On Quadratization of Pseudo-Boolean Functions*

Endre Boros and Aritanan Gruber†
RUTCOR, Rutgers University
640 Bartholomew Road, Piscataway, NJ 08854

Introduction

Set functions, i.e., real mappings form the family of subsets of a finite set to the reals are known and widely used in discrete mathematics for almost a century, and in particular in the last 50 years. If we replace a finite set with its characteristic vector, then the same set function can be interpreted as a mapping from the set of binary vectors to the reals. Such mappings are called pseudo-Boolean functions and were introduced in the works of Peter L. Hammer in the 1960s, see the seminal book (Hammer and Rudeanu 1968). Pseudo-Boolean functions are different from set functions, only in the sense that their algebraic representation, a multilinear polynomial expression, is usually assumed to be available as an input representation:

$$f(x_1, x_2, \ldots, x_n) = \sum_{S \subseteq V} a_S \prod_{j \in S} x_j, \qquad (1)$$

where $V = \{1, 2, ..., n\}$ is the set of variable indices, and where we assume that $x_j \in \mathbb{B} = \{0, 1\}$ for all $j \in V$. Due to this assumption we have $x_j^2 = x_j$, and hence any polynomial expression in such binary variables is indeed equivalent with a multilinear one. In fact it was shown in (Hammer and Rudeanu 1968) that any set function has a unique multilinear polynomial representation. The *degree* of f is defined as the size of the largest set S with a nonzero coefficient in the above (unique) multilinear polynomial representation of f: $\deg(f) = \min\{|S| \mid S \subseteq V,\ a_S \neq 0\}$. Clearly, the degree of a constant function is zero. We say that f is a *quadratic* (resp. *linear*) pseudo-Boolean function, if $\deg(f) \leq 2$ (resp. $\deg(f) \leq 1$).

The problem of minimizing a pseudo-Boolean function (over the set of binary vectors) appears to be the common form of numerous optimization problems, including the well-known MAX-SAT and MAX-CUT problems, and have applications in areas ranging from physics through chip design to computer vision; see e.g., the surveys (Boros and Hammer 2002; Kolmogorov and Zabih 2004; Roth and Black 2009).

Some of these applications lead to the minimization of a quadratic pseudo-Boolean function, and hence such quadratic binary optimization problems received ample attention in the past decades. The survey (Boros and Hammer 2002) describes a large set of computational tools for such problems. One of the most frequently used technique is based on roof-duality (Hammer, Hansen, and Simeone 1984), and aims at finding in polynomial time a simpler form of the given quadratic minimization problem, by fixing some of the variables at their provably optimum value (persistency) and decomposing the residual problem into variable disjoint smaller subproblems, see (Boros and Hammer 1989; Boros et al. 2008). The method in fact was found very effective in computer vision problems, where frequently it can fix up to 80-90% of the variables at their provably optimum value. This algorithm was recoded by computer vision experts and a very efficient implementation, called QPBO, is freely downloadable, see (Rother et al. 2007).

In many applications of pseudo-Boolean optimization the objective function (1) is a higher degree multilinear polynomial. For such problems there are substantially fewer effective techniques available. In particular, there is no analogue to the persistencies (fixing variables at their provably optimum value) provided by roof-duality for the quadratic case. On the other hand, more and more applications would demand efficient methods for the minimization of such higher degree pseudo-Boolean functions; see e.g., (Hansen 1979; Buchheim and Rinaldi 2007; Roth and Black 2009). This increased interest, in particular in the computer vision community, lead to a systematic study of methods to reduce a higher degree minimization problem to a quadratic one; see e.g., (Freedman and Drineas 2005; Ishikawa 2009; 2011; Kolmogorov and Zabih 2004; Rosenberg 1975; Rother et al. 2009; Ramalingam et al. 2011; Zivny, Cohen, and Jeavons 2009a; 2009b; Zivny and Jeavons 2010).

In this paper first we recall known "quadratization" techniques from the literature. Next, we provide several new techniques for quadratization, analyze their effectiveness, and recall recent computer vision applications demonstrating their usefulness (Fix et al. 2011).

*The authors gratefully acknowledge the partial support by NSF grants IIS 0803444 and by CMMI 0856663.

†This author also gratefully acknowledge the partial support by the joint CAPES (Brazil)/Fulbright (USA) fellowship process BEX-2387050/15061676.

Basic Model and Literature Review

Given a pseudo-Boolean function $f : \mathbb{B}^n \to \mathbb{R}$ as in (1), where $\mathbb{R}$ denotes the set of reals, the following minimization problem

$$\min_{x \in \mathbb{B}^n} f(x) \tag{2}$$

is the common form of numerous combinatorial optimization problems. To reduce the above problem to a quadratic minimization problem, we are looking for a quadratic pseudo-Boolean function $g(x, w)$, where $w \in \mathbb{B}^m$ is a set of "new" variables, such that the equality

$$f(x) \;=\; \min_{w \in \mathbb{B}^m} g(x, w) \tag{3}$$

holds for all $x \in \mathbb{B}^n$. We shall call such a g the *quadratization* of f. The major objective in the problem of quadratizing a given pseudo-Boolean function f is to find such a quadratic function g that satisfies (3). Secondary objectives are the minimization of the number of new variables m, and the "submodularity" of g.

Given two binary vectors $x, y \in \mathbb{B}^V$, we define their *disjunction* and *conjunction* respectively by $(x \vee y)_j = x_j \vee y_j$ and $(x \wedge y)_j = x_j \wedge y_j$ for all indices $j \in V$. Then, we call a pseudo-Boolean function $f(x)$ *submodular*, if

$$f(x \vee y) + f(x \wedge y) \leq f(x) + f(y)$$

holds for any two vectors $x, y \in \mathbb{B}^V$. Submodular functions play an important role in optimization, since problem (2) which is NP-hard in general, is known to be polynomially solvable if f is submodular (Grötschel, Lovász, and Schrijver 1981; Iwata, Fleischer, and Fujishige 2000; Schrijver 2000). Let us add that if f is a quadratic pseudo-Boolean function, then it is submodular if and only if all quadratic terms have nonpositive coefficients. This property leads to a very simple, network flow based minimization algorithm (Hammer 1965). In fact the QPBO implementation returns automatically a minimizing solution for submodular inputs. A similarly efficient characterization of submodularity for cubic pseudo-Boolean functions was also given in (Billionnet and Minoux 1985). Recognition of submodularity for pseudo-Boolean functions of degree 4 or higher was shown to be NP-hard (Gallo and Simeone 1989).

Let us note that if a pseudo-Boolean function f given in (1) can be quadratized (3) by a submodular quadratic function g, then f itself must be submodular. It is however not obvious which submodular pseudo-Boolean functions can be quadratized by submodular quadratic functions. As we shall see later, it is easy to show that any cubic submodular function can be quadratized keeping submodularity. However, recent results in (Zivny, Cohen, and Jeavons 2009a; 2009b) show that certain degree 4 submodular pseudo-Boolean functions cannot be represented as in (3) by a submodular quadratic function. This is so, even though we do not limit the number of new variables in (3).

In this paper we study techniques to find functions g satisfying (3) for a given pseudo-Boolean function f with the purpose that such a functions should be "small" and "easy" to minimize. More precisely, we shall compare techniques by evaluating the number of new variables, the number of terms, and the number of positive quadratic terms, which is a vague measure of non-submodularity.

Literature Review

Let us first recall the quadratization method suggested by (Rosenberg 1975), based on the idea of traditional penalty functions. This method replaces a product xy of two binary variables by a new binary variable w (and hence decreases the degree by one of all terms involving both x and y), and adds to f a quadratic penalty function $p(x, y, w)$ such that

$$p(x,y,w) \begin{cases} = 0 & \text{if } w = xy, \\ \geq 1 & \text{otherwise.} \end{cases} \tag{4}$$

Since f is multilinear, we can write it as $f = xyA + B$, where A is a multilinear polynomial not involving x and y, and where B is a multilinear polynomial not involving the product xy. Assume now that p is a quadratic function satisfying (4), and M is a positive real with $M > \max |A|$, where the maximization is taken over all binary assignments of the variables of A. Then, the function $\widetilde{f} = zA + B + Mp$ on $n + 1$ variables has the same minima as f. (Rosenberg 1975) showed that

$$p(x, y, w) \;=\; xy - 2xw - 2yw + 3w$$

is a quadratic function satisfying (4). The above idea then can be applied recursively, until the resulting function $\widetilde{f}$ becomes quadratic. It is easy to see that this is a polynomial transformation, e.g., one never needs more than $O(n^2 \log d)$ new variables, where d is the degree of f. The drawback of this approach is that the resulting quadratic function has many "large" coefficients, due to the recursive application of the "big M" substitution. It also introduces many positive quadratic terms, even if the input f is a nice submodular function. These two effects make the minimization of the resulting $\widetilde{f}$ a hard problem, even in approximative sense.

Of course, it would be a simpler approach to replace the product $xy \cdots z$ of several variables by a new variable w and enforce the equality $xy \cdots z = w$ by a quadratic penalty function. It is easy to see that this is not possible with more than two variables.

Let us also note that finding with this approach a quadratization with the minimum number of variables is itself an NP-hard problem. To see this let us consider the cubic pseudo-Boolean function

$$f(x_0, x_1, ..., x_n) \;=\; \sum_{(i,j) \in E} x_0 x_i x_j$$

where E is the edge set of a graph G on vertex set $V = \{1, 2, ..., n\}$. It is easy to verify that for any quadratization of f, there is one with no more new variables, in which we substitute by new variables only products of the form $x_0 x_i$, $i \in C$ for a subset $C \subseteq V$ and the "optimal" quadratization corresponds to a minimum size vertex cover C of G.

A simple quadratization of negative monomials was introduced recently by (Kolmogorov and Zabih 2004) for degree

3 monomials, and by (Freedman and Drineas 2005) for arbitrary degree monomials.

$$-x_1x_2\cdots x_d \;=\; \min_{w\in\mathbb{B}} w\left((d-1)-\sum_{j=1}^{d} x_j\right). \qquad (5)$$

Remarkably, all quadratic terms have negative coefficients in this transformation. In particular, if f involves only negative higher degree terms, then the application of (5) yields a submodular quadratization of it. It is also a nice feature of this approach that it does not introduces "large" coefficients. Let us also remark that for every minimizing assignment of f there is a corresponding minimum of the quadratized form g obtained by the repeated applications of (5) satisfying $w = x_1x_2\cdots x_d$. Thus, the above transformation can also be viewed as a substitution of a higher degree product.

Let us note that this transformation can be extended easily for positive monomials, as well. For this note first that the equality in (5) is based only on the fact that the symbols "x_i" stand for a binary value. Thus, introducing *negated literals* $\overline{x}_i = 1 - x_i$ we can write

$$\begin{aligned} x_1x_2\cdots x_d - x_{d-1}x_d \;&=\; -\sum_{i=1}^{d-2}\overline{x}_i\prod_{j=i+1}^{d} x_j \\ &=\; \min_{w\in\mathbb{B}^{d-2}}\sum_{i=1}^{d-2} w_i(d-i-\overline{x}_i-\sum_{j=i+1}^{d} x_j) \end{aligned} \qquad (6)$$

Hence (5) implies a quadratization of positive monomials as well. For a degree d term we need $d-2$ new variables and get $d-1$ positive (non-submodular) quadratic terms.

Let us add that if a subset of the variables are negated on the left in (5) then the corresponding quadratic terms will have positive coefficients on the right hand side. In particular, if all variables are negated, then all quadratic terms have positive coefficients. Since the new variable w does not appear elsewhere we can replace it with its negation, not changing the minimization in this way, and get again a submodular quadratization

$$\begin{aligned} -\overline{x}_1\overline{x}_2\cdots\overline{x}_d \;&=\; \min_{w\in\mathbb{B}} w\left((d-1)-\sum_{j=1}^{d}\overline{x}_j\right) \\ &=\; -1+\sum_{j=1}^{d} x_j + \min_{w'\in\mathbb{B}} w'\left(1-\sum_{j=1}^{d} x_j\right) \end{aligned} \qquad (7)$$

where $w' = 1 - w$. Let us call a pseudo-Boolean function f a *unary negaform* if it can be represented as a negative combination of terms involving either only unnegated variables, or only negated ones. It is easy to show that unary negaforms are submodular, and in fact (5) and (7) provides a submodular quadratization for such functions.

Unary negaforms (more precisely, their negations) were considered by (Billionnet and Minoux 1985) and they showed that all cubic submodular functions can be represented by unary negaforms. They also provided a network flow model for the minimization of a unary negaform. The above (5), (7) submodular quadratization also leads to a network flow based minimization by the results of (Hammer 1965) and these two network flow models are of very similar size (though they are not identical). Thus, the above observations can be viewed as a new simple proof for the results of (Billionnet and Minoux 1985).

Let us remark finally that higher degree submodular functions cannot typically be represented as unary negaforms. This is implied e.g., by the results of (Zivny, Cohen, and Jeavons 2009a; 2009b) since we just proved that a unary negaform always has a submodular quadratization.

The trick that (5) can be extended by using negated variables was also observed by (Rother et al. 2009) (they called it type-II transformation). They also noticed that one can apply (5) to a subproduct (of a monomial), under some conditions. In particular, they quadratized separately the negated and unnegated variables in a monomial and derived a new transformation (called type-I):

$$-\prod_{j\in S_0}\overline{x}_j\prod_{j\in S_1} x_j = \min_{u,v\in\mathbb{B}} -uv + u\sum_{j\in S_0} x_j + v\sum_{j\in S_1}\overline{x}_j. \qquad (8)$$

No matter which variation from above we use to quadratize a degree d positive monomial, we always need at least $d-2$ new variables. In recent publications (Ishikawa 2009; 2011) provided a more compact quadratization for positive monomials, using only about half as many variables as the previous methods. To formulate this result, let is consider the positive term $t(x) = x_1x_2\cdots x_d$ of degree d, set $k = \lfloor\frac{d-1}{2}\rfloor$, an consider new binary variables $w = (w_1, w_2, ..., w_k)$. Define $S_1 = \sum_{j=1}^{d} x_j$, $S_2 = \sum_{1\le i<j\le d} x_ix_j$, $A = \sum_{j=1}^{k} w_j$ and $B = \sum_{j=1}^{k}(4j-1)w_j$. Then the following equalities hold:

$$\prod_{j=1}^{d} x_j \;=\; S_2 + \min_{w\in\mathbb{B}^k} B - 2AS_1 \qquad (9)$$

if $d = 2k+2$, and

$$\prod_{j=1}^{d} x_j \;=\; S_2 + \min_{w\in\mathbb{B}^k} B - 2AS_1 + w_k\,(S_1 - d + 1) \qquad (10)$$

if $d = 2k+1$.

Let us note that S_1 and S_2 are symmetric functions of x, while A is a symmetric function of w. However, B is not a symmetric function of w. It is an interesting question on its own if one could find a quadratization of $t(x)$ which is symmetric in both x and w, and needs substantially fewer new variables than d.

Let us also note that while this method introduces substantially fewer variables than the previous methods, it also introduces $\binom{d}{2}$ positive quadratic terms which makes the resulting quadratization highly non-submodular. Despite of this negative feature, (Ishikawa 2011) reported very good computational results, in particular when compared to the quadratization of (Rosenberg 1975).

Let us conclude this section by pointing out that all of the above methods, except (Rosenberg 1975) introduce individual new variables for each of the monomials of f. In many applications this is a disadvantage, increasing the size and frequently the level of non-submodularity of the resulting quadratization.

We can also note that the quadratization of negative terms are quite well solved by (5), since we need only one new variable (per term) and the resulting quadratic form is submodular.

In the sequel we present some new quadratization techniques, and for the above reasons, we focus primarily on positive terms and/or on the issue of using fewer than one per term new variables.

Multiple Splits of Terms

We show first a generalization of some of the above results. We still focus on a single term, and introduce a general scheme to split this term into several fragments in order to decrease the maximum degree.

Let p, q be positive integers, and denote by $[q] = \{1, 2, ..., q\}$ the set of positive integers up to q. Assume that $\phi_i : \mathbb{B}^p \to \mathbb{B}$ are Boolean functions for $i \in [q]$ satisfying the following conditions:

$$\begin{aligned} &\min_{y\in\mathbb{B}^p} \sum_{i=1}^{q} \phi_i(y) \;=\; 1, \quad \text{and} \\ &\forall I \subseteq [q],\; I \neq [q],\; \exists y_I \in \mathbb{B}^p \;\; \text{s.t.} \quad \sum_{i\in I} \phi_i(y_I) \;=\; 0. \end{aligned} \tag{11}$$

In other words, the sum of the ϕ functions have a positive minimum, but if we leave out any of the summands, the minimum becomes zero. For instance, for $p = 2$, $q = 3$ the functions $\phi_1 = y_1$, $\phi_2 = y_2$ and $\phi_3 = \overline{y}_1\overline{y}_2$ form such a set.

Theorem 1 *Let ϕ_i be Boolean functions satisfying condition* (11)*, and $P_i \subseteq [d]$ be subsets for $i \in [q]$ covering $[d]$. Then we have*

$$\prod_{j=1}^{d} x_j \;=\; \min_{y\in\mathbb{B}^p} \sum_{i=1}^{q} \phi_i(y) \prod_{j\in P_i} x_j. \tag{12}$$

Proof If $\prod_{j\in P_i} x_j = 1$ for all $i \in [q]$ then we have

$$1 \;=\; \min_{y\in\mathbb{B}^p} \sum_{i=1}^{q} \phi_i(y) \;=\; 1$$

by (11). If there is an index $k \in [q]$ for which $\prod_{j\in P_k} x_j = 0$, then by (11) there exists a $y^* \in \mathbb{B}^p$ such that $\phi_i(y^*) = 0$ for all $i \neq k$, and consequently we have $\phi_k(y^*) = 1$. Thus,

$$\begin{aligned} 0 \;&\le \min_{y\in\mathbb{B}^p} \sum_{i=1}^{q} \phi_i(y) \prod_{j\in P_i} x_j \\ &\le \sum_{i=1}^{q} \phi_i(y^*) \prod_{j\in P_i} x_j \;=\; \prod_{j\in P_k} x_j \;=\; 0 \end{aligned}$$

follows, proving the claim. □

Remarks and Examples:

- $\phi_1 = y_1$ and $\phi_2 = \overline{y}_1 = 1 - y_1$ provides a 2-split;
- $\phi_1 = y_1$, $\phi_2 = y_2$, and $\phi_3 = \overline{y}_1\overline{y}_2$ provides a 3-split;
- any binary tree of depth p with q leaves defines an appropriate system of ϕ_i functions, however not all systems correspond to such a tree (see e.g., the above 3-split);
- p variables can in general provide a $q \le 2^p$-split transforming a degree d term to q terms of maximum degree $p + \lceil \frac{d}{q} \rceil$.
- a 2-split combined with (5) yields the following quadratization of a cubic term
$$\begin{aligned} xyz &= \min_{u\in\mathbb{B}} xu + \overline{u}yz \\ &= \min_{u\in\mathbb{B}} xu + yz - uyz \\ &= \min_{u,v\in\mathbb{B}} xu + yz + (2 - y - u - z)v; \end{aligned}$$
- combining the above with a 2-split yields the following quadratization of a quartic term $txyz = \min_{u\in\mathbb{B}} txu + \overline{u}yz = \min_{u,v,w,s\in\mathbb{B}} tv + xu + (2-v-x-u)w + yz + (2-u-y-z)s$; another way doing this is $txyz = \min_{u\in\mathbb{B}} tu + xyz - uxyz = \min_{u,v,w,s\in\mathbb{B}} tu + xv + yz + (2 - v - y - z)w + (3 - u - x - y - z)s$.

It is interesting that in all of the above attempts to quadratize a positive degree d term, we had to include at least $d-1$ positive quadratic terms. We in fact conjecture that this is necessary.

Splitting of Common Parts

In this section first we still focus on positive terms, and introduce a quadratization which associates a single new variable with several terms, achieving a simultaneous decrease in their degrees.

Theorem 2 *Let $C \subseteq [n]$, $\mathcal{H} \subseteq 2^{[n]\setminus C}$, and consider a fragment of a pseudo-Boolean function of the form*

$$\phi \;=\; \sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in H\cup C} x_j,$$

where $\alpha_H \ge 0$ for all $H \in \mathcal{H}$. Then we have

$$\phi \;=\; \min_{w\in\mathbb{B}} \left(\sum_{H\in\mathcal{H}} \alpha_H \right) \overline{w} \prod_{j\in C} x_j + \sum_{H\in\mathcal{H}} \alpha_H \, w \prod_{j\in H} x_j. \tag{13}$$

Proof We claim that $w = \prod_{j\in C} x_j$ at a minimum, in which case the left and right hand sides are identical. To see this claim, observe that we have the inequalities

$$\phi \le \sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in H} x_j,$$

and

$$\phi \le \sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in C} x_j.$$

Furthermore, these two right hand sides are the values of the right hand side of (13) corresponding to $w = 1$ and $w = 0$, respectively. Thus, $w = \prod_{j\in C} x_j$ indeed achieves a value not larger than any of those. $\square$

We extend the idea of splitting away common parts with a single new variable top negative terms, as well.

Theorem 3 *Let $C \subseteq [n]$, $\mathcal{H} \subseteq 2^{[n]\setminus C}$, and consider a fragment of a pseudo-Boolean function of the form*

$$\phi \;=\; -\sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in H\cup C} x_j,$$

where $\alpha_H \geq 0$ for all $H \in \mathcal{H}$. Then we have

$$\phi \;=\; \min_{w\in\mathbb{B}} \sum_{H\in\mathcal{H}} \alpha_H\, w \left(1 - \prod_{j\in C} x_j - \prod_{j\in H} x_j\right). \tag{14}$$

Proof We can prove, similarly to the previous proof that $w = \prod_{j\in C} x_j$ at a minimum. For this let us note that if $w = \prod_{j\in C} x_j$, then the right hand side in (14) is identical with ϕ, since $\prod_{j\in C} x_j \left(1 - \prod_{j\in C} x_j\right) = 0$ for all assignments $x \in \mathbb{B}^n$. Furthermore, we have the inequalities

$$\phi \;\leq\; 0,$$

and

$$\phi \;\leq\; \sum_{H\in\mathcal{H}} \alpha_H \left(1 - \prod_{j\in C} x_j - \prod_{j\in H} x_j\right),$$

where the right hand side values are the right hand side values of (14) corresponding to $w = 0$ and $w = 1$, respectively. Thus, again $w = \prod_{j\in C} x_j$ achieves the smallest possible value. $\square$

Conclusions

In this paper we proposed new quadratization techniques, possibly decreasing the number of new variables needed, when compared to earlier, term-wise quadratization techniques, without introducing "large" coefficients and/or unnecessarily many non-submodular terms.

In fact, using Theorem 2 recursively, we can find a quadratization $g(x, w)$ of any function f (even if f is already quadratic) in which we have at most $n - 1$ positive quadratic terms, where n is the number of variables of f. This shows that the difficulty of minimizing f is not coming from the excessive number of non-submodular terms. In fact if we restrict our input to quadratic pseudo-Boolean functions in n variables and with at most $n-1$ positive terms, the minimization problems remains as hard as general quadratic minimization.

On the positive side a recent publication (Fix et al. 2011) demonstrates that for a large class of binary optimization problems arising from computer vision problems quadratization based primarily on Theorem 2 is very effective. When compared to the recent results of (Ishikawa 2011) we obtained quadratizations with substantially fewer new variables and positive quadratic terms. Applying e.g., the polynomial preprocessing algorithm QPBO we managed to run faster and fix substantially more variables at their optimum values than in (Ishikawa 2011).

There are several interesting open ends. The most basic one perhaps is a better and more complete understanding of quadratization techniques. Note that in the above results, we did not plan to restrict the number of new variables. It just happened in each case that the resulting quadratized function involved only polynomially many new variables. Is this always the case? Can we get better/easier to minimize quadratizations in terms of exponentially many new variables?

Which submodular functions have submodular quadratization? How to recognize those? How to find efficiently such a quadratization?

References

Billionnet, A., and Minoux, M. 1985. Maximizing a supermodular pseudoboolean function: A polynomial algorithm for supermodular cubic functions. *Discrete Applied Mathematics* 12(1):1 – 11.

Boros, E., and Hammer, P. L. 1989. A max-flow approach to improved roof-duality in quadratic $0-1$ minimization. Technical Report RRR 15-1989, RUTCOR, Rutgers University.

Boros, E., and Hammer, P. L. 2002. Pseudo-boolean optimization. *Discrete Applied Mathematics* 123(1-3):155–225.

Boros, E.; Hammer, P. L.; Sun, R.; and Tavares, G. 2008. A max-flow approach to improved lower bounds for quadratic unconstrained binary optimization (qubo). *Discrete Optimization* 5(2):501–529.

Buchheim, C., and Rinaldi, G. 2007. Efficient reduction of polynomial zero-one optimization to the quadratic case. *SIAM Journal on Optimization* 18(4):1398–1413.

Fix, A.; Gruber, A.; Boros, E.; and Zabih, R. 2011. A graph cut algorithm for higher-order markov random fields. In *ICCV*.

Freedman, D., and Drineas, P. 2005. Energy minimization via graph cuts: Settling what is possible. In *CVPR (2)*, 939–946. IEEE Computer Society.

Gallo, G., and Simeone, B. 1989. On the supermodular knapsack problem. *Mathematical Programming* 45:295–309.

Grötschel, M.; Lovász, L.; and Schrijver, A. 1981. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica* 1(2):169–197.

Hammer, P., and Rudeanu, S. 1968. *Boolean Methods in Operations Research and Related Areas*. Springer Verlag.

Hammer, P.; Hansen, P.; and Simeone, B. 1984. Roof duality, complementation and persistency in quadratic $0-1$ optimization. *Mathematical Programming* 28:121–155.

Hammer, P. L. 1965. Some network flow problems solved with pseudo-boolean programming. *Operations Research* 13(3):388–399.

Hansen, P. 1979. Methods of nonlinear $0-1$ programming. *Annals of Discrete Mathematics* 5:53–70.

Ishikawa, H. 2009. Higher-order gradient descent by fusion-move graph cut. In *ICCV*, 568–574. IEEE.

Ishikawa, H. 2011. Transformation of general binary mrf minimization to the first-order case. *IEEE Trans. Pattern Anal. Mach. Intell.* 33(6):1234–1249.

Iwata, S.; Fleischer, L.; and Fujishige, S. 2000. A combinatorial, strongly polynomial-time algorithm for minimizing submodular functions. In *STOC*, 97–106.

Kolmogorov, V., and Zabih, R. 2004. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.* 26(2):147–159.

Ramalingam, S.; Russell, C.; Ladicky, L.; and Torr, P. H. S. 2011. Efficient minimization of higher order submodular functions using monotonic boolean functions. *CoRR* abs/1109.2304.

Rosenberg, I. 1975. Reduction of bivalent maximization to the quadratic case. Technical report, Centre d'Etudes de Recherche Operationnelle.

Roth, S., and Black, M. J. 2009. Fields of experts. *International Journal of Computer Vision* 82(2):205–229.

Rother, C.; Kolmogorov, V.; Lempitsky, V. S.; and Szummer, M. 2007. Optimizing binary mrfs via extended roof duality. In *CVPR*. IEEE Computer Society.

Rother, C.; Kohli, P.; Feng, W.; and Jia, J. 2009. Minimizing sparse higher order energy functions of discrete variables. In *CVPR*, 1382–1389. IEEE.

Schrijver, A. 2000. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *J. Comb. Theory, Ser. B* 80(2):346–355.

Zivny, S., and Jeavons, P. G. 2010. Classes of submodular constraints expressible by graph cuts. *Constraints* 15(3):430–452.

Zivny, S.; Cohen, D. A.; and Jeavons, P. G. 2009a. The expressive power of binary submodular functions. *Discrete Applied Mathematics* 157(15):3347–3358.

Zivny, S.; Cohen, D. A.; and Jeavons, P. G. 2009b. The expressive power of binary submodular functions. In Królovic, R., and Niwinski, D., eds., *MFCS*, volume 5734 of *Lecture Notes in Computer Science*, 744–757. Springer.