QUBO		Time	
00000000	00000000		

What is Quadratization?

Quadratization Techniques

# **Quadratization of Pseudo-Boolean Functions**

#### Endre Boros

# RUTCOR, Rutgers University

University of Primorska, November 19, 2012<sup>1</sup>

<sup>&</sup>lt;sup>1</sup> Joint work with A. Fix, A. Gruber, G. Tavares and R. Zabih  $\bigcirc$   $\land$   $\bigcirc$   $\land$   $\bigcirc$   $\land$   $\bigcirc$   $\land$   $\bigcirc$   $\land$   $\bigcirc$   $\land$   $\bigcirc$ 

QUBO		$\operatorname{Time}$	
• <b>0</b> 000000	00000000		

What is Quadratization?

Quadratization Techniques

・ロト ・ 四ト ・ 日ト ・ 日 ・

### Outline

# Quadratic Unconstrained Binary Optimization

## • Quadratic Pseudo-Boolean Functions

- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

# Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **B** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization
0000000			

#### Variables and Literals

- Variables:  $x_1, x_2, ..., x_n \in \{0, 1\}.$
- Negations:  $\overline{x}_i = 1 x_i \in \{0, 1\}$  for i = 1, ..., n
- Literals:  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$

### Quadratic Pseudo-Boolean Function (QPBF):

#### $f: \{0,1\}^n o \mathbb{R}$

うして ふゆう ふほう ふほう ふしつ

$$f(x_1, ..., x_n) = c_0 + \sum_{j=1}^n c_j x_j + \sum_{1 \le i < j \le n} c_{ij} x_i x_j$$

Quadratic Unconstrained Binary Optimization (QUBO)

 $\min_{(x_1,...,x_n)\in\{0,1\}^n} f(x_1,...,x_n)$ 

	TT		~ >
0000000	0000000	0000	0000000000
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techni

#### Variables and Literals

- Variables:  $x_1, x_2, ..., x_n \in \{0, 1\}$ . • Negations:  $\overline{x}_i = 1 - x_i \in \{0, 1\}$  for i = 1, ..., n
- Literals:  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$

#### Quadratic Pseudo-Boolean Function (QPBF):

$$f(x_1, ..., x_n) = c_0 + \sum_{j=1}^n c_j x_j + \sum_{1 \le i < j \le n} c_{ij} x_i x_j$$

Quadratic Unconstrained Binary Optimization (QUBO)

 $\min_{(x_1,...,x_n)\in\{0,1\}^n} f(x_1,...,x_n)$ 

		a	~ ``
0000000	0000000	0000	0000000000
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techn

#### Variables and Literals

- Variables:  $x_1, x_2, ..., x_n \in \{0, 1\}$ .
- Negations:  $\overline{x}_i = 1 x_i \in \{0, 1\}$  for i = 1, ..., n
- Literals:  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$

#### Quadratic Pseudo-Boolean Function (QPBF):

$$f(x_1, ..., x_n) = c_0 + \sum_{j=1}^n c_j x_j + \sum_{1 \le i < j \le n} c_{ij} x_i x_j$$

Quadratic Unconstrained Binary Optimization (QUBO)

 $\min_{(x_1,...,x_n)\in\{0,1\}^n} f(x_1,...,x_n)$ 

うして ふゆう ふほう ふほう ふしつ

	TT 1 TO .		$\sim$
0000000	0000000	0000	0000000000
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Technic

#### Variables and Literals

- Variables:  $x_1, x_2, ..., x_n \in \{0, 1\}$ .
- Negations:  $\overline{x}_i = 1 x_i \in \{0, 1\}$  for i = 1, ..., n
- Literals:  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$

## Quadratic Pseudo-Boolean Function (QPBF):

$$f(x_1, ..., x_n) = c_0 + \sum_{j=1}^n c_j x_j + \sum_{1 \le i < j \le n} c_{ij} x_i x_j$$

 $f: \{0,1\}^n \to \mathbb{R}$ 

うして ふゆう ふほう ふほう ふしつ

Quadratic Unconstrained Binary Optimization (QUBO)

 $\min_{(x_1,...,x_n)\in\{0,1\}^n} f(x_1,...,x_n)$ 

	TT		>>
0000000	0000000	0000	0000000000
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Technic

#### Variables and Literals

- Variables:  $x_1, x_2, ..., x_n \in \{0, 1\}$ .
- Negations:  $\overline{x}_i = 1 x_i \in \{0, 1\}$  for i = 1, ..., n
- Literals:  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$

## Quadratic Pseudo-Boolean Function (QPBF):

$$f(x_1, ..., x_n) = c_0 + \sum_{j=1}^n c_j x_j + \sum_{1 \le i < j \le n} c_{ij} x_i x_j$$

 $f: \{0,1\}^n \to \mathbb{R}$ 

うして ふゆう ふほう ふほう ふしつ

Quadratic Unconstrained Binary Optimization (QUBO)

$$\min_{(x_1,...,x_n)\in\{0,1\}^n} f(x_1,...,x_n)$$

QUBO
00000000

Polynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

## Outline

# Quadratic Unconstrained Binary Optimization

• Quadratic Pseudo-Boolean Functions

## • Representations and Bounds

- Origin of Graph Cut Models
- Network Model for General QUBO

## Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

## **3** What is Quadratization?

- Quadratization
- Submodular Functions

## Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

Represen	tations and Bounds		
00000000	0000000	0000	0000000000
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques



QUBO 00 <b>0</b> ●0000	Polynomial Time Preprocessing 00000000	What is Quadratization? 0000	Quadratization Techniques 00000000000
Represen	tations and Bounds		
D			1. 1 . 1

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のへぐ

**Posiforms**: Nonnegative (except maybe the constant terms) multi-linear polynomials in 2n literals  $x_1, \overline{x}_1, ..., x_n, \overline{x}_n$  $^{3}$ F

$$f = -2 - x_1 - x_2 - x_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 \qquad \text{QPB}$$

QUBO 00 <b>0</b> ●0000	Polynomial Time Preprocessing 00000000	What is Quadratization? 0000	Quadratization Techniques
Represe	entations and Bounds		
	<b>iforms</b> : Nonnegative (except mayb $n$ literals $x_1, \overline{x}_1,, x_n, \overline{x}_n$	e the constant terms) multi	-linear polynomials
f	$= -2 - x_1 - x_2 - x_3 + x_1 x_2 + = -5 + \overline{x}_1 + \overline{x}_2 + \overline{x}_3 + x_1 x_2 + $		posiform

QUBO 00000000	Polynomial Time Preprocessing 00000000	What is Quadratization? 0000	Quadratization Techniques 00000000000
Repres	entations and Bounds		
	<b>siforms</b> : Nonnegative (except mayb $2n$ literals $x_1, \overline{x}_1,, x_n, \overline{x}_n$	e the constant terms) multi-	-linear polynomials
f	$= -2 - x_1 - x_2 - x_3 + x_1 x_2 + = -5 + \overline{x}_1 + \overline{x}_2 + \overline{x}_3 + x_1 x_2 + = -4 + \overline{x}_3 + \overline{x}_1 \overline{x}_2 + x_1 x_3 + x_2 $	$x_1x_3 + x_2x_3$ quadratic p	

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ ∃ のへで

QUBC 000€0		Polynomial Time Preprocessing 00000000	What is Quadr 0000	atization? Quadratizat	ion Techniques
Rep	resent	ations and Bounds			
		<b>ms</b> : Nonnegative (except maybe erals $x_1, \overline{x}_1,, x_n, \overline{x}_n$	the constant to	rms) multi-linear polyno	mials
	=	$\begin{array}{c} -2-x_1-x_2-x_3+x_1x_2+x\\ -5+\overline{x}_1+\overline{x}_2+\overline{x}_3+x_1x_2+x\\ -4+\overline{x}_3+\overline{x}_1\overline{x}_2+x_1x_3+x_2x_3\\ -3+x_1x_2x_3+\overline{x}_1\overline{x}_2\overline{x}_3 \end{array}$	$x_1x_3 + x_2x_3$	QPBF quadratic posiform quadratic posiform cubic posiform	

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 の�?

QUBO 00 <b>0</b> ●0000	Polynomial Time Preprocessing 00000000	What is Quadratization? 0000	Quadratization Techniques
Represent	tations and Bounds		

 $\begin{array}{lll} f &=& -2 - x_1 - x_2 - x_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 \\ &=& -5 + \overline{x}_1 + \overline{x}_2 + \overline{x}_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 \\ &=& -4 + \overline{x}_3 + \overline{x}_1 \overline{x}_2 + x_1 x_3 + x_2 x_3 \\ &=& -3 + x_1 x_2 x_3 + \overline{x}_1 \overline{x}_2 \overline{x}_3 \end{array} \qquad \begin{array}{lll} \text{QPBF} \\ \text{quadratic posiform} \\ \text{quadratic posiform} \\ \text{cubic posiform} \end{array}$ 

#### **Roof Dual Bound**: $C_2(f) \leq f$

(Hammer, Hansen and Simeone, 1984)

ション ふゆ マ キャット マックシン

 $C_2(f) = \text{largest } C \text{ s.t. } f = C + \phi \text{ for some quadratic posiform } \phi.$ 

Depresentations and Downda	
Representations and Bounds	

- $\begin{array}{rcl} f &=& -2 x_1 x_2 x_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -5 + \overline{x}_1 + \overline{x}_2 + \overline{x}_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -4 + \overline{x}_3 + \overline{x}_1 \overline{x}_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -3 + x_1 x_2 x_3 + \overline{x}_1 \overline{x}_2 \overline{x}_3 & 0 \end{array}$
- QPBF quadratic posiform quadratic posiform cubic posiform

#### **Roof Dual Bound**: $C_2(f) \leq f$

(Hammer, Hansen and Simeone, 1984)

 $C_2(f) = \text{largest } C \text{ s.t. } f = C + \phi \text{ for some quadratic posiform } \phi.$ 

Cubic Dual Bound:  $C_3(f) \leq f$ 

(Boros, Crama and Hammer, 1992)

ション ふゆ マ キャット マックシン

 $C_3(f) = \text{largest } C \text{ s.t. } f = C + \psi \text{ for some cubic posiform } \psi.$ 

QUBO 00 <b>0</b> ●0000	Polynomial Time Preprocessing 00000000	What is Quadratization? 0000	Quadratization Techniques 00000000000
Represent	tations and Bounds		

- $\begin{array}{rcl} f &=& -2 x_1 x_2 x_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -5 + \overline{x}_1 + \overline{x}_2 + \overline{x}_3 + x_1 x_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -4 + \overline{x}_3 + \overline{x}_1 \overline{x}_2 + x_1 x_3 + x_2 x_3 & 0 \\ &=& -3 + x_1 x_2 x_3 + \overline{x}_1 \overline{x}_2 \overline{x}_3 & 0 \end{array}$
- QPBF quadratic posiform quadratic posiform cubic posiform

**Roof Dual Bound**:  $C_2(f) \leq f$ 

(Hammer, Hansen and Simeone, 1984)

 $C_2(f) = \text{largest } C \text{ s.t. } f = C + \phi \text{ for some quadratic posiform } \phi.$ 

Cubic Dual Bound:  $C_3(f) \leq f$ 

(Boros, Crama and Hammer, 1992)

ション ふゆ マ キャット マックシン

 $C_3(f) = \text{largest } C \text{ s.t. } f = C + \psi \text{ for some cubic posiform } \psi.$ 

 $\mathbf{C_2}(f) \le \mathbf{C_3}(f) \le \dots \le \mathbf{C_n}(f) = \min f$ 

QUBO
00000000

Polynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

## Outline

# Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

# Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
○○○○ <b>○</b> ●○○	00000000	0000	00000000000
Network	Model for Submodular C	UBO (Hammer, 196	5)

- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows
- There is a one-to-one correspondence between values of f and s t cut values of  $G_f$ . (Hammer, 1965)

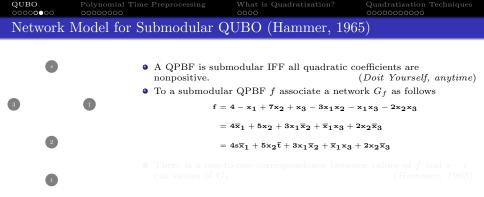
ション ふゆ く は く は く む く む く し く

QUBO 0000 <b>0</b> 000		al Time Preprocessing O	What is Quadratization? 0000	Quadratization Techniques
Netwo	rk Model f	or Submodular (	QUBO (Hammer, 19	65)
3	0	nonpositive. • To a submodul	pmodular IFF all quadratic ar QPBF $f$ associate a net = 4 - x <sub>1</sub> + 7x <sub>2</sub> + x <sub>3</sub> - 3x <sub>1</sub> x <sub>2</sub>	(Doit Yourself, anytime) work $G_f$ as follows
2			to-one correspondence betw f.	
t				

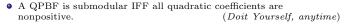
◆□▶ ◆□▶ ◆三▶ ◆三▶ →□ ◆○◆

QUBO 000000		l Time Preprocessing	What is Quadratization? 0000	Quadratization Techniques
Netw	ork Model fo	or Submodular	QUBO (Hammer, 19	965)
<ul> <li>A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yours</li> <li>To a submodular QPBF f associate a network G<sub>f</sub> as fol f = 4 - x<sub>1</sub> + 7x<sub>2</sub> + x<sub>3</sub> - 3x<sub>1</sub>x<sub>2</sub> - x<sub>1</sub>x<sub>3</sub> - 2x<sub>2</sub>x<sub>3</sub></li> </ul>		(Doit Yourself, anytime) work $G_f$ as follows $- \mathbf{x_1 x_3} - 2\mathbf{x_2 x_3}$		
	2		$= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3$ $=$	$+2x_2\overline{x}_3$
	0			

◆□▶ ◆□▶ ◆三▶ ◆三▶ →□ ◆○◆







• To a submodular QPBF f associate a network  $G_f$  as follows

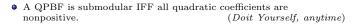
 $\mathbf{f} = 4 - \mathbf{x_1} + 7\mathbf{x_2} + \mathbf{x_3} - 3\mathbf{x_1}\mathbf{x_2} - \mathbf{x_1}\mathbf{x_3} - 2\mathbf{x_2}\mathbf{x_3}$ 

 $= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

 $= 4\mathbf{s}\overline{\mathbf{x}}_1 + 5\mathbf{x}_2\overline{\mathbf{t}} + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

• There is a one-to-one correspondence between values of f and s-tcut values of  $G_f$ . (Hammer, 1965)





• To a submodular QPBF f associate a network  $G_f$  as follows

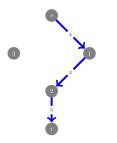
$$\mathbf{f} = 4 - \mathbf{x_1} + 7\mathbf{x_2} + \mathbf{x_3} - 3\mathbf{x_1x_2} - \mathbf{x_1x_3} - 2\mathbf{x_2x_3}$$

$$= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$$

 $=4\mathbf{x}\overline{\mathbf{x}}_1+\mathbf{5}\mathbf{x_2}\overline{\mathbf{t}}+\mathbf{3}\mathbf{x_1}\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x_3}+\mathbf{2}\mathbf{x_2}\overline{\mathbf{x}}_3$ 

• There is a one-to-one correspondence between values of f and s-tcut values of  $G_f$ . (Hammer, 1965)





- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows

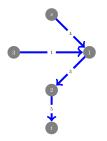
 $\mathbf{f} = \mathbf{4} - \mathbf{x_1} + \mathbf{7}\mathbf{x_2} + \mathbf{x_3} - \mathbf{3}\mathbf{x_1}\mathbf{x_2} - \mathbf{x_1}\mathbf{x_3} - \mathbf{2}\mathbf{x_2}\mathbf{x_3}$ 

 $= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

 $=4\mathbf{s}\overline{\mathbf{x}}_1+5\mathbf{x}_2\overline{\mathbf{t}}+3\mathbf{x}_1\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x}_3+2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

• There is a one-to-one correspondence between values of f and s-t cut values of  $G_f$ . (Hammer, 1965)





- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows

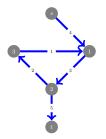
 $\mathbf{f} = \mathbf{4} - \mathbf{x_1} + \mathbf{7}\mathbf{x_2} + \mathbf{x_3} - \mathbf{3}\mathbf{x_1}\mathbf{x_2} - \mathbf{x_1}\mathbf{x_3} - \mathbf{2}\mathbf{x_2}\mathbf{x_3}$ 

 $= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

 $=4\mathbf{x}\overline{\mathbf{x}}_1+5\mathbf{x}_2\overline{\mathbf{t}}+3\mathbf{x}_1\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x}_3+2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

• There is a one-to-one correspondence between values of f and s-t cut values of  $G_f$ . (Hammer, 1965)





- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows

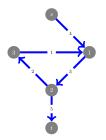
 $\mathbf{f} = \mathbf{4} - \mathbf{x_1} + \mathbf{7}\mathbf{x_2} + \mathbf{x_3} - \mathbf{3}\mathbf{x_1}\mathbf{x_2} - \mathbf{x_1}\mathbf{x_3} - \mathbf{2}\mathbf{x_2}\mathbf{x_3}$ 

 $= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

 $=4\mathbf{x}\overline{\mathbf{x}}_1+5\mathbf{x}_2\overline{\mathbf{t}}+3\mathbf{x}_1\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x}_3+2\mathbf{x}_2\overline{\mathbf{x}}_3$ 

• There is a one-to-one correspondence between values of f and s-t cut values of  $G_f$ . (Hammer, 1965)





- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows

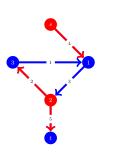
$$f = 4 - x_1 + 7x_2 + x_3 - 3x_1x_2 - x_1x_3 - 2x_2x_3$$

$$= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$$

$$=4\mathbf{x}\overline{\mathbf{x}}_1+5\mathbf{x}_2\overline{\mathbf{t}}+3\mathbf{x}_1\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x}_3+2\mathbf{x}_2\overline{\mathbf{x}}_3$$

• There is a one-to-one correspondence between values of f and s - t cut values of  $G_f$ . (Hammer, 1965)





- A QPBF is submodular IFF all quadratic coefficients are nonpositive. (Doit Yourself, anytime)
- To a submodular QPBF f associate a network  $G_f$  as follows

$$\mathbf{f} = 4 - \mathbf{x_1} + 7\mathbf{x_2} + \mathbf{x_3} - 3\mathbf{x_1x_2} - \mathbf{x_1x_3} - 2\mathbf{x_2x_3}$$

$$= 4\overline{\mathbf{x}}_1 + 5\mathbf{x}_2 + 3\mathbf{x}_1\overline{\mathbf{x}}_2 + \overline{\mathbf{x}}_1\mathbf{x}_3 + 2\mathbf{x}_2\overline{\mathbf{x}}_3$$

$$=4\mathbf{x}\overline{\mathbf{x}}_1+5\mathbf{x}_2\overline{\mathbf{t}}+3\mathbf{x}_1\overline{\mathbf{x}}_2+\overline{\mathbf{x}}_1\mathbf{x}_3+2\mathbf{x}_2\overline{\mathbf{x}}_3$$

• There is a one-to-one correspondence between values of f and s - t cut values of  $G_f$ . (Hammer, 1965)

 $f(0,1,0) = C(\{s,2\},\{1,3,t\}) = 11$ 

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ □ のへで

QUBC	С	
0000	000	10

Polynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

・ロト ・ 四ト ・ 日ト ・ 日 ・

## Outline

# Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models

# • Network Model for General QUBO

- Polynomial Time Preprocessing
  - Components of the Algorithm
  - Computational Results

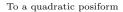
# **B** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

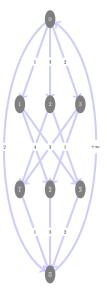
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000●	00000000	0000	00000000000



 $\phi = 2 \times_0 \times_0 + 2\overline{\mathbf{x}}_1 \times_0 + 6\overline{\mathbf{x}}_2 \times_0 + 4\overline{\mathbf{x}}_3 \times_0 + 8\mathbf{x}_1 \mathbf{x}_2 + 6\mathbf{x}_1 \mathbf{x}_3 + 2\mathbf{x}_2 \mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへで



QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000	0000000	0000	0000000000

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

#### • Homogenize it by $x_0$ .

- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
- Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
  - Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

◆□▶ ◆□▶ ★□▶ ★□▶ ● ● ●

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
  - Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
  - Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

( = ) (

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
  - Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三回 ● ○○○

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
- Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
- Associate to  $\gamma x_0 x_0$  one arc  $(x_0, \overline{x}_0)$  with capacity  $c(x_0, \overline{x}_0) = \gamma$  and add arc  $(\overline{x}_0, x_0)$  with capacity  $c(\overline{x}_0, x_0) = +\infty$ .

▲□▶ ▲圖▶ ▲圖▶ ▲圖▶ 二層

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000	0000000	0000	0000000000

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

- Homogenize it by  $x_0$ .
- Associate to each term  $\alpha uv \ (u \neq v)$  two arcs  $(u, \overline{v})$  and  $(v, \overline{u})$  with capacities  $c(u, \overline{v}) = c(v, \overline{u}) = \alpha/2$ .
- Associate to γx<sub>0</sub>x<sub>0</sub> one arc (x<sub>0</sub>, x
  <sub>0</sub>) with capacity c(x<sub>0</sub>, x
  <sub>0</sub>) = γ and add arc (x
  <sub>0</sub>, x<sub>0</sub>) with capacity c(x
  <sub>0</sub>, x<sub>0</sub>) = +∞.

▲ロト ▲周ト ▲ヨト ▲ヨト ヨー のく⊙

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

 $N_{\phi}$  is a symmetric network: twin pair of paths, cycles and flows

- If  $u_0, u_1, ..., u_k$  is a directed path (cycle) in  $N_{\phi}$  then so is  $\overline{u}_k, \overline{u}_{k-1}, ..., \overline{u}_1, \overline{u}_0$ .
- Every feasible circulation in  $N_{\phi}$  has its symmetric twin also feasible, and hence their convex combination is a feasible symmetric circulation.

0

```
\begin{array}{rcl} 1+\mathbf{x}_1\mathbf{x}_3+\overline{\mathbf{x}}_3 & = & \mathbf{x}_0\overline{\mathbf{x}}_1+\mathbf{x}_1\mathbf{x}_3+\overline{\mathbf{x}}_3\mathbf{x}_0+\overline{\mathbf{x}}_0\overline{\mathbf{x}}_0\\ & = & \overline{\mathbf{x}}_0\mathbf{x}_1+\overline{\mathbf{x}}_1\overline{\mathbf{x}}_3+\mathbf{x}_3\overline{\mathbf{x}}_0+\mathbf{x}_0\mathbf{x}_0\\ & = & \mathbf{x}_1\mathbf{x}_3+1 \end{array}
```

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三回 ● ○○○

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
0000000	0000000	0000	0000000000

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

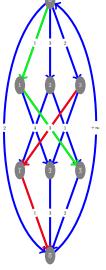
$$V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$$

 $N_{\phi}$  is a symmetric network: twin pair of paths, cycles and flows

- If  $u_0, u_1, ..., u_k$  is a directed path (cycle) in  $N_{\phi}$  then so is  $\overline{u}_k, \overline{u}_{k-1}, ..., \overline{u}_1, \overline{u}_0$ .
- Every feasible circulation in  $N_{\phi}$  has its symmetric twin also feasible, and hence their convex combination is a feasible symmetric circulation.

 $\begin{array}{rcl} \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 &=& \mathbf{x}_0 \overline{\mathbf{x}}_1 + \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 \mathbf{x}_0 + \overline{\mathbf{x}}_0 \overline{\mathbf{x}}_0 \\ &=& \overline{\mathbf{x}}_0 \mathbf{x}_1 + \overline{\mathbf{x}}_1 \overline{\mathbf{x}}_3 + \mathbf{x}_3 \overline{\mathbf{x}}_0 + \mathbf{x}_0 \mathbf{x}_0 \end{array}$ 

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三回 ● ○○○



QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

 $V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$ 

 $N_{\phi}$  is a symmetric network: twin pair of paths, cycles and flows

- If  $u_0, u_1, ..., u_k$  is a directed path (cycle) in  $N_{\phi}$  then so is  $\overline{u}_k, \overline{u}_{k-1}, ..., \overline{u}_1, \overline{u}_0$ .
- Every feasible circulation in  $N_{\phi}$  has its symmetric twin also feasible, and hence their convex combination is a feasible symmetric circulation.

$$\begin{array}{rcl} \overline{\mathbf{x}}_1 + \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 & = & \mathbf{x}_0 \overline{\mathbf{x}}_1 + \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 \mathbf{x}_0 + \overline{\mathbf{x}}_0 \overline{\mathbf{x}}_0 \\ & = & \overline{\mathbf{x}}_0 \mathbf{x}_1 + \overline{\mathbf{x}}_1 \overline{\mathbf{x}}_3 + \mathbf{x}_3 \overline{\mathbf{x}}_0 + \mathbf{x}_0 \mathbf{x}_0 \\ & = & \mathbf{x}_1 \mathbf{x}_3 + 1 \end{array}$$

うして ふゆう ふほう ふほう ふしつ

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000			

To a quadratic posiform

 $\phi = 2\mathbf{x}_0\mathbf{x}_0 + 2\overline{\mathbf{x}}_1\mathbf{x}_0 + 6\overline{\mathbf{x}}_2\mathbf{x}_0 + 4\overline{\mathbf{x}}_3\mathbf{x}_0 + 8\mathbf{x}_1\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_3 + 2\mathbf{x}_2\mathbf{x}_3$ we associate a directed network  $N_{\phi}$  on vertex set

 $V(N_{\phi}) = \{x_0, \overline{x}_0, x_1, \overline{x}_1, \dots, x_n, \overline{x}_n\} \qquad (x_0 \equiv 1)$ 

 $N_{\phi}$  is a symmetric network: twin pair of paths, cycles and flows

- If  $u_0, u_1, ..., u_k$  is a directed path (cycle) in  $N_{\phi}$  then so is  $\overline{u}_k, \overline{u}_{k-1}, ..., \overline{u}_1, \overline{u}_0$ .
- Every feasible circulation in  $N_{\phi}$  has its symmetric twin also feasible, and hence their convex combination is a feasible symmetric circulation.

۲

 $\overline{\mathbf{x}}_1 + \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 = \mathbf{x}_0 \overline{\mathbf{x}}_1 + \mathbf{x}_1 \mathbf{x}_3 + \overline{\mathbf{x}}_3 \mathbf{x}_0 + \overline{\mathbf{x}}_0 \overline{\mathbf{x}}_0 \\ = \overline{\mathbf{x}}_0 \mathbf{x}_1 + \overline{\mathbf{x}}_1 \overline{\mathbf{x}}_3 + \mathbf{x}_3 \overline{\mathbf{x}}_0 + \mathbf{x}_0 \mathbf{x}_0 \\ = \mathbf{x}_1 \mathbf{x}_2 + \mathbf{1}$ 

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 の�?

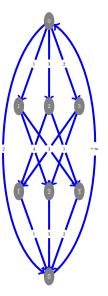
QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sun	, 1989, 1992)	

- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value  $C_2(f)$  is the maximum flow value on arc  $(\bar{x}_0, x_0)$  in a feasible circulation in  $N_{\phi}$ , where  $\phi$  is an arbitrary quadratic posiform of f.
- If  $N_{\psi}$  is the residual network corresponding to such a maximum circulation, then the strong components of  $N_{\psi} \setminus \{(x_0, \overline{x}_0)\}$  induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

▲□▶ ▲圖▶ ▲圖▶ ▲圖▶ 二層

 Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sur	n. 1989, 1992)	



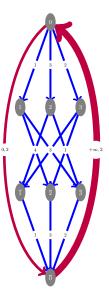
- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value  $C_2(f)$  is the maximum flow value on arc  $(\overline{x}_0, x_0)$  in a feasible circulation in  $N_{\phi}$ , where  $\phi$  is an arbitrary quadratic posiform of f.

If  $N_{\psi}$  is the residual network corresponding to such a maximum circulation, then the strong components of  $N_{\psi} \setminus \{(x_0, \overline{x}_0)\}$  induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

・ロト ・ 同ト ・ ヨト ・ ヨト ・ ヨ

Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sur	. 1989, 1992)	

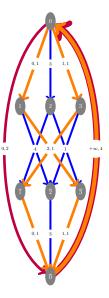


- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value  $C_2(f)$  is the maximum flow value on arc  $(\overline{x}_0, x_0)$  in a feasible circulation in  $N_{\phi}$ , where  $\phi$  is an arbitrary quadratic posiform of f.
- If  $N_{\psi}$  is the residual network corresponding to such a maximum circulation, then the strong components of  $N_{\psi} \setminus \{(x_0, \overline{x}_0)\}$  induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

▲□▶ ▲圖▶ ▲圖▶ ▲圖▶ 二層

 Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sur	. 1989, 1992)	

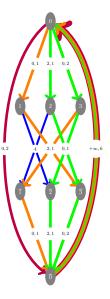


- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value  $C_2(f)$  is the maximum flow value on arc  $(\overline{x}_0, x_0)$  in a feasible circulation in  $N_{\phi}$ , where  $\phi$  is an arbitrary quadratic posiform of f.
- If  $N_{\psi}$  is the residual network corresponding to such a maximum circulation, then the strong components of  $N_{\psi} \setminus \{(x_0, \overline{x}_0)\}$  induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

▲□▶ ▲圖▶ ▲圖▶ ▲圖▶ 二層

 Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sur	n. 1989, 1992)	



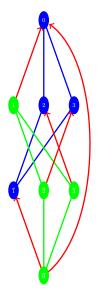
- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value C<sub>2</sub>(f) is the maximum flow value on arc (x
  <sub>0</sub>, x<sub>0</sub>) in a feasible circulation in N<sub>φ</sub>, where φ is an arbitrary quadratic posiform of f.
- If N<sub>ψ</sub> is the residual network corresponding to such a maximum circulation, then the strong components of N<sub>ψ</sub> \ {(x<sub>0</sub>, x<sub>0</sub>)} induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

cf. persistency (Hammer, Hansen and Simeone, 1984) cf. decomposition (Billionet and Sutter, 1992)

・ロト ・ 同ト ・ ヨト ・ ヨト ・ ヨ

Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sur	n. 1989, 1992)	



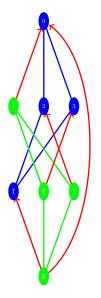
- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value C<sub>2</sub>(f) is the maximum flow value on arc (x
  <sub>0</sub>, x<sub>0</sub>) in a feasible circulation in N<sub>φ</sub>, where φ is an arbitrary quadratic posiform of f.
- If N<sub>ψ</sub> is the residual network corresponding to such a maximum circulation, then the strong components of N<sub>ψ</sub> \ {(x<sub>0</sub>, x
  <sub>0</sub>)} induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

cf. persistency (Hammer, Hansen and Simeone, 1984) cf. decomposition (Billionet and Sutter, 1992)

うして ふゆう ふほう ふほう ふしつ

Recursive application of roof-duality does not provide further improvements!

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
000000 <b>0</b> ●	00000000	0000	00000000000
Implication	Networks (Boros, Hammer, Sun	, 1989, 1992)	



- Two quadratic posiforms  $\phi$  and  $\psi$  represent the same QPBF if and only if  $N_{\psi}$  is the residual network of  $N_{\phi}$  corresponding to a symmetric feasible circulation.
- The roof dual value C<sub>2</sub>(f) is the maximum flow value on arc (x
  <sub>0</sub>, x<sub>0</sub>) in a feasible circulation in N<sub>φ</sub>, where φ is an arbitrary quadratic posiform of f.
- If N<sub>ψ</sub> is the residual network corresponding to such a maximum circulation, then the strong components of N<sub>ψ</sub> \ {(x<sub>0</sub>, π<sub>0</sub>)} induce a decomposition of f, in which each component can be minimized independently of the others to obtain a minimum of f.

cf. persistency (Hammer, Hansen and Simeone, 1984) cf. decomposition (Billionet and Sutter, 1992)

・ロト ・ 日 ・ ・ 田 ・ ・ 田 ・ ・ 日 ・ う へ つ

• Recursive application of roof-duality does not provide further improvements!

QUBO 00000000	Polynomial Time Preprocessing	What is Quadrat 0000
Outline		

Quadratization Techniques

・ロト ・ 四ト ・ 日ト ・ 日 ・

#### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

# 2 Polynomial Time Preprocessing

#### • Components of the Algorithm

• Computational Results

### **3** What is Quadratization?

- Quadratization
- Submodular Functions

### Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

QUBO	Polynomial	Time	Preprocessing
0000000	0000000		

What is Quadratization?

Quadratization Techniques

# Components of the Algorithm

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

- Build implication network
- Compute maximum flow; **fix variables by persistency** (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.
- Output remaining strong components, if any.

QUBO	Polynomial Time	Preprocessing
00000000	0000000	

Quadratization Techniques 00000000000

#### Components of the Algorithm

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

# • Build implication network

- Compute maximum flow; fix variables by persistency (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.
- Output remaining strong components, if any.

QUBO	Polynomial Time Preprocessing	What is Quadratization?
00000000	0000000	0000

Quadratization Techniques

ション ふゆ マ キャット マックシン

#### Components of the Algorithm

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

- Build implication network
- Compute maximum flow; fix variables by persistency (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.
- Output remaining strong components, if any.

QUBO	Polynomial Time Preprocessing $000000000000000000000000000000000000$	What is Quadratization?	Quadratization Techniques
00000000		0000	00000000000
Compone	nts of the Algorithm		

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

- Build implication network
- Compute maximum flow; fix variables by persistency (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.

ション ふゆ マ キャット マックシン

• Output remaining strong components, if any.

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
00000000	⊙●○○○○○○	0000	00000000000
Componer	nts of the Algorithm		

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

- Build implication network
- Compute maximum flow; fix variables by persistency (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.

ション ふゆ マ キャット マックシン

• Output remaining strong components, if any.

QUBO Polynomial Time Preprocessing		What is Quadratization?	Quadratization Techniques
00000000	0000000	0000	0000000000
C	1 C 1 A 1 1 1		

#### Components of the Algorithm

The **purpose** of the preprocessing algorithm is to **fix** some of the variables at their optimum values and **decompose** the remaining problem into several smaller problems which do not share variables.

- Build implication network
- Compute maximum flow; fix variables by persistency (increase capacities of some arcs)
- Probe remaining variables and repeat all of the above as long as there is some change.
- Output remaining strong components, if any.

If the input QPBF is submodular, then the above procedure will fix all the variables at their optimal values in the first round, without any probing.

ション ふゆ マ キャット しょう くしゃ

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techni
00000000	0000000		00000000000
Outline			

・ロト ・ 四ト ・ 日ト ・ 日 ・

#### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

#### 2 Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

#### **3** What is Quadratization?

- Quadratization
- Submodular Functions

#### Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

QUBO	Polynomial Time	e Preprocessing
00000000	00000000	

What is Quadratization?

Quadratization Techniques

・ロト ・ 画 ・ ・ 画 ・ ・ 目 ・ うへぐ

# Via Minimization in VLSI Design

		Percentage of Variables Fixed by					
Problem <sup>1</sup>	n	Roof D	uality	Pr	obing	ALL	Time
		(strong)	(weak)	(forcing)	(equalities)	TOOLS	(sec)
via.c1y	829	93.6%	6.4%	0%	0%	100%	0.03
via.c2y	981	94.7%	5.3%	0%	0%	100%	0.06
via.c3y	1328	94.6%	5.4%	0%	0%	100%	0.09
via.c4y	1367	96.4%	3.6%	0%	0%	100%	0.09
via.c5y	1203	93.1%	6.9%	0%	0%	100%	0.08
via.c1n	828	57.4%	9.6%	32.4%	0.6%	100%	0.49
via.c2n	980	12.4%	4.4%	83.1%	0.1%	100%	7.14
via.c3n	1327	6.8%	5.7%	87.3%	0.2%	100%	18.17
via.c4n	1366	11.1%	1.3%	87.6%	0%	100%	23.08
via.c5n	1202	3.4%	1.4%	95.0%	0.2%	100%	17.13

<sup>1</sup> S. Homer and M. Peinado. Design and performance of parallel and distributed approximation algorithms for maxcut. *Journal of Parallel and Distributed Computing* **46** (1997) 48-61.

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniq
00000000	0000000	0000	0000000000
<b>TT</b> - <b>O</b>			

#### Vertex Cover in Planar Graphs

	Averages for 100 graphs in each of the 4 groups			
	Variables 1	Fixed (%)	Time	(sec)
n	A. D. N. <sup>2</sup>	$\mathbf{QUBO}^3$	A. D. N. <sup>2</sup>	$\mathbf{QUBO}^3$
1000	68.4	100	4.06	0.05
2000	67.4	100	12.24	0.16
3000	65.5	100	30.90	0.27
4000	62.7	100	60.45	0.53

 $^2$  Alber, Dorn, Niedermeier. Experimental evaluation of a tree decomposition based algorithm for vertex cover on planar graphs. Discrete Applied Mathematics 145 (2005) 219-231; 750 GHz, Linux PC, 720 MB

▲□▶ ▲圖▶ ▲目▶ ▲目▶ 目 のへで

 $^3$  Pentium 4, 2.8 GHz, Windows XP, 512 MB

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization
00000000	0000000	0000	00000000000

#### Jumbo Vertex Cover in Planar Graphs

	Computing Times $(\min)^4$		
Vertices	Planar Density		
	10%	50%	90%
50,000	0.7	2.3	0.9
100,000	2.9	10.2	3.9
250,000	19.5	69.8	26.3
500,000	79.3	277.3	106.9

▲□▶ ▲□▶ ▲三▶ ▲三▶ 三三 のへで

 $^4$  Averages over 3 experiments on a Xeon 3.06 GHz, XP, 3.5 GB RAM; ALL problems had 100% of their variables fixed.

QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadra
00000000	00000000	0000	000000

Quadratization Techniques 00000000000

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のへぐ

#### One Dimensional Ising Models

		Average Comp	Average Computing Time (s)		
$\sigma$	Number of Spins	Branch, Cut & Price <sup>5</sup>	Biq Maq <sup>5</sup>	$\mathbf{QUBO}^{6}$	
2.5	100	699	68	1	
	150	92 079	388	3	
	200	N/A	993	9	
	250	N/A	6567	14	
	300	N/A	34 572	21	
3.0	100	256	59	1	
	150	13 491	293	2	
	200	61 271	$1 \ 034$	3	
	250	55 795	3 594	4	
	300	$55\ 528$	8 496	5	

 $^5$  F. Rendl, G. Rinaldi, A. Wiegele. (2007). Solving max-cut to optimality by intersecting semidefinite and polyhedral relaxations.

<sup>6</sup> ALL problems were solved by QUBO.

QUBO	Polynomial Time Preprocessing	What is Quadratization
00000000	0000000	0000

Quadratization Techniques

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のへぐ

# Larger One Dimensional Ising Models

		Average of 3	Average of 3 Problems		
σ	n	Variables not fixed	QUBO Time $(s)^7$		
2.5	500	5	13		
	750	22	30		
	1000	24	53		
	1250	20	81		
	1500	32	124		
3.0	500	0	4		
	750	0	12		
	1000	0	23		
	1250	0	37		
	1500	0	<b>59</b>		

 $^7$  Pentium M, 1.6 GHz 760 MB RAM

QUBO	Polynomial Time Preprocess	
00000000	0000000	

・ロト ・ 四ト ・ 日ト ・ 日 ・

#### Outline

#### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

### Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

### **3** What is Quadratization?

# Quadratization

• Submodular Functions

### Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

QUBO	

olynomial Time Preprocessing

What is Quadratization? 0000

Quadratization Techniques

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

♣ Keep *m* small!

 ♦ Have *g* as submodular as possible!
 ♥ Do not introduce large coefficients!
 ♦ Have it ALL!

 Rosenberg, 1975: All PBFs have polynomial sized quadratizations.
 Zivny, Cohen and Jeavons, 2009: Not all submodular PBFs have submodular quadratizations.
 Ishikawa, 2009, 2011: All PBFs have small quadratizations with no large coefficients.

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへで

QUBO	Tim

olynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

うして ふゆう ふほう ふほう ふしつ

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

#### Keep m small!

- $\diamond$  Have g as submodular as possible!
- $\heartsuit$  Do not introduce **large** coefficients!
- ♦ Have it ALL!

#### **Rosenberg, 1975: All PBFs have polynomial sized** quadratizations.

Zivny, Cohen and Jeavons, 2009: Not all submodular

PBFs have submodular quadratizations.

Ishikawa, 2009, 2011: All PBFs have small quadratizations with no large coefficients.

QUBO	Time

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

Keep m small!
 Have g as submodular as possible!
 Do not introduce large coefficients!
 Have it ALL!
 Rosenberg, 1975: All PBFs have polynomial sized quadratizations.
 Zivny, Cohen and Jeanons, 2009: Not all submodular PBFs have submodular quadratizations.
 Ishikawa, 2009, 2011: All PBFs have small quadratizations with no large coefficients.

◆□▶ ◆□▶ ◆ □▶ ◆ □▶ ● □ ● ● ●

QUBO	Tir

olynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

うして ふゆう ふほう ふほう ふしつ

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

 $\clubsuit$  Keep m small!

 $\diamond$  Have g as submodular as possible!

- $\heartsuit$  Do not introduce **large** coefficients!
- ♠ Have it ALL!

# **Rosenberg, 1975: All** PBFs have **polynomial sized** quadratizations.

Zivny, Cohen and Jeavons, 2009: Not all submodular

PBFs have submodular quadratizations.

Ishikawa, 2009, 2011: All PBFs have small quadratizations with no large coefficients.

QUBO	Τiı

lynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

Keep m small!
 Have g as submodular as possible!
 Do not introduce large coefficients!
 Have it ALL!
 Rosenberg, 1975: All PBFs have polynomial sized quadratizations.
 Zivny, Cohen and Jeavons, 2009: Not all submodular PBFs have submodular quadratizations.
 Ishikawa, 2009, 2011: All PBFs have small quadratizations with large coefficients.

◆□▶ ◆□▶ ◆目▶ ◆目▶ 目 のへぐ

QUBO	Tir

lynomial Time Preprocessing

What is Quadratization?

Quadratization Techniques

ション ふゆ マ キャット しょう くしゃ

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

• Keep m small! • Have g as submodular as possible!

- $\heartsuit$  Do not introduce **large** coefficients!
- ♠ Have it ALL!

# Rosenberg, 1975: All PBFs have polynomial sized quadratizations.

 Zivny, Cohen and Jeavons, 2009: Not all submodular PBFs have submodular quadratizations.
 Ishikawa, 2009, 2011: All PBFs have small quadratizations with m large coefficients.

QUBO	Tim

What is Quadratization?

Quadratization Techniques

ション ふゆ マ キャット しょう くしゃ

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

Keep m small!

- $\diamond$  Have *g* as submodular as possible!
- $\heartsuit$  Do not introduce **large** coefficients!
- ♠ Have it ALL!

# Rosenberg, 1975: All PBFs have polynomial sized quadratizations.

Zivny, Cohen and Jeavons, 2009: Not all submodular PBFs have submodular quadratizations.

**Ishikawa, 2009, 2011:** All PBFs have **small** quadratizations with no **large** coefficients.

QUBO		Time
0000000	00000000	

ション ふゆ マ キャット しょう くしゃ

#### Quadratization of PBFs

• Given  $f: \{0,1\}^n \to \mathbb{R}$  find quadratic  $g: \{0,1\}^{n+m} \to \mathbb{R}$  such that

$$f(\mathbf{x}) = \min_{\mathbf{y} \in \{0,1\}^m} g(\mathbf{x}, \mathbf{y}) \quad \forall \ \mathbf{x} \in \{0,1\}^n.$$

♣ Keep *m* small!

- $\diamond$  Have *g* as submodular as possible!
- $\heartsuit$  Do not introduce **large** coefficients!
- ♠ Have it ALL!

# Rosenberg, 1975: All PBFs have polynomial sized quadratizations.

Zivny, Cohen and Jeavons, 2009: Not all submodular PBFs have submodular quadratizations.

Ishikawa, 2009, 2011: All PBFs have small quadratizations with no large coefficients.

QUBO	Polynomial Tir	
00000000	00000000	

What is Quadratization?

Quadratization Techniques

・ロト ・ 四ト ・ 日ト ・ 日 ・

#### Outline

#### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

# Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

### Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

- Polynomial recognition if deg(f) ≤ 3. (Billionnet and Minoux, 1985)
   Recognition is NP-hard if deg(f) ≥ 4. (Gallo and Simeone, 1989; Crama 1989)
- A QPBF is submodular iff it has no positive quadratic terms.
- A submodular QPBO is solved by the network based preprocessing.

(Hammer, 1965)

・ロト ・ 四ト ・ 日ト ・ 日 ・

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

- Polynomial recognition if deg(f) ≤ 3. (Billionnet and Minoux, 1985)
- Recognition is NP-hard if  $\deg(f) \ge 4$ .

(Gallo and Simeone, 1989; Crama 1989) • A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)

• A submodular QPBO is solved by the network based preprocessing.

(*Hammer*, 1965)

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

- Polynomial recognition if  $deg(f) \le 3$ . (Billionnet and Minoux, 1985)
- Recognition is NP-hard if  $\deg(f) \ge 4$ .
  - (Gallo and Simeone, 1989; Crama 1989)
- A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)
- A submodular QPBO is solved by the network based preprocessing.

(*Hammer*, 1965)

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

• Polynomial recognition if  $\deg(f) \leq 3$ .

(Billionnet and Minoux, 1985)

• Recognition is NP-hard if  $\deg(f) \ge 4$ .

(Gallo and Simeone, 1989; Crama 1989)

- A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)
- A submodular QPBO is solved by the network based preprocessing.

(*Hammer*, 1965)

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

• Polynomial recognition if  $\deg(f) \leq 3$ .

(Billionnet and Minoux, 1985)

• Recognition is NP-hard if  $\deg(f) \ge 4$ .

(Gallo and Simeone, 1989; Crama 1989)

- A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)
- A submodular QPBO is solved by the network based preprocessing.

(Hammer, 1965)

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

• Polynomial recognition if  $\deg(f) \leq 3$ .

(Billionnet and Minoux, 1985)

• Recognition is NP-hard if  $\deg(f) \ge 4$ .

(Gallo and Simeone, 1989; Crama 1989)

- A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)
- A submodular QPBO is solved by the network based preprocessing.

(Hammer, 1965)

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

• A PBF  $f: \{0,1\}^n \to \mathbb{R}$  is submodular if

 $f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y}) \quad \forall \ \mathbf{x}, \mathbf{y} \in \{0, 1\}^n.$ 

• Polynomial recognition if  $\deg(f) \leq 3$ .

(Billionnet and Minoux, 1985)

• Recognition is NP-hard if  $\deg(f) \ge 4$ .

(Gallo and Simeone, 1989; Crama 1989)

- A QPBF is submodular iff it has no positive quadratic terms. (Nemhauser and Wolsey, 1981)
- A submodular QPBO is solved by the network based preprocessing.

(Hammer, 1965)

ション ふゆ マ キャット マックタン

- Which PBFs have submodular quadratization?
- How to recognize if a PBF has a submodular quadratization?

QUBO		Time	
00000000	00000000		

Quadratization Techniques

### Outline

### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

## Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

### • Penalty Function

- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

Quadratization Techniques

#### Rosenberg's Penalty Functions Method (1975)

$$p(x, y, w) = xy - 2xw - 2yw + 3w = \begin{cases} = 0 \text{ if } w = xy, \\ \ge 1 \text{ if } w \neq xy \end{cases}$$

 $f(x, y, ...) = xyA + B = \min_{w \in \{0, 1\}} wA + B + Mp(x, y, w)$ 

if M is large enough.

・ロト ・ 四ト ・ 日ト ・ 日 ・

- Many positive quadratic terms with large coefficients (recursion!), even if the input is subodular.
- **NP-hard** to find a quadratization in this way with the **minimum number of new variables**.
- Not possible to substitute the product of 3 or more variables with a single new variable.

$$p(x, y, w) = xy - 2xw - 2yw + 3w = \begin{cases} = 0 \text{ if } w = xy, \\ \ge 1 \text{ if } w \neq xy \end{cases}$$

$$f(x, y, ...) = xyA + B = \min_{w \in \{0, 1\}} wA + B + Mp(x, y, w)$$

if 
$$M$$
 is large enough.

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

- Many positive quadratic terms with large coefficients (recursion!), even if the input is subodular.
- **NP-hard** to find a quadratization in this way with the **minimum number of new variables**.
- Not possible to substitute the product of 3 or more variables with a single new variable.

$$p(x, y, w) = xy - 2xw - 2yw + 3w = \begin{cases} = 0 \text{ if } w = xy, \\ \ge 1 \text{ if } w \neq xy \end{cases}$$

$$f(x, y, ...) = xyA + B = \min_{w \in \{0, 1\}} wA + B + Mp(x, y, w)$$

if M is large enough.

- Many positive quadratic terms with large coefficients (recursion!), even if the input is subodular.
- **NP-hard** to find a quadratization in this way with the **minimum number of new variables**.
- Not possible to substitute the product of 3 or more variables with a single new variable.

$$p(x, y, w) = xy - 2xw - 2yw + 3w = \begin{cases} = 0 \text{ if } w = xy, \\ \ge 1 \text{ if } w \neq xy \end{cases}$$

$$f(x, y, ...) = xyA + B = \min_{w \in \{0, 1\}} wA + B + Mp(x, y, w)$$

if 
$$M$$
 is large enough.

ション ふゆ マ キャット しょう くしゃ

- Many positive quadratic terms with large coefficients (recursion!), even if the input is subodular.
- **NP-hard** to find a quadratization in this way with the **minimum number of new variables**.
- Not possible to substitute the product of 3 or more variables with a single new variable.

$$p(x, y, w) = xy - 2xw - 2yw + 3w = \begin{cases} = 0 \text{ if } w = xy, \\ \ge 1 \text{ if } w \neq xy \end{cases}$$

$$f(x, y, ...) = xyA + B = \min_{w \in \{0, 1\}} wA + B + Mp(x, y, w)$$

if 
$$M$$
 is large enough.

ション ふゆ マ キャット しょう くしゃ

- Many positive quadratic terms with large coefficients (recursion!), even if the input is subodular.
- **NP-hard** to find a quadratization in this way with the **minimum number of new variables**.
- Not possible to substitute the product of 3 or more variables with a single new variable.

QUBO		Time	
00000000	00000000		

Quadratization Techniques

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

### Outline

### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

## Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

• Penalty Function

### • Termwise Quadratization

- Multiple Split of Terms
- Splitting Off Common Parts
- Results

### Negative Terms

• Kolmogorov and Zabih (2004), Fredman and Drineas (2005):

$$-x_1 x_2 \cdots x_d = \min_{w \in \{0,1\}} w(d-1 - x_1 - x_2 \cdots - x_d)$$

• Rother, Kohli, Feng and Jia (2009):

$$-\prod_{j\in N} \overline{x}_j \prod_{j\in P} x_j = \min_{u,v\in\{0,1\}} -uv + u \sum_{j\in N} x_j + v \sum_{j\in P} \overline{x}_j$$

• Only one or two new variables per term; at most one positive quadratic term; no large coefficients.

#### Theorem (vs. Billionet and Minoux (1985))

Quadratization Techniques

・ロト ・ 同ト ・ ヨト ・ ヨト ・ ヨ

### Negative Terms

• Kolmogorov and Zabih (2004), Fredman and Drineas (2005):

$$-x_1 x_2 \cdots x_d = \min_{w \in \{0,1\}} w(d-1 - x_1 - x_2 \cdots - x_d)$$

• Rother, Kohli, Feng and Jia (2009):

$$-\prod_{j\in N}\overline{x}_j\prod_{j\in P}x_j = \min_{u,v\in\{0,1\}}-uv+u\sum_{j\in N}x_j+v\sum_{j\in P}\overline{x}_j$$

• Only one or two new variables per term; at most one positive quadratic term; no large coefficients.

#### Theorem (vs. Billionet and Minoux (1985))

Quadratization Techniques

#### Negative Terms

• Kolmogorov and Zabih (2004), Fredman and Drineas (2005):

$$-x_1 x_2 \cdots x_d = \min_{w \in \{0,1\}} w(d-1 - x_1 - x_2 \cdots - x_d)$$

• Rother, Kohli, Feng and Jia (2009):

$$-\prod_{j\in N}\overline{x}_j\prod_{j\in P}x_j = \min_{u,v\in\{0,1\}}-uv+u\sum_{j\in N}x_j+v\sum_{j\in P}\overline{x}_j$$

• Only one or two new variables per term; at most one positive quadratic term; no large coefficients.

#### Theorem (vs. Billionet and Minoux (1985))

Quadratization Techniques

うして ふゆう ふほう ふほう ふしつ

#### Negative Terms

• Kolmogorov and Zabih (2004), Fredman and Drineas (2005):

$$-x_1 x_2 \cdots x_d = \min_{w \in \{0,1\}} w(d-1 - x_1 - x_2 \cdots - x_d)$$

• Rother, Kohli, Feng and Jia (2009):

$$-\prod_{j\in N}\overline{x}_j\prod_{j\in P}x_j = \min_{u,v\in\{0,1\}}-uv + u\sum_{j\in N}x_j + v\sum_{j\in P}\overline{x}_j$$

• Only one or two new variables per term; at most one positive quadratic term; no large coefficients.

#### Theorem (vs. Billionet and Minoux (1985))

Quadratization Techniques

ション ふゆ く は く は く む く む く し く

#### Positive Terms

• Ishikawa (2009, 2011):

$$\prod_{j=1}^{a} x_j = S_2(\mathbf{x}) + \min_{\mathbf{w} \in \{0,1\}^k} B(\mathbf{w}) - 2A(\mathbf{w})S_1(\mathbf{x}) + \rho \left[S_1(\mathbf{x}) - d + 1\right]$$

where  $d = 2k + 2 - \rho$ ,  $\rho \in \{0, 1\}$ , and

$$S_1(\mathbf{x}) = \sum_{j=1}^d x_j \qquad S_2(\mathbf{x}) = \sum_{\substack{1 \le i < j \le d \\ k}} x_i x_j$$
$$A(\mathbf{w}) = \sum_{j=1}^k w_j \qquad B(\mathbf{w}) = \sum_{j=1}^k (4j-1)w_j$$

• Only  $\approx d/2$  new variables per term; no large coefficients; many positive quadratic terms.

Quadratization Techniques

i

ション ふゆ く は く は く む く む く し く

#### Positive Terms

d

• Ishikawa (2009, 2011):

$$\prod_{j=1}^{a} x_j = S_2(\mathbf{x}) + \min_{\mathbf{w} \in \{0,1\}^k} B(\mathbf{w}) - 2A(\mathbf{w})S_1(\mathbf{x}) + \rho \left[S_1(\mathbf{x}) - d + 1\right]$$

where  $d = 2k + 2 - \rho$ ,  $\rho \in \{0, 1\}$ , and

$$S_1(\mathbf{x}) = \sum_{j=1}^d x_j \qquad S_2(\mathbf{x}) = \sum_{\substack{1 \le i < j \le d \\ k}} x_i x_j$$
$$A(\mathbf{w}) = \sum_{j=1}^k w_j \qquad B(\mathbf{w}) = \sum_{j=1}^k (4j-1)w$$

• Only  $\approx d/2$  new variables per term; no large coefficients; many positive quadratic terms.

QUBO		$\operatorname{Time}$	
00000000	00000000		

Quadratization Techniques

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

### Outline

### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

### Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

 $i \in I$ 

Quadratization Techniques

#### Multiple Splits

Assume that  $\phi_i(\mathbf{w}) \in \{0,1\}$  for  $i \in [q], \mathbf{w} \in \{0,1\}^p$  such that

$$\min_{\mathbf{w}\in\{0,1\}^p} \sum_{i=1}^q \phi_i(\mathbf{w}) = 1, \quad \text{and}$$
$$\forall I \subsetneqq [q] \quad \exists \mathbf{w}^* \in \{0,1\}^p \quad \text{s.t.} \quad \sum \phi_i(\mathbf{w}^*) = 0.$$

or instance  $\phi_1 = u_1$ ,  $\phi_2 = u_2$ , and  $\phi_2 = \overline{u_1} \overline{u_2}$  is such a system

#### Theorem

If  $P_i$ ,  $i \in [q]$  are subsets of indices covering [d], then we have

$$\prod_{j=1}^{d} x_{j} = \min_{\mathbf{w} \in \{0,1\}^{p}} \sum_{i=1}^{q} \phi_{i}(\mathbf{w}) \prod_{j \in P_{i}} x_{j}$$

With  $p = \lceil \log q \rceil$  new variables we can split a degree d = kq term into q terms of degree k + p.

▲ロト ▲囲 ▶ ▲ 臣 ▶ ▲ 臣 ▶ ● 臣 ■ ● の Q (2)

 $i \in I$ 

Quadratization Techniques

#### Multiple Splits

A

Assume that  $\phi_i(\mathbf{w}) \in \{0,1\}$  for  $i \in [q], \mathbf{w} \in \{0,1\}^p$  such that

$$\min_{\mathbf{w}\in\{0,1\}^p} \sum_{i=1}^q \phi_i(\mathbf{w}) = 1, \quad \text{and}$$
$$I \subsetneq [q] \quad \exists \mathbf{w}^* \in \{0,1\}^p \quad \text{s.t.} \quad \sum \phi_i(\mathbf{w}^*) = 0.$$

For instance  $\phi_1 = w_1$ ,  $\phi_2 = w_2$ , and  $\phi_3 = \overline{w}_1 \overline{w}_2$  is such a system.

#### Theorem

If  $P_i$ ,  $i \in [q]$  are subsets of indices covering [d], then we have

$$\prod_{j=1}^{d} x_{j} = \min_{\mathbf{w} \in \{0,1\}^{p}} \sum_{i=1}^{q} \phi_{i}(\mathbf{w}) \prod_{j \in P_{i}} x_{j}$$

With  $p = \lceil \log q \rceil$  new variables we can split a degree d = kq term into q terms of degree k + p.

▲ロト ▲囲 ▶ ▲ 臣 ▶ ▲ 臣 ▶ ● 臣 ■ ● の Q (2)

Quadratization Techniques

#### Multiple Splits

Assume that  $\phi_i(\mathbf{w}) \in \{0,1\}$  for  $i \in [q], \mathbf{w} \in \{0,1\}^p$  such that

$$\min_{\mathbf{w} \in \{0,1\}^p} \sum_{i=1}^q \phi_i(\mathbf{w}) = 1, \quad \text{and}$$
$$\forall I \subsetneqq [q] \quad \exists \mathbf{w}^* \in \{0,1\}^p \quad \text{s.t.} \quad \sum_{i \in I} \phi_i(\mathbf{w}^*) = 0$$

For instance  $\phi_1 = w_1$ ,  $\phi_2 = w_2$ , and  $\phi_3 = \overline{w}_1 \overline{w}_2$  is such a system.

#### Theorem

If  $P_i$ ,  $i \in [q]$  are subsets of indices covering [d], then we have

$$\prod_{j=1}^{d} x_{j} = \min_{\mathbf{w} \in \{0,1\}^{p}} \sum_{i=1}^{q} \phi_{i}(\mathbf{w}) \prod_{j \in P_{i}} x_{j}$$

With  $p = \lceil \log q \rceil$  new variables we can split a degree d = kq term into q terms of degree k + p.

▲□▶ ▲□▶ ▲三▶ ▲三▶ 三三 のへで

Quadratization Techniques

#### Multiple Splits

Assume that  $\phi_i(\mathbf{w}) \in \{0,1\}$  for  $i \in [q], \mathbf{w} \in \{0,1\}^p$  such that

$$\min_{\mathbf{w}\in\{0,1\}^p} \sum_{i=1}^q \phi_i(\mathbf{w}) = 1, \quad \text{and}$$
$$\forall I \subsetneqq [q] \quad \exists \mathbf{w}^* \in \{0,1\}^p \quad \text{s.t.} \quad \sum_{i\in I} \phi_i(\mathbf{w}^*) = 0.$$

For instance  $\phi_1 = w_1$ ,  $\phi_2 = w_2$ , and  $\phi_3 = \overline{w}_1 \overline{w}_2$  is such a system.

#### Theorem

If  $P_i$ ,  $i \in [q]$  are subsets of indices covering [d], then we have

$$\prod_{j=1}^{d} x_{j} = \min_{\mathbf{w} \in \{0,1\}^{p}} \sum_{i=1}^{q} \phi_{i}(\mathbf{w}) \prod_{j \in P_{i}} x_{j}$$

With  $p = \lceil \log q \rceil$  new variables we can split a degree d = kq term into q terms of degree k + p.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへで

QUBO		$\operatorname{Time}$	
00000000	00000000		

Quadratization Techniques

・ロト ・ 四ト ・ 日ト ・ 日 ・

### Outline

### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

### Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

0000000 0000000	0000	00000000000

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

$$g(\mathbf{x}) = \sum_{H \in \mathcal{H}} \alpha_H \prod_{j \in C \cup H} x_j,$$

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

Theorem (Set of Positive Terms)

$$g(\mathbf{x}) = \min_{w \in \{0,1\}} \left( \sum_{H \in \mathcal{H}} \alpha_H \right) w \prod_{j \in C} x_j + \sum_{H \in \mathcal{H}} \alpha_H \overline{w} \prod_{j \in H} x_j.$$

Theorem (Set of Negative Terms)

$$-g(\mathbf{x}) = \min_{w \in \{0,1\}} \sum_{H \in \mathcal{H}} \alpha_H w \left( 1 - \prod_{j \in C} x_j - \prod_{j \in H} x_j \right)$$

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三日 のへで

0000000 0000000 0000 0000 00000	QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
	00000000	0000000	0000	000000000000

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

$$g(\mathbf{x}) = \sum_{H \in \mathcal{H}} \alpha_H \prod_{j \in C \cup H} x_j,$$

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

Theorem (Set of Positive Terms)

$$g(\mathbf{x}) = \min_{w \in \{0,1\}} \left( \sum_{H \in \mathcal{H}} \alpha_H \right) w \prod_{j \in C} x_j + \sum_{H \in \mathcal{H}} \alpha_H \overline{w} \prod_{j \in H} x_j.$$

Theorem (Set of Negative Terms)

$$-g(\mathbf{x}) = \min_{w \in \{0,1\}} \sum_{H \in \mathcal{H}} \alpha_H w \left( 1 - \prod_{j \in C} x_j - \prod_{j \in H} x_j \right).$$

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三日 のへで

00000000 00000000 00000 0000 000000000	QUBO	Polynomial Time Preprocessing	What is Quadratization?	Quadratization Techniques
	00000000	0000000	0000	000000000000

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

$$g(\mathbf{x}) = \sum_{H \in \mathcal{H}} \alpha_H \prod_{j \in C \cup H} x_j,$$

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

Theorem (Set of Positive Terms)

$$g(\mathbf{x}) = \min_{w \in \{0,1\}} \left( \sum_{H \in \mathcal{H}} \alpha_H \right) w \prod_{j \in C} x_j + \sum_{H \in \mathcal{H}} \alpha_H \overline{w} \prod_{j \in H} x_j.$$

Theorem (Set of Negative Terms)

$$-g(\mathbf{x}) = \min_{w \in \{0,1\}} \sum_{H \in \mathcal{H}} \alpha_H w \left( 1 - \prod_{j \in C} x_j - \prod_{j \in H} x_j \right).$$

▲ロト ▲園ト ▲ヨト ▲ヨト 三百一のへで

QUBO		Time	
00000000	00000000		

Quadratization Techniques

・ロト ・ 一日 ・ ・ 日 ・ ・ 日 ・ ・ 日 ・

### Outline

### Quadratic Unconstrained Binary Optimization

- Quadratic Pseudo-Boolean Functions
- Representations and Bounds
- Origin of Graph Cut Models
- Network Model for General QUBO

### Polynomial Time Preprocessing

- Components of the Algorithm
- Computational Results

# **3** What is Quadratization?

- Quadratization
- Submodular Functions

# Quadratization Techniques

- Penalty Function
- Termwise Quadratization
- Multiple Split of Terms
- Splitting Off Common Parts
- Results

Quadratization Techniques

うして ふゆう ふほう ふほう ふしつ

#### Corollary

A PBF in n variables, with t terms of degree d has a quadratization with  $\approx n + k \binom{n}{k} + \frac{td}{k}$  new variables and with at most n - 1 positive quadratic terms, for any  $k \geq 1$ .

Ishikawa's method provides a quadratization with  $\approx n + \frac{td}{2}$  new variables and  $\max\{\binom{n}{2}, t\binom{d}{2}\}$  positive quadratic terms.

Figure : Performance comparison of reductions, on Ishikawa's benchmarks. Relative performance of our method is shown as  $\Delta$ . (Joint work with Alexander Fix and Ramin Zabih (Cornell University).)

#### Corollary

A PBF in n variables, with t terms of degree d has a quadratization with  $\approx n + k \binom{n}{k} + \frac{td}{k}$  new variables and with at most n - 1 positive quadratic terms, for any  $k \geq 1$ .

Ishikawa's method provides a quadratization with  $\approx n + \frac{td}{2}$  new variables and max $\{\binom{n}{2}, t\binom{d}{2}\}$  positive quadratic terms.

Figure : Performance comparison of reductions, on Ishikawa's benchmarks. Relative performance of our method is shown as  $\Delta$ . (Joint work with Alexander Fix and Ramin Zabih (Cornell University).)

うして ふゆう ふほう ふほう ふしつ

### Corollary

A PBF in n variables, with t terms of degree d has a quadratization with  $\approx n + k \binom{n}{k} + \frac{td}{k}$  new variables and with at most n - 1 positive quadratic terms, for any  $k \geq 1$ .

Ishikawa's method provides a quadratization with  $\approx n + \frac{td}{2}$  new variables and max $\{\binom{n}{2}, t\binom{d}{2}\}$  positive quadratic terms.

	New variables	# positive terms	# terms	% fixed by QPBO
Ishikawa	224,346	421,897	1,133,811	80.4%
Our method	236,806	38,343	677, 183	96.1%
Δ	+6%	-90%	-40%	+20%

Figure : Performance comparison of reductions, on Ishikawa's benchmarks. Relative performance of our method is shown as  $\Delta$ . (Joint work with Alexander Fix and Ramin Zabih (Cornell University).)