A Second Order Total Variation MIP Model for Image Segmentation Using the L_0 Norm
(*on going work*)

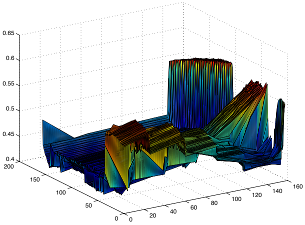
R. Shen,[1] S. Canu, G. Reinelt, F. Lenzen, C. Schnörr

Journée Optimisation dans les Réseaux, 17 novembre 2017

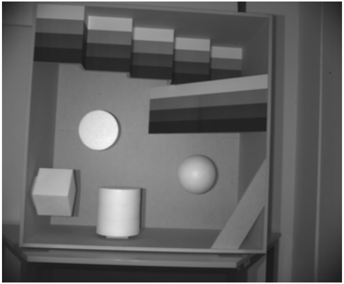
[1]Electronic address: ruobing.shen@informatik.uni-heidelberg.de

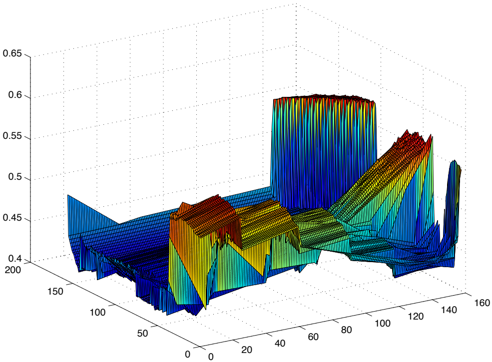
Road map

1. The problem
2. The 1d case
3. Mixed integer (binary) programming (MIP)
4. The 2d case
    - The 2d case
    - The multi-cut constraints



The problem: deepth estimation



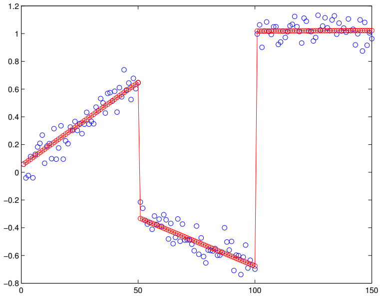


Fundamental hypothesis

piecewise linear model

Road map

1. The problem
2. The 1d case
3. Mixed integer (binary) programming (MIP)
4. The 2d case
   - The 2d case
   - The multi-cut constraints



The second order TV Potts model in the 1d case

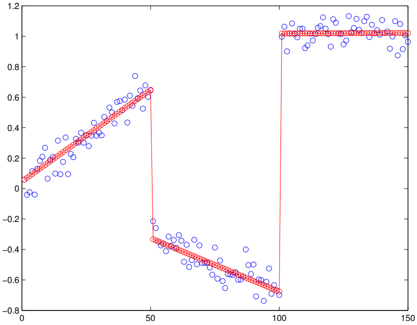
The objective: given n observations $(z_1, \dots, z_n)$

Retrieve the best linear fit with k discontinuities

$$\min_{\mathbf{w}\in\mathbb{R}^n} \quad \|\mathbf{w}-\mathbf{z}\|_1$$
$$\text{s.t.} \quad \|\nabla_x^2 \mathbf{w}\|_0 \leq k$$

piece wise linear model

$$w_i = a_\ell i + b_\ell$$
$$\ell = 1, \dots, k^* + 1;$$

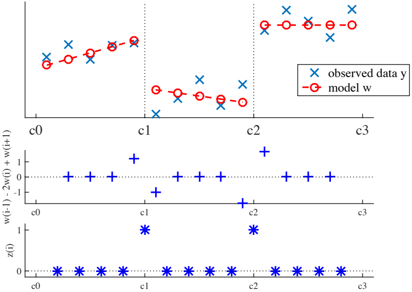


$\|\nabla_x^2 \mathbf{w}\|_0 \leq k$ with k small

- $\|\nabla_x^2 \mathbf{w}\|_0 = 0$ impose linear model
- $\|\nabla_x^2 \mathbf{w}\|_0 \neq 0$ allow discontinuity

The second order TV Potts model as a MIP

$$\min_{\mathbf{w}\in\mathbb{R}^n} \quad \|\mathbf{w}-\mathbf{z}\|_1$$
$$\text{s.t.} \quad \|\nabla_x^2 \mathbf{w}\|_0 \le k$$



$$\nabla_x^2 \mathbf{w} \propto w_{i-1} - 2w_i + w_{i+1}$$

$$\begin{cases} w_{i-1} - 2w_i + w_{i+1} = 0 & \Rightarrow \quad x_i = 1, \quad i = 2,\dots,n-1, \\ w_{i-1} - 2w_i + w_{i+1} \neq 0 & \Rightarrow \quad x_i = 0, \quad i = 2,\dots,n-1 \end{cases}$$

For a given M large enough (the big M trick)

$$\min_{\mathbf{w}\in\mathbb{R}^n, \mathbf{x}\in\{0,1\}^{n-1}} \quad \sum_{i=1}^{n} |w_i - z_i|$$
$$\text{s.t.} \quad |w_{i-1} - 2w_i + w_{i+1}| \le M(x_{i-1} + x_i), \quad i = 2,\dots,n-1,$$
$$\sum_{i=1}^{n-1} x_i \le k.$$

Eliminating the absolute values: positive and negative parts

$$\min_{\mathbf{w}\in\mathbb{R}^n,\mathbf{x}\in\{0,1\}^{n-1}} \sum_{i=1}^{n} |w_i - z_i|$$

$$\text{s.t. } |w_{i-1} - 2w_i + w_{i+1}| \leq M(x_{i-1} + x_i), \quad i = 2, \ldots, n-1,$$

$$\sum_{i=1}^{n-1} x_i \leq k.$$

Positive and negative parts: $z_+, z_- \geq 0$

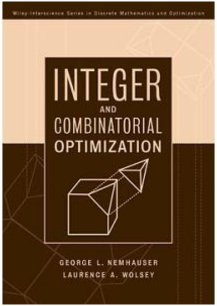
$$z = z_+ - z_- \qquad \text{and} \qquad |z| = z_+ + z_-$$

$$\underset{\mathbf{w},\varepsilon^+,\varepsilon^-\in\mathbb{R}^n,\mathbf{x}\in\{0,1\}^{n-1}}{\text{minimize}} \quad \sum_{i=1}^{n}(\varepsilon_i^+ + \varepsilon_i^-)$$

$$\begin{aligned}
\text{s.t.} \quad & \mathbf{w} - \mathbf{z} = \varepsilon^+ - \varepsilon^- \\
& w_{i-1} - 2w_i + w_{i+1} \leq M(x_{i-1} + x_i), \quad i = 2, \ldots, n-1 \\
& -w_{i-1} + 2w_i - w_{i+1} \leq M(x_{i-1} + x_i), \quad i = 2, \ldots, n-1 \\
& \sum_{i=1}^{n-1} x_i \leq k \qquad\qquad 0 \leq \varepsilon^+,\ 0 \leq \varepsilon^-
\end{aligned}$$

Road map

1. The problem
2. The 1d case
3. Mixed integer (binary) programming (MIP)
4. The 2d case
   - The 2d case
   - The multi-cut constraints



Mixed integer linear program (MILP)

- linear cost
- linear constraints
- integer and continuous variables

Definition (mixed integer linear program – MILP (canonical form))

$$\begin{cases} \min\limits_{\mathbf{w}\in\mathbb{R}^p,\ \mathbf{x}\in\mathbb{N}^q} & f(\mathbf{w},\mathbf{x}) = \mathrm{c}^t\mathbf{w} + \mathrm{d}^t\mathbf{x} & \longleftarrow \text{linear} \\ \text{s.t.} & A\mathbf{w} + B\mathbf{z} \leq \mathrm{b} & \longleftarrow \text{linear} \\ & \mathbf{w} \geq 0, & \end{cases}$$

for some given $\mathrm{c} \in \mathbb{R}^p, \mathrm{d} \in \mathbb{R}^m, A \in \mathbb{R}^{m\times p}, B \in \mathbb{R}^{m\times q}$ and $\mathrm{d} \in \mathbb{R}^q$.

- A mixed binary linear program is a MILP with $\mathbf{x} \in \{0,1\}^q$ binary.
- When its domain is not empty and bounded, a MILP admit a unique global minimum.

Mixed integer quadratic program (MIQP)

- quadratic cost
- linear constraints
- integer and continuous variables

Definition (mixed integer quadratic program – MIQP)

$$\left\{\begin{array}{lll} \min\limits_{\mathbf{z}=(\mathbf{w},\mathbf{x})\in\mathbb{R}^p\times\mathbb{N}^q} & f(\mathbf{z}) = \frac{1}{2}\mathbf{z}^t Q\mathbf{z} + \mathrm{c}^t\mathbf{z} & \longleftarrow \text{quadratic} \\ \text{s.t.} & A\mathbf{z} \leq \mathrm{b} & \longleftarrow \text{linear} \\ & \mathbf{z} \geq 0, & \end{array}\right.$$

for some given symmetric matrix $Q \in \mathbb{R}^{(p+q)\times(p+q)}$

Mixed integer quadratically constrained quadratic program (MIQCP).

- quadratic cost, quadratic constraints, integer and continuous variables

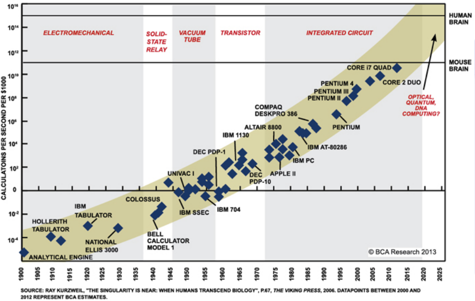
Problems hierarchy

MILP (= MIQP with $Q = 0$) $\subset$ MIQP $\subset$ MIQCP

Progresses in MILP

in 1989

MILP is a powerful modeling tool, "They are, however, theoretically complicated and computationally cumbersome" [Bix10]



from 1996 to 2016 [Bix10, AW13]

	improvement factor
machine	$\times 2^{10} = 1000 - 1600$
solver	$\times 1000 - 3600$
formulation	???
global	$\times 1 - 5\ 10^6$

a year to solve 10 – 20 years ago $\longrightarrow$ now 30 seconds

"mixed integer linear techniques are nowadays mature, that is fast, robust, and are able to solve problems with up to millions of variables" [GMMS12]

Mixed integer software (available with matlab)

Software package	Matlab function
Open source	
GLPK	glpk for mixed integer linear programming
COIN OR	(not matlab yet)
Commercial	
Matlab	intlinprog for mixed integer linear programming
CPLEX	cplexmilp for mixed integer linear programming
	cplexmiqp for mixed integer quadratic programming
	cplexmiqcp for mixed integer quadratically constrained pg
GUROBI	gurobi for MILP, MIQP and MIQCQP
Xpress	xpress for MILP, MIQP and MIQCQP
SCIP	opti_scip for MILP, MIQP and MIQCQP

Mixed Integer Linear Programming Benchmark (MIPLIB2010)

recommend CPLEX, GUROBI and Xpress (NOT intlinprog)

http://plato.asu.edu/ftp/milpc.html

So far

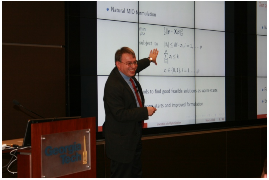
- The MIP hierarchy: MILP (linear) $\subset$ MIQP (quadratic) $\subset$ MIQCQP

- use standard MIP software

- the joy of having an exact solution

- formulate the problems as a MILP or MIQP (if possible)

- Help the solver: finding optimal solution faster
  - find a nice initialization
  - use stronger constraints



How to accelerate the solver?

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^n, \mathbf{x}\in\{0,1\}^{n-1}} \quad & \sum_{i=1}^{n} |w_i - z_i| \\ \text{s.t.} \quad & |w_{i-1} - 2w_i + w_{i+1}| \leq M(x_{i-1} + x_i), \quad i = 2, \ldots, n-1, \\ & \sum_{i=1}^{n-1} x_i \leq k. \end{aligned}$$

Initialization: use first order algorithm (ADMM, proximal, greedy...)

- initialize $\mathbf{w}$ and $\mathbf{x}$
- initialize parameter M

Stronger constraints

- based on the convex hull of the actual constraints
- typically: $\|D_n w\|_1 \leq k$
- local implied bound cuts to treat the big-M [BBF+16]

[BKM15] claim: *Adding these bounds typically leads to improved performance of the MIO, especially in delivering lower bound certificates*

First order algorithm :toward a local minimum

$$D_n = \begin{pmatrix} 1 & -2 & 1 & 0 & \dots & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 \\ & & & \ddots & & \\ 0 & \dots & 0 & 1 & -2 & 1 \end{pmatrix} \quad \in \mathbb{R}^{n-2 \times n}$$

$$\left\{ \begin{array}{cl} \min\limits_{\mathbf{w} \in \mathbb{R}^n} & \|\mathbf{w} - \mathbf{z}\|_1 \\ \text{s.t.} & \|D_n \mathbf{w}\|_0 \leq k \end{array} \right. \quad \Rightarrow \quad \left\{ \begin{array}{cl} \min\limits_{\mathbf{w} \in \mathbb{R}^n, \mathbf{r} \in \mathbb{R}^{n-2}} & \|\mathbf{w} - \mathbf{z}\|_1 \\ \text{s.t.} & \|\mathbf{r}\|_0 \leq k \quad \text{and} \quad \mathbf{r} = D_n \mathbf{w} \end{array} \right.$$

The augmented Lagrangian with $\lambda > 0$

$$L(\mathbf{w}, \mathbf{r}, \Lambda) = \|\mathbf{w} - \mathbf{z}\|_1 + \Lambda^t(\mathbf{r} - D_n \mathbf{w}) + \frac{\lambda}{2}\|\mathbf{r} - D_n \mathbf{w}\|^2 + I_{\{\|\mathbf{r}\|_0 \leq k\}}$$

The ADMM algorithm:

$$\begin{aligned} \mathbf{w}^{k+1} &= \arg\min_{\mathbf{w}} L(\mathbf{w}^k, \mathbf{r}^k, \Lambda^k) \\ \mathbf{r}^{k+1} &= \arg\min_{\mathbf{r}} L(\mathbf{w}^{k+1}, \mathbf{r}^k, \Lambda^k) \\ \Lambda^{k+1} &= \Lambda^k + \rho(D_n \mathbf{w}^{k+1} - \mathbf{r}^{k+1}) \end{aligned}$$

Combine the best of the two worlds

The problem:

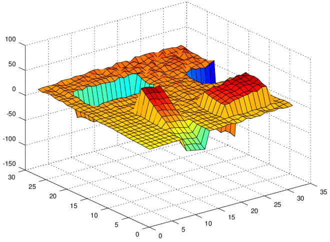
$$
\begin{aligned}
\min_{\mathbf{w}\in\mathbb{R}^n,\mathbf{x}\in\{0,1\}^{n-1}} \quad & \sum_{i=1}^{n} |w_i - y_i| \\
\text{s.t.} \quad & |w_{i-1} - 2w_i + w_{i+1}| \leq M(x_{i-1} + x_i), \quad i = 2, \ldots, n-1, \\
& \sum_{i=1}^{n-1} x_i \leq k.
\end{aligned}
$$

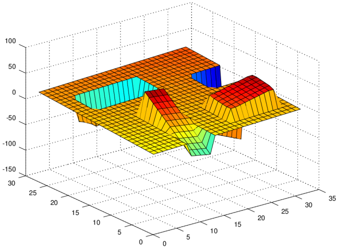
The proposed solution [BKM15]

1. $(\mathbf{w}, \mathbf{x}) \leftarrow$ ADMM alternating proximal gradient method
2. use $\mathbf{w}$ and $\mathbf{x}$ as a warm start for MILP (with Cplex)
   - $(\mathbf{w}, \mathbf{x}) \leftarrow$ Polish coefficients on the active set
   - initialize the constant M

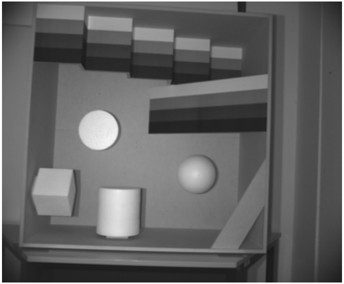
Road map

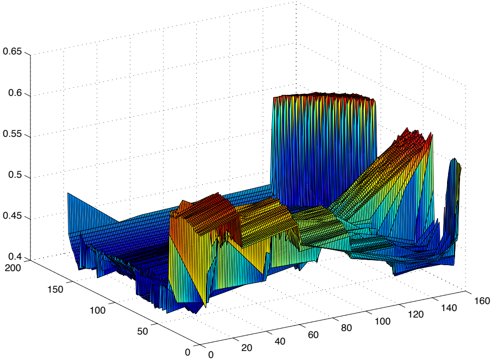
1. The problem
2. The 1d case
3. Mixed integer (binary) programming (MIP)
4. The 2d case
   - The 2d case
   - The multi-cut constraints





The problem: deepth estimation based on image Z





Fundamental hypothesis

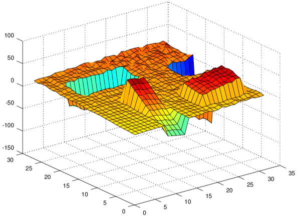
piecewise linear model

ℓ_0 constraints on row and columns

$$\min_{W \in R^{m\times n}} \quad \|W - Z\|_1$$

$$\text{s.t.} \quad \|\nabla_x^2 W\|_0 \le k_{r_i}, \quad \forall i = 1, \dots, n,$$

$$\|\nabla_y^2 W\|_0 \le k_{c_j}, \quad \forall j = 1, \dots, m.$$



Fundamental hypothesis

piecewise 2d linear model

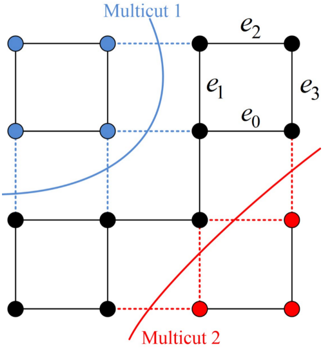
the problem as a MILP

$$
\begin{aligned}
\min_{W \in R^{m \times n}} \quad & \|W - Z\|_1 \\
\text{s.t.} \quad & \|\nabla_x^2 W\|_0 \leq k_{r_i}, \quad \forall i = 1, \dots, n, \\
& \|\nabla_y^2 W\|_0 \leq k_{c_j}, \quad \forall j = 1, \dots, m.
\end{aligned}
$$

$$
\begin{aligned}
\underset{\substack{W \in \mathbb{R}^{n \times m}}}{\text{minimize}} \quad & \sum_{i=1}^{n} \sum_{j=1}^{m} |W_{ij} - Z_{ij}| \\
x \in \{0,1\}^{n \times (m-1)} \quad & \text{row} \\
y \in \{0,1\}^{(n-1) \times m} \quad & \text{column} \\
\text{s.t.} \quad & |W_{i,j+1} + W_{i,j-1} - 2W_{ij}| \leq M_{r_i}(x_{i,j-1} + x_{ij}), \quad i = 1, n \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad j = 2, m-1 \\
& |W_{i+1,j} + W_{i-1,j} - 2W_{ij}| \leq M_{c_j}(y_{i-1,j} + y_{ij}), \quad j = 1, m \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i = 2, n-1 \\
& \sum_{j=2}^{m-1} x_{ij} \leq k_{r_i}, \qquad\qquad i = 1, , n \\
& \sum_{i=2}^{n-1} y_{ij} \leq k_{c_j}, \qquad\qquad j = 1, , m,
\end{aligned}
$$

A strongest formulation with multi-cut constraints

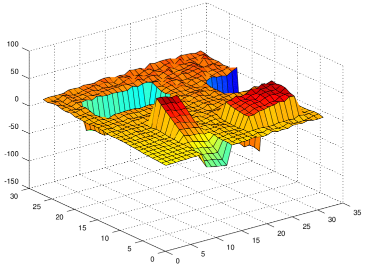
row cut: $x \in \{0,1\}^{n\times(m-1)}$ column cut: $y \in \{0,1\}^{(n-1)\times m}$

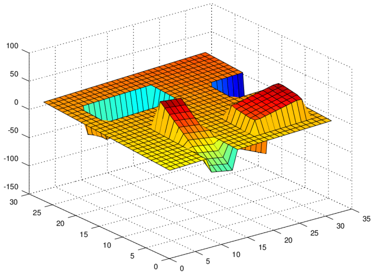


$4 \times (n-2)^2$ additional constraints

$$y_1 + x_2 + y_3 \geq x_0, \qquad x_0 + x_2 + y_3 \geq y_1,$$
$$x_0 + y_1 + y_3 \geq x_2, \qquad x_0 + y_1 + x_2 \geq y_3.$$

The matlab simulation






```
function [ W,X,Y,fval ] = Snd_Order_TV_MILP( Z,kr,kc,Tmax,disp )

Z           28x28   = 784

Reduced MIP has 5283 rows, 4541 columns, and 22807 nonzeros.
Reduced MIP has 1104 binaries, 0 generals, 0 SOSs, and 0 indicators.
```

10 sec to find the minimum
more than 10 minutes to prove it
can be reduced [BBF+16]

Road map (done)

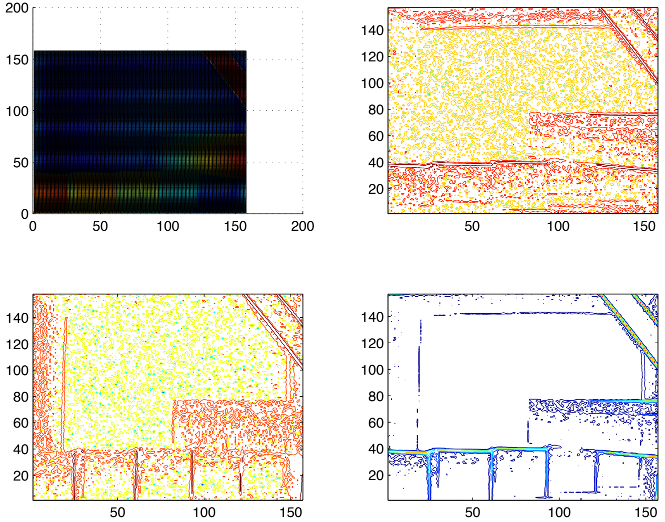
1. The problem
2. The 1d case
3. Mixed integer (binary) programming (MIP)
4. The 2d case

Conclusion

- Machine learning with MIP [SRC17]

pros	cons
it works	it does not scale
global optimum	only linear or quadratic
flexible	show some instability
that is what we want to do	it's not what we want to do

- Future work
  - efficient generic solver
  - efficient implementation: parallelization, randomisation, GPU
  - efficient hyper parameter calibration


200
150
100
50
0
0
50
100
150
200
140
120
100
80
60
40
20
50
100
150
140
120
100
80
60
40
20
50
100
150
140
120
100
80
60
40
20
50
100
150

[AW13] Tobias Achterberg and Roland Wunderling. Mixed integer programming: Analyzing 12 years of progress. In *Facets of Combinatorial Optimization*, pages 449–481. Springer, 2013.

[BBF+16] Pietro Belotti, Pierre Bonami, Matteo Fischetti, Andrea Lodi, Michele Monaci, Amaya Nogales-Gómez, and Domenico Salvagnin. On handling indicator constraints in mixed integer programming. *Computational Optimization and Applications*, 65(3):545–566, 2016.

[Bix10] Robert E Bixby. Mixed-integer programming: It works better than you may think. In *FERC Conference*, 2010.

[BKM15] Dimitris Bertsimas, Angela King, and Rahul Mazumder. Best subset selection via a modern optimization lens. *arXiv preprint arXiv:1507.03133*, 2015.

[GMMS12] Björn Geißler, Alexander Martin, Antonio Morsi, and Lars Schewe. Using piecewise linear functions for solving minlps. In *Mixed Integer Nonlinear Programming*, pages 287–314. Springer, 2012.

[SRC17] Ruobing Shen, Gerhard Reinelt, and Stéphane Canu. A first derivative potts model for segmentation and denoising using MILP. *CoRR*, abs/1709.07212, 2017.