Tightening linearizations of non-linear binary optimization problems

Elisabeth Rodríguez Heck and Yves Crama

QuantOM, HEC Management School, University of Liège

Partially supported by Belspo - IAP Project COMEX

January 2016, Louvain-la-Neuve, ORBEL 30

Definitions

Definition: Pseudo-Boolean functions

A pseudo-Boolean function is a mapping $f : \{0,1\}^n \to \mathbb{R}$.

Multilinear representation

Every pseudo-Boolean function f can be represented uniquely by a multilinear polynomial (Hammer, Rosenberg, Rudeanu [5]).

Example:

$$f(x_1, x_2, x_3) = 9x_1x_2x_3 + 8x_1x_2 - 6x_2x_3 + x_1 - 2x_2 + x_3$$

Pseudo-Boolean Optimization

Many problems formulated as optimization of a pseudo-Boolean function

Pseudo-Boolean Optimization

$$\min_{x \in \{0,1\}^n} f(x)$$

- **Optimization is $\mathcal{NP}$-hard**, even if f is quadratic (MAX-2-SAT, MAX-CUT modelled by quadratic f).
- Approaches:
  - **Linearization**: extensive literature in integer programming.
  - Quadratization: exact algorithms, heuristics, polyhedral results...
  - Direct resolution methods

Pseudo-Boolean Optimization

Many problems formulated as optimization of a pseudo-Boolean function

Pseudo-Boolean Optimization

$$\min_{x \in \{0,1\}^n} f(x)$$

- **Optimization is $\mathcal{NP}$-hard**, even if f is quadratic (MAX-2-SAT, MAX-CUT modelled by quadratic f).
- Approaches:
    - **Linearization**: extensive literature in integer programming.
    - Quadratization: exact algorithms, heuristics, polyhedral results...
    - Direct resolution methods

Standard linearization (SL)

$$\min_{\{0,1\}^n} \sum_{S\in\mathcal{S}} a_S \prod_{k\in S} x_k + \sum_{i=1}^{n} a_i x_i,$$

$\mathcal{S} = \{S \subseteq \{1,\ldots,n\} \mid a_S \neq 0 \text{ and } |S| \geq 2\}$ (non-constant and non-linear monomials)

1. Substitute monomials

$$\begin{aligned} \min & \sum_{S\in\mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\ \text{s.t. } & y_S = \prod_{k\in S} x_k, && \forall S \in \mathcal{S} \\ & y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\ & x_k \in \{0,1\}, && \forall k = 1,\ldots,n \end{aligned}$$

Standard linearization (SL)

$$\min_{\{0,1\}^n} \sum_{S \in \mathcal{S}} a_S \prod_{k \in S} x_k + \sum_{i=1}^{n} a_i x_i,$$

$\mathcal{S} = \{S \subseteq \{1, \ldots, n\} \mid a_S \neq 0 \text{ and } |S| \geq 2\}$ (non-constant and non-linear monomials)

1. Substitute monomials

$$\begin{aligned} \min \; & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\ \text{s.t.} \; & y_S = \prod_{k \in S} x_k, && \forall S \in \mathcal{S} \\ & y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\ & x_k \in \{0,1\}, && \forall k = 1, \ldots, n \end{aligned}$$

2. Linearize constraints

$$\begin{aligned} \min \; & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\ \text{s.t.} \; & y_S \leq x_k, && \forall k \in S, \forall S \in \mathcal{S} \\ & y_S \geq \sum_{k \in S} x_k - (|S| - 1), && \forall S \in \mathcal{S} \\ & y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\ & x_k \in \{0,1\}, && \forall k = 1, \ldots, n \end{aligned}$$

Standard linearization (SL)

$$\min_{\{0,1\}^n} \sum_{S \in \mathcal{S}} a_S \prod_{k \in S} x_k + \sum_{i=1}^{n} a_i x_i,$$

$\mathcal{S} = \{S \subseteq \{1, \dots, n\} \mid a_S \neq 0 \text{ and } |S| \geq 2\}$ (non-constant and non-linear monomials)

1. Substitute monomials

$$\begin{aligned}
\min \; & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\
\text{s.t. } \; & y_S = \prod_{k \in S} x_k, && \forall S \in \mathcal{S} \\
& y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\
& x_k \in \{0,1\}, && \forall k = 1, \dots, n
\end{aligned}$$

2. Linearize constraints

$$\begin{aligned}
\min \; & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\
\text{s.t. } \; & y_S \leq x_k, && \forall k \in S, \forall S \in \mathcal{S} \\
& y_S \geq \sum_{k \in S} x_k - (|S| - 1), && \forall S \in \mathcal{S} \\
& y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\
& x_k \in \{0,1\}, && \forall k = 1, \dots, n
\end{aligned}$$

Standard linearization (SL)

$$\min_{\{0,1\}^n} \sum_{S \in \mathcal{S}} a_S \prod_{k \in S} x_k + \sum_{i=1}^{n} a_i x_i,$$

$\mathcal{S} = \{S \subseteq \{1, \dots, n\} \mid a_S \neq 0 \text{ and } |S| \geq 2\}$ (non-constant and non-linear monomials)

1. Substitute monomials

$$\begin{aligned} \min \ & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\ \text{s.t. } & y_S = \prod_{k \in S} x_k, && \forall S \in \mathcal{S} \\ & y_S \in \{0,1\}, && \forall S \in \mathcal{S} \\ & x_k \in \{0,1\}, && \forall k = 1, \dots, n \end{aligned}$$

3. Linear relaxation

$$\begin{aligned} \min \ & \sum_{S \in \mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i \\ \text{s.t. } & y_S \leq x_k, && \forall k \in S, \forall S \in \mathcal{S} \\ & y_S \geq \sum_{k \in S} x_k - (|S| - 1), && \forall S \in \mathcal{S} \\ & 0 \leq y_S \leq 1, && \forall S \in \mathcal{S} \\ & 0 \leq x_k \leq 1, && \forall k = 1, \dots, n \end{aligned}$$

Linearized problem

Linearized problem

$$\min_{(x,y)\in P^*_{SL}} \sum_{S\in\mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i, \text{ where}$$

$$P^*_{SL} = conv(\{(x,y) \in \{\mathbf{0},\mathbf{1}\}^{n+|\mathcal{S}|} \mid y_S \le x_k \ \ \forall k \in S, y_S \ge \sum_{k\in S} x_k - (|S|-1)\})$$

Relaxing integrality constraints we obtain the **standard linearization polytope**

$$P_{SL} = conv(\{(x,y) \in [\mathbf{0},\mathbf{1}]^{n+|\mathcal{S}|} \mid y_S \le x_k \ \ \forall k \in S, y_S \ge \sum_{k\in S} x_k - (|S|-1)\})$$

Linearized problem

Linearized problem

$$\min_{(x,y)\in P^*_{SL}} \sum_{S\in\mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i, \text{ where}$$

$$P^*_{SL} = conv(\{(x,y) \in \{\mathbf{0},\mathbf{1}\}^{n+|\mathcal{S}|} \mid y_S \le x_k \;\; \forall k \in S, y_S \ge \sum_{k\in S} x_k - (|S|-1)\})$$

Relaxing integrality constraints we obtain the **standard linearization polytope**

$$P_{SL} = conv(\{(x,y) \in [\mathbf{0},\mathbf{1}]^{n+|\mathcal{S}|} \mid y_S \le x_k \;\; \forall k \in S, y_S \ge \sum_{k\in S} x_k - (|S|-1)\})$$

For a function containing a single non-linear monomial, $P^*_{SL} = P_{SL}$

Linearized problem

Linearized problem

$$\min_{(x,y)\in P^*_{SL}} \sum_{S\in\mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i, \text{ where}$$

$$P^*_{SL} = conv(\{(x,y) \in \{\mathbf{0},\mathbf{1}\}^{n+|\mathcal{S}|} \mid y_S \leq x_k \ \ \forall k \in S, y_S \geq \sum_{k\in S} x_k - (|S|-1)\})$$

Relaxing integrality constraints we obtain the **standard linearization polytope**

$$P_{SL} = conv(\{(x,y) \in [\mathbf{0},\mathbf{1}]^{n+|\mathcal{S}|} \mid y_S \leq x_k \ \ \forall k \in S, y_S \geq \sum_{k\in S} x_k - (|S|-1)\})$$

For a function containing a single non-linear monomial, $P^*_{SL} = P_{SL}$

Linearized problem

Linearized problem

$$\min_{(x,y)\in P^*_{SL}} \sum_{S\in\mathcal{S}} a_S y_S + \sum_{i=1}^{n} a_i x_i, \text{ where}$$

$$P^*_{SL} = conv(\{(x,y) \in \{\mathbf{0},\mathbf{1}\}^{n+|\mathcal{S}|} \mid y_S \leq x_k \ \ \forall k \in S, y_S \geq \sum_{k\in S} x_k - (|S|-1)\})$$

Relaxing integrality constraints we obtain the **standard linearization polytope**

$$P_{SL} = conv(\{(x,y) \in [\mathbf{0},\mathbf{1}]^{n+|\mathcal{S}|} \mid y_S \leq x_k \ \ \forall k \in S, y_S \geq \sum_{k\in S} x_k - (|S|-1)\})$$

Not for two non-linear terms, in general P_{SL} is a very weak relaxation!!!

Definition of the 2-links

Definition: 2-links

Consider two monomials $S, T \in \mathcal{S}$ such that $|S \cap T| \geq 2$, and their corresponding variables y_S, y_T. The 2-link between S and T is

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S|$$

- For $|S \cap T| < 2$, the 2-links are implied by the standard linearization.

Definition of the 2-links

Definition: 2-links

Consider two monomials $S, T \in \mathcal{S}$ such that $|S \cap T| \geq 2$, and their corresponding variables y_S, y_T. The 2-link between S and T is

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S|$$

- For $|S \cap T| < 2$, the 2-links are implied by the standard linearization.
- Interpretation: if $y_S = \prod_{i \in S} x_i = 1$ and $y_T = \prod_{i \in T} x_i = 0$, a variable in $T \setminus S$ must be zero.

Definition of the 2-links

Definition: 2-links

Consider two monomials $S, T \in \mathcal{S}$ such that $|S \cap T| \geq 2$, and their corresponding variables y_S, y_T. The 2-link between S and T is

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S|$$

- For $|S \cap T| < 2$, the 2-links are implied by the standard linearization.
- Interpretation: if $y_S = \prod_{i \in S} x_i = 1$ and $y_T = \prod_{i \in T} x_i = 0$, a variable in $T \setminus S$ must be zero.

Validity and strength

Proposition: Validity

For any $S, T \in \mathcal{S}$, the corresponding 2-link is valid for P^*_{SL}.

Proposition: Facet-defining for nested monomials

Consider a pseudo-Boolean function defined on l monomials such that $S^{(1)} \subseteq S^{(2)} \subseteq \cdots \subseteq S^{(l)}$ and $|S^{(1)}| \geq 2$. Then, the 2-links corresponding to consecutive monomials in the nest

$$y_{S^{(k)}} \leq y_{S^{(k+1)}} - \sum_{i \in S^{(k+1)} \setminus S^{(k)}} x_i + |S^{(k+1)} \setminus S^{(k)}|$$

$$y_{S^{(k+1)}} \leq y_{S^{(k)}},$$

for $k = 1, \ldots, l-1$, are facet-defining for $P^{*,nest}_{SL}$.

Validity and strength

Proposition: Validity

For any $S, T \in \mathcal{S}$, the corresponding 2-link is valid for P^*_{SL}.

Proposition: Facet-defining for nested monomials

Consider a pseudo-Boolean function defined on l monomials such that $S^{(1)} \subseteq S^{(2)} \subseteq \cdots \subseteq S^{(l)}$ and $|S^{(1)}| \geq 2$. Then, the 2-links corresponding to consecutive monomials in the nest

$$y_{S^{(k)}} \leq y_{S^{(k+1)}} - \sum_{i \in S^{(k+1)} \setminus S^{(k)}} x_i + |S^{(k+1)} \setminus S^{(k)}|$$

$$y_{S^{(k+1)}} \leq y_{S^{(k)}},$$

for $k = 1, \ldots, l-1$, are facet-defining for $P^{*,nest}_{SL}$.

The case of two non-linear monomials

Consider f containing **exactly two non-linear terms** S, T ($|S \cap T| \geq 2$), and the corresponding 2-links

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S| \tag{1}$$

$$y_T \leq y_S - \sum_{i \in S \setminus T} x_i + |S \setminus T| \tag{2}$$

Proposition: Facet-defining

The 2-links (1) and (2) are facet-defining for P^*_{SL}.

The case of two non-linear monomials

Consider f containing **exactly two non-linear terms** S, T ($|S \cap T| \geq 2$), and the corresponding 2-links

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S| \tag{1}$$

$$y_T \leq y_S - \sum_{i \in S \setminus T} x_i + |S \setminus T| \tag{2}$$

Proposition: Facet-defining

The 2-links (1) and (2) are facet-defining for P^*_{SL}.

Notation

$$P^{2links}_{SL} = P_{SL} \cap \{(x, y_S, y_T) \text{ such that } (1), (2) \text{ are satisfied}\}$$

The case of two non-linear monomials

Consider f containing **exactly two non-linear terms** S, T ($|S \cap T| \geq 2$), and the corresponding 2-links

$$y_S \leq y_T - \sum_{i \in T \setminus S} x_i + |T \setminus S| \quad (1)$$

$$y_T \leq y_S - \sum_{i \in S \setminus T} x_i + |S \setminus T| \quad (2)$$

Proposition: Facet-defining

The 2-links (1) and (2) are facet-defining for P^*_{SL}.

Notation

$$P_{SL}^{2links} = P_{SL} \cap \{(x, y_S, y_T) \text{ such that } (1), (2) \text{ are satisfied}\}$$

The case of two non-linear monomials

Theorem: Perfect formulation for two intersecting non-linear monomials

$$P_{SL}^{2links} = P_{SL}^{*}$$

Idea of the proof:

The case of two non-linear monomials

Theorem: Perfect formulation for two intersecting non-linear monomials

$$P_{SL}^{2links} = P_{SL}^{*}$$

Idea of the proof:

- Consider the four polytopes obtained by fixing y_S, y_T to 0 or to 1 in P_{SL}^{2links}:
  - P_{00}: $y_S = 0, y_T = 0$
  - P_{11}: $y_S = 1, y_T = 1$
  - P_{10}: $y_S = 1, y_T = 0$
  - P_{01}: $y_S = 0, y_T = 1$

The case of two non-linear monomials

Theorem: Perfect formulation for two intersecting non-linear monomials

$$P_{SL}^{2links} = P_{SL}^{*}$$

Idea of the proof:

- Consider the four polytopes obtained by fixing y_S, y_T to 0 or to 1 in P_{SL}^{2links}:
  - P_{00}: $y_S = 0, y_T = 0$
  - P_{11}: $y_S = 1, y_T = 1$
  - P_{10}: $y_S = 1, y_T = 0$
  - P_{01}: $y_S = 0, y_T = 1$
- Observe that they all have integer vertices.

The case of two non-linear monomials

Theorem: Perfect formulation for two intersecting non-linear monomials

$$P_{SL}^{2links} = P_{SL}^{*}$$

Idea of the proof:

- Consider the four polytopes obtained by fixing y_S, y_T to 0 or to 1 in P_{SL}^{2links}:
  - P_{00}: $y_S = 0, y_T = 0$
  - P_{11}: $y_S = 1, y_T = 1$
  - P_{10}: $y_S = 1, y_T = 0$
  - P_{01}: $y_S = 0, y_T = 1$
- Observe that they all have integer vertices.
- See that $P_{SL}^{2links} = conv(P_{00} \cup P_{11} \cup P_{10} \cup P_{01})$.

The case of two non-linear monomials

Theorem: Perfect formulation for two intersecting non-linear monomials

$$P_{SL}^{2links} = P_{SL}^{*}$$

Idea of the proof:

- Consider the four polytopes obtained by fixing y_S, y_T to 0 or to 1 in P_{SL}^{2links}:
  - P_{00}: $y_S = 0, y_T = 0$
  - P_{11}: $y_S = 1, y_T = 1$
  - P_{10}: $y_S = 1, y_T = 0$
  - P_{01}: $y_S = 0, y_T = 1$
- Observe that they all have integer vertices.
- See that $P_{SL}^{2links} = conv(P_{00} \cup P_{11} \cup P_{10} \cup P_{01})$.

Computational experiments: Motivation

- Example of function containing 3 non-linear monomials for which optimizing over P_{SL}^{2links} leads to a fractional solution:

$$f(x) = 5x_1x_2x_4 - 3x_1x_3x_4 - 3x_1x_2x_3 + 2x_3$$

- It is hopeless (unless $\mathcal{P} = \mathcal{NP}$) to find a concise perfect formulation for the general case.

Computational experiments: Motivation

- Example of function containing 3 non-linear monomials for which optimizing over P_{SL}^{2links} leads to a fractional solution:

$$f(x) = 5x_1x_2x_4 - 3x_1x_3x_4 - 3x_1x_2x_3 + 2x_3$$

- It is hopeless (unless $\mathcal{P} = \mathcal{NP}$) to find a concise perfect formulation for the general case.
- Are the 2-links still useful for the general case?

Computational experiments: Motivation

- Example of function containing 3 non-linear monomials for which optimizing over P_{SL}^{2links} leads to a fractional solution:

$$f(x) = 5x_1x_2x_4 - 3x_1x_3x_4 - 3x_1x_2x_3 + 2x_3$$

- It is hopeless (unless $\mathcal{P} = \mathcal{NP}$) to find a concise perfect formulation for the general case.
- Are the 2-links still useful for the general case?

Computational experiments: Objectives

Two objectives of the computational experiments:

- **Quality of the bounds**: of P_{SL} and P_{SL}^{2links}.
- **Computational performance**: of exact resolution methods with different types of cuts

Method name	CPLEX cuts	2-links (User cuts)
No cuts (P_{SL})	✗	✗
User cuts (P_{SL}^{2links})	✗	✓
CPLEX cuts	✓	✗
CPLEX & user cuts	✓	✓

Computational experiments: Objectives

Two objectives of the computational experiments:

- **Quality of the bounds**: of P_{SL} and P_{SL}^{2links}.
- **Computational performance**: of exact resolution methods with different types of cuts

Method name	**CPLEX cuts**	**2-links (User cuts)**
No cuts (P_{SL})	✗	✗
User cuts (P_{SL}^{2links})	✗	✓
CPLEX cuts	✓	✗
CPLEX & user cuts	✓	✓

Random instances: Generation

- Same generation procedure as Buchheim, Rinaldi ([1]).
- Input: n (variables), m (monomials).

Random instances: Generation

- Same generation procedure as Buchheim, Rinaldi ([1]).
- Input: n (variables), m (monomials).
- Monomials are always generated by randomly choosing the variables in it and the coefficient.

Random instances: Generation

- Same generation procedure as Buchheim, Rinaldi ([1]).
- Input: n (variables), m (monomials).
- Monomials are always generated by randomly choosing the variables in it and the coefficient.
  1. **Same-degree**. All monomials have the same degree $d = 3$ or 4.

Random instances: Generation

- Same generation procedure as Buchheim, Rinaldi ([1]).
- Input: n (variables), m (monomials).
- Monomials are always generated by randomly choosing the variables in it and the coefficient.
  1. **Same-degree**. All monomials have the same degree $d = 3$ or 4.
  2. **Random-degree**. The degree d of each monomial is chosen with probability 2^{d-1} (less high-degree monomials, more low-degree monomials).

Random instances: Generation

- Same generation procedure as Buchheim, Rinaldi ([1]).
- Input: n (variables), m (monomials).
- Monomials are always generated by randomly choosing the variables in it and the coefficient.
    1. **Same-degree**. All monomials have the same degree $d = 3$ or 4.
    2. **Random-degree**. The degree d of each monomial is chosen with probability 2^{d-1} (less high-degree monomials, more low-degree monomials).

Random instances: Results

Instance			LP gap (%)		IP execution times (secs)			
d	n	m	P_{SL}	P_{SL}^{2links}	no cuts	user	cplex	c & u
3	400	800	4.51	3.49	3.65	**2.57**	7.46	6.68
3	400	900	9.31	7.93	502.41	243.58	104.52	**87.75**
3	400	1000	14.77	13.13	841.36	**434.76**	1334.96	1884.21
3	600	1100	2.78	2.32	14.09	**9.88**	16.07	14.52
3	600	1200	6.06	5.37	645.16	333.94	**197.13**	270.07
3	600	1300	10.17	9.15	>3600	>3600	**2157.84**	2234.61
4	400	550	4.37	3.26	36.97	17.10	14.76	**11.6**
4	400	600	8.15	5.91	58.79	**13.86**	63.1	20.19
4	400	650	10.22	7.72	177.74	681.06	**348.79**	514.13
4	400	700	12.25	8.92	1343.18	1179.95	602.68	**329.05**
4	600	750	1.54	1.28	3.42	**3.05**	6.15	5.89
4	600	800	2.59	2.14	16.54	**12.08**	18.37	15.5
4	600	850	5.20	4.02	475.43	359.65	664.29	**316.73**
4	600	900	9.38	7.59	103.49	**42.29**	1526.84	1475.3

Table: Results for random (same-degree) instances

Random instances: Results

Instance			LP gap (%)		IP execution times (secs)			
d	n	m	P_{SL}	P_{SL}^{2links}	no cuts	user	cplex	c & u
12.6	200	600	12.21	10.15	10.42	8.08	7.15	**5.81**
11.2	200	700	12.73	10.73	78.72	30.12	34.74	**28.17**
11	200	800	18.99	16.10	748.15	254.81	118.55	**111.64**
13.6	200	900	27.29	23.72	889.37	**690.72**	1029.25	863.39
11.2	400	900	3.03	2.43	3.09	**1.72**	4.15	3.88
11	400	1000	3.50	2.82	19.56	**6.77**	8.87	8.44
11.4	400	1100	7.27	6.64	55.64	347.27	59.86	**53.66**
11.8	400	1200	7.04	6.45	256.80	**117.35**	254.46	147.80
13.8	600	1300	1.38	1.21	2.97	**2.53**	5.42	5.42
11.4	600	1400	3.86	3.57	294.03	238.87	**124.30**	135.38
12.2	600	1500	4.63	4.10	593.70	228.02	100.28	**86.36**
12.6	600	1600	5.00	4.53	1374.74	561.85	345.37	**280.95**

Table: Results for random (random-degree) instances

Vision instances: Idea

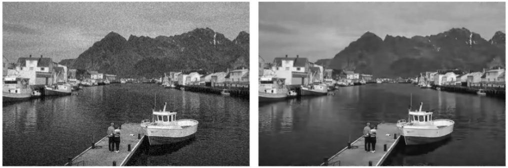

Figure: Image from "Corel database" with additive Gaussian noise [6].

Figure: Restoration of an old digitalized image with scratches [6].

Vision instances: Input generation

- **Base image**

1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(a) top left rect

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(b) centre rect

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(c) cross

Figure: Base images: size 10×10

- **Perturbation**: None, Low (change any pixel with probability 5%), High (change zero's with probability 50%).

Vision instances: Input generation

- **Base image**

1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(a) top left rect

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(b) centre rect

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(c) cross

Figure: Base images: size 10×10

- **Perturbation**: None, Low (change any pixel with probability 5%), High (change zero's with probability 50%).

Vision instances: Image restoration model

- **Input**: $p_{ij} \in \{0, 1\}$ value of pixel (i, j) in the input image.
- **Variables**: $x_{ij} \in \{0, 1\}$ value assigned to pixel (i, j) in the output.
- **Objective function**: $\min f(x) = L(x) + P(x)$

Vision instances: Image restoration model

- **Input**: $p_{ij} \in \{0, 1\}$ value of pixel (i, j) in the input image.
- **Variables**: $x_{ij} \in \{0, 1\}$ value assigned to pixel (i, j) in the output.
- **Objective function**: $\min f(x) = L(x) + P(x)$
  1. **Similarity** between input and output $L(x) = 25(p_{ij} - x_{ij})^2$ (linear).

Vision instances: Image restoration model

- **Input**: $p_{ij} \in \{0, 1\}$ value of pixel (i, j) in the input image.
- **Variables**: $x_{ij} \in \{0, 1\}$ value assigned to pixel (i, j) in the output.
- **Objective function**: $\min f(x) = L(x) + P(x)$
  1. **Similarity** between input and output $L(x) = 25(p_{ij} - x_{ij})^2$ (linear).
  2. **Smoothness** of the image (polynomial: 2×2 windows - degree 4).

Vision instances: Image restoration model

- **Input**: $p_{ij} \in \{0, 1\}$ value of pixel (i, j) in the input image.
- **Variables**: $x_{ij} \in \{0, 1\}$ value assigned to pixel (i, j) in the output.
- **Objective function**: $\min f(x) = L(x) + P(x)$
  1. **Similarity** between input and output $L(x) = 25(p_{ij} - x_{ij})^2$ (linear).
  2. **Smoothness** of the image (polynomial: 2×2 windows - degree 4).

Window assignments		Penalty
0 0 0 0	1 1 1 1	10
0 0 0 1	0 0 1 0	20
1 1 0 0	0 0 1 1	30
1 0 0 1	0 1 1 0	40

Vision instances: Image restoration model

- **Input**: $p_{ij} \in \{0, 1\}$ value of pixel (i, j) in the input image.
- **Variables**: $x_{ij} \in \{0, 1\}$ value assigned to pixel (i, j) in the output.
- **Objective function**: $\min f(x) = L(x) + P(x)$
  1. **Similarity** between input and output $L(x) = 25(p_{ij} - x_{ij})^2$ (linear).
  2. **Smoothness** of the image (polynomial: 2×2 windows - degree 4).

Window assignments		Penalty
0 0 0 0	1 1 1 1	10
0 0 0 1	0 0 1 0	20
1 1 0 0	0 0 1 1	30
1 0 0 1	0 1 1 0	40

Vision instances: Results

Instance (10 × 15)		LP gap (%)		IP execution times (secs)			
Base image	Perturbation	P_{SL}	$P_{SL}^{2\,links}$	no cuts	user	cplex	c & u
top left rect	none	621.80	318.05	> 3600	> 3600	6.22	**1.98**
top left rect	low	749.58	396.66	> 3600	> 3600	15.50	**2.04**
top left rect	high	480.87	251.87	> 3600	> 3600	38.49	**3.35**
centre rect	none	859.13	458.65	> 3600	> 3600	7.94	**2.04**
centre rect	low	1015.13	552.04	> 3600	> 3600	15.74	**2.59**
centre rect	high	464.31	242.59	> 3600	> 3600	49.42	**3.11**
cross	none	1608.33	883.33	> 3600	> 3600	32.37	**2.26**
cross	low	1790.63	999.23	> 3600	> 3600	20.78	**2.54**
cross	high	468.24	245.07	> 3600	> 3600	38.22	**3.46**

Table: Results for vision instances of size 10×15, $n = 150$ $m = 1033$

Vision instances: Results

Instance (15×15)		LP gap (%)		IP execution times (secs)			
Base image	Perturbation	P_{SL}	$P_{SL}^{2\,links}$	no cuts	user	cplex	c & u
top left rect	none	660.90	340.26	> 3600	> 3600	19.5	**3.49**
top left rect	low	714.29	374.27	> 3600	> 3600	28.06	**6.41**
top left rect	high	565.72	302.48	> 3600	> 3600	111.3	**12.86**
centre rect	none	698.13	366.75	> 3600	> 3600	30.12	**4.71**
centre rect	low	851.09	457.40	> 3600	> 3600	38.33	**8.44**
centre rect	high	483.33	253.69	> 3600	> 3600	97.17	**10.34**
cross	none	1284.52	698.57	> 3600	> 3600	16.54	**5.63**
cross	low	1457.22	801.10	> 3600	> 3600	22.30	**7.26**
cross	high	530.46	282.23	> 3600	> 3600	103.75	**11.02**

Table: Results for vision instances of size 15×15, $n = 225$ $m = 1598$

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:
    - Random instances: results depend on ratio $\frac{m}{n}$, adding 2-links helps.

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:
  - Random instances: results depend on ratio $\frac{m}{n}$, adding 2-links helps.
  - Vision instances: CPLEX cuts make the difference from infeasible to feasible. Adding 2-links always improves even more.

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:
  - Random instances: results depend on ratio $\frac{m}{n}$, adding 2-links helps.
  - Vision instances: CPLEX cuts make the difference from infeasible to feasible. Adding 2-links always improves even more.

Open questions:

- Other problem structures for which 2-links can help computationally? Provide a complete description? When is SL enough?

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:
  - Random instances: results depend on ratio $\frac{m}{n}$, adding 2-links helps.
  - Vision instances: CPLEX cuts make the difference from infeasible to feasible. Adding 2-links always improves even more.

Open questions:

- Other problem structures for which 2-links can help computationally? Provide a complete description? When is SL enough?
- Similar inequalities linking 3 terms?

Conclusions

Main contributions:

- Definition of the 2-links to strengthen P_{SL} linearization.
- For f containing two intersecting non-linear monomials, P_{SL}^{2links} is a complete description.
- Computational experiments for the general case:
  - Random instances: results depend on ratio $\frac{m}{n}$, adding 2-links helps.
  - Vision instances: CPLEX cuts make the difference from infeasible to feasible. Adding 2-links always improves even more.

Open questions:

- Other problem structures for which 2-links can help computationally? Provide a complete description? When is SL enough?
- Similar inequalities linking 3 terms?

Some references I

- C. Buchheim and G. Rinaldi. Efficient reduction of polynomial zero-one optimization to the quadratic case. SIAM Journal on Optimization, 18(4):1398–1413, 2007.
- Y. Crama and P. L. Hammer. Boolean functions: Theory, algorithms, and applications. Cambridge University Press, New York, N. Y., 2011.
- R. Fortet. L'algèbre de boole et ses applications en recherche opérationelle. Cahiers du Centre d'Études de recherche opérationelle, 4:5–36, 1959.
- F. Glover and E. Woolsey. Further reduction of zero-one polynomial programming problems to zero-one linear programming problems. Operations Research, 21(1):156–161, 1973.
- P. L. Hammer, I. Rosenberg, and S. Rudeanu. On the determination of the minima of pseudo-boolean functions. Studii si Cercetari Matematice, 14:359–364, 1963. in Romanian.

Some references II

S. Roth. High-Order Markov Random Fields for Low-Level Vision. PhD thesis, Brown University, Providence, Rhode Island, USA, 2007.