- [15] Alberto Del Pia and Aida Khajavirad. The running intersection relaxation of the multilinear polytope. *Mathematics of Operations Research*, 2021. doi:[10.1287/moor.2021.1121](https://doi.org/10.1287/moor.2021.1121).
- [16] Alberto Del Pia and Matthias Walter. Simple odd  $\beta$ -cycle inequalities for binary polynomial optimization. In Karen Aardal and Laura Sanità, editors, *Integer Programming and Combinatorial Optimization*, pages 181–194. Springer International Publishing, 2022. doi:[10.1007/978-3-031-06901-7\\_14](https://doi.org/10.1007/978-3-031-06901-7_14).
- [17] Sourour Elloumi, Amélie Lambert, and Arnaud Lazare. Solving unconstrained 0-1 polynomial programs through quadratic convex reformulation. *Journal of Global Optimization*, 80(2):231–248, 2021. doi:[10.1007/s10898-020-00972-2](https://doi.org/10.1007/s10898-020-00972-2).
- [18] Robert Fortet. Applications de l’algebre de boole en recherche opérationnelle. *Revue Française de Recherche Opérationnelle*, 4(14):17–26, 1960.
- [19] Robert Fortet. L’algebre de boole et ses applications en recherche operationnelle. *Trabajos de Estadística*, 4:17–26, 1960. doi:[10.1007/BF03006558](https://doi.org/10.1007/BF03006558).
- [20] Fabio Furini and Emiliano Traversi. Extended linear formulation for binary quadratic problems. *Optimization Online*, 2013.
- [21] Fred Glover. Improved linear integer programming formulations of nonlinear integer problems. *Management Science*, 22(4):455–460, 1975. doi:[10.1287/mnsc.22.4.455](https://doi.org/10.1287/mnsc.22.4.455).
- [22] Marcel J.E. Golay. The merit factor of long low autocorrelation binary sequences. *IEEE Transactions on Information Theory*, 28(3):543–549, 1982. doi:[10.1109/TIT.1982.1056505](https://doi.org/10.1109/TIT.1982.1056505).

- [23] Peter L. Hammer, Ivo G. Rosenberg, and Sergiu Rudeanu. On the determination of the minima of pseudo-boolean functions. *Studii și Cercetări Matematice*, 14(3):359–364, 1963. (in Rumanian).
- [24] Peter L. Hammer and Sergiu Rudeanu. *Boolean Methods in Operations Research and Related Areas*, volume 7. Springer Science & Business Media, 2012. doi:[10.1007/978-3-642-85823-9](https://doi.org/10.1007/978-3-642-85823-9).
- [25] Robert Jeroslow. On defining sets of vertices of the hypercube by linear inequalities. *Discrete Mathematics*, 11(2):119–124, 1975. doi:[10.1016/0012-365X\(75\)90003-5](https://doi.org/10.1016/0012-365X(75)90003-5).
- [26] Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972. doi:[10.1007/978-1-4684-2001-2\\_9](https://doi.org/10.1007/978-1-4684-2001-2_9).
- [27] Frauke Liers, Enzo Marinari, Ulrike Pagacz, Federico Ricci-Tersenghi, and Vera Schmitz. A non-disordered glassy model with a tunable interaction range. *Journal of Statistical Mechanics: Theory and Experiment*, 2010(05):L05003, 2010. doi:[10.1088/1742-5468/2010/05/L05003](https://doi.org/10.1088/1742-5468/2010/05/L05003).
- [28] Stephan Mertens. Exhaustive search for low-autocorrelation binary sequences. *Journal of Physics A: Mathematical and General*, 29(18):L473, sep 1996. doi:[10.1088/0305-4470/29/18/005](https://doi.org/10.1088/0305-4470/29/18/005).
- [29] Stephan Mertens and Christine Bessenrodt. On the ground states of the bernasconi model. *Journal of Physics A: Mathematical and General*, 31(16):3731–3749, apr 1998. doi:[10.1088/0305-4470/31/16/004](https://doi.org/10.1088/0305-4470/31/16/004).
- [30] Tom Packebusch and Stephan Mertens. Low autocorrelation binary sequences. *Journal of Physics A: Mathematical and Theoretical*, 49(16):165001, 2016. doi:[10.1088/1751-8113/49/16/165001](https://doi.org/10.1088/1751-8113/49/16/165001).
- [31] Elisabeth Rodríguez-Heck. *Linear and quadratic reformulations of nonlinear optimization problems in binary variables*. PhD thesis, Université de Liège, Liège, Belgique, 2018. URL: <https://hdl.handle.net/2268/227242>.
- [32] Hanif D. Sherali and Cihan H. Tuncbilek. A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. *Journal of Global Optimization*, 2(1):101–112, Mar 1992. doi:[10.1007/BF00121304](https://doi.org/10.1007/BF00121304).
- [33] Xiaojie Zhang and Paul H. Siegel. Adaptive cut generation for improved linear programming decoding of binary linear codes. In *2011 IEEE International Symposium on Information Theory Proceedings*, pages 1638–1642, 2011. doi:[10.1109/ISIT.2011.6033822](https://doi.org/10.1109/ISIT.2011.6033822).