Real-Time Solution of Mixed-Integer Quadratic Programs Using Decision Diagrams

Shaoning Han

Department of Mathematics
National University of Singapore

NUSRI Workshop
December 2025

Collaborators

Andres Gomez
ISE, USC

Leonardo Lozano
OBAIS, U of Cincinnati

Agenda

Introduction

We consider

$$\begin{aligned} \min_{x,z}\ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t. }\ & x_i(1-z_i) = 0 \quad \forall i \in [n] \\ & z \in Z \subseteq \{0,1\}^n, \end{aligned} \tag{MIQP}$$

where

- $Q \succeq 0$ is PSD
- $z_i = \mathbb{I}_{x_i \neq 0}$ is the "support" of x_i
- Z capture logical conditions - cardinality, disjunction, conjunction, implication

Introduction

We consider

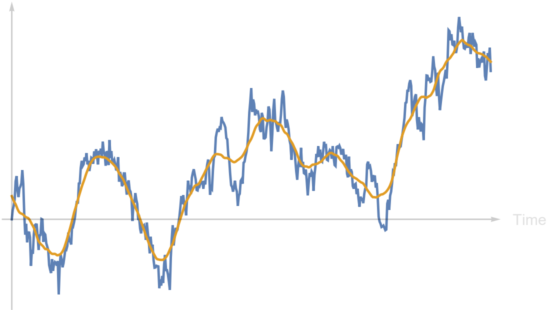
$$\begin{aligned} \min_{x,z}\ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.}\ & x_i(1-z_i) = 0 \quad \forall i \in [n] \\ & z \in Z \subseteq \{0,1\}^n, \end{aligned} \tag{MIQP}$$

where

- $Q \succeq 0$ is PSD
- $z_i = \mathbb{I}_{x_i \neq 0}$ is the "support" of x_i
- Z capture logical conditions - cardinality, disjunction, conjunction, implication

Motivation Application – Monitoring Problem

Monitoring Problem Assume at each time stamp i, a datapoint y_i is generated by a sensor system. Using the most recent observations $\{y_i\}_{i=1}^n$ **at each time stamp**, one aims to infer the true value of time series process $\{x_i\}_{i=1}^n$ to detect changes or anomalies.

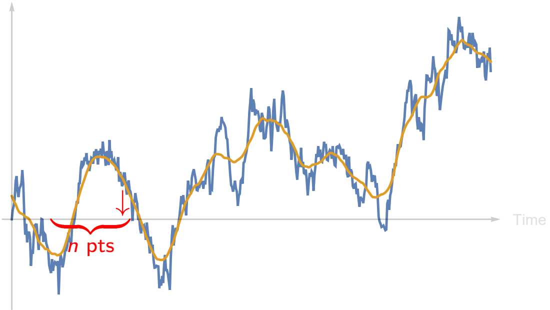


Applications

- Manufacturing system Yan et al. (2017)
- Personalized medicine Dunn et al. (2018)

Motivation Application – Monitoring Problem

Monitoring Problem Assume at each time stamp i, a datapoint y_i is generated by a sensor system. Using the most recent observations $\{y_i\}_{i=1}^n$ **at each time stamp**, one aims to infer the true value of time series process $\{x_i\}_{i=1}^n$ to detect changes or anomalies.

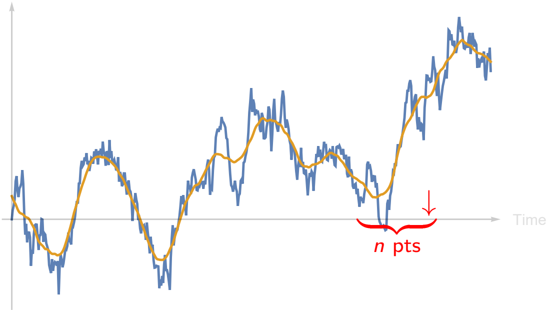


Applications

- Manufacturing system Yan et al. (2017)
- Personalized medicine Dunn et al. (2018)

Motivation Application – Monitoring Problem

Monitoring Problem Assume at each time stamp i, a datapoint y_i is generated by a sensor system. Using the most recent observations $\{y_i\}_{i=1}^n$ **at each time stamp**, one aims to infer the true value of time series process $\{x_i\}_{i=1}^n$ to detect changes or anomalies.



Applications

- Manufacturing system Yan et al. (2017)
- Personalized medicine Dunn et al. (2018)

Motivation Application – Monitoring Problem

At each time stamp, the monitoring problem can be modeled as

$$\min_{x,z\in\mathbb{R}^n} \underbrace{\frac{1}{2}\sum_{i=1}^{n}(x_i - y_i)^2}_{\text{fitness}} + \underbrace{\frac{1}{2}x^\top Rx}_{\text{regularizer}} + \underbrace{\mu\|x\|_0}_{\text{sparsity}}$$

where $Q = I + R$, $c = \mu\mathbf{1}$, $d = -y$,

Commonly used regularizer

- Moving average: $x^\top Rx = \lambda\sum_i\left(x_i - \frac{1}{k}\sum_{j=1}^{k} x_{i-j}\right)^2$ (bandwidth k)
- Ridge: $x^\top Rx = \lambda\|x\|_2^2$ (bandwidth k=0)
- Hodrick-Prescott: $x^\top Rx = \lambda\sum_i(x_{i-2} - 2x_{i-1} + x_i)^2$ (bandwidth $k = 2$)
- k-order differences (bandwidth k)

Motivation Application – Monitoring Problem

At each time stamp, the monitoring problem can be modeled as

$$\min_{x,z\in\mathbb{R}^n} \underbrace{\frac{1}{2}\sum_{i=1}^{n}(x_i - y_i)^2}_{\text{fitness}} + \underbrace{\frac{1}{2}x^\top Rx}_{\text{regularizer}} + \underbrace{\mu\sum_{i=1}^{n} z_i}_{\text{sparsity}}$$

$$\text{s.t. } x_i(1-z_i) = 0,\ z_i \in \{0,1\} \quad \forall i = 1,\dots,n$$

where $Q = I + R$, $c = \mu\mathbf{1}$, $d = -y$, $\|x\|_0 = \#$ nonzero entries in x

Commonly used regularizer

- Moving average: $x^\top Rx = \lambda\sum_i \left(x_i - \frac{1}{k}\sum_{j=1}^{k} x_{i-j}\right)^2$ (bandwidth k)
- Ridge: $x^\top Rx = \lambda\|x\|_2^2$ (bandwidth k=0)
- Hodrick-Prescott: $x^\top Rx = \lambda\sum_i(x_{i-2} - 2x_{i-1} + x_i)^2$ (bandwidth $k = 2$)
- k-order differences (bandwidth k)

Motivation Application – Monitoring Problem

At each time stamp, the monitoring problem can be modeled as

$$\min_{x,z\in\mathbb{R}^n} \underbrace{\frac{1}{2}\sum_{i=1}^{n}(x_i-y_i)^2}_{\text{fitness}} + \underbrace{\frac{1}{2}x^\top Rx}_{\text{regularizer}} + \underbrace{\mu\sum_{i=1}^{n} z_i}_{\text{sparsity}}$$

$$\text{s.t. } x_i(1-z_i)=0,\ z_i\in\{0,1\} \quad \forall i=1,\dots,n$$

where $Q = I + R,\ c = \mu\mathbf{1},\ d = -y,\ \ Z = \{0,1\}^n$

Commonly used regularizer

- Moving average: $x^\top Rx = \lambda\sum_i\left(x_i - \frac{1}{k}\sum_{j=1}^k x_{i-j}\right)^2$ (bandwidth k)
- Ridge: $x^\top Rx = \lambda\|x\|_2^2$ (bandwidth k=0)
- Hodrick-Prescott: $x^\top Rx = \lambda\sum_i(x_{i-2}-2x_{i-1}+x_i)^2$ (bandwidth $k=2$)
- k-order differences (bandwidth k)

Goal

How hard is the problem?

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Q x + d^\top x + c^\top z \\ \text{s.t. } & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z \subseteq \{0,1\}^n \end{aligned}$$

- $\mathcal{NP}$-hard in general, e.g., OLS, $Q = I +$ Rank-one

Goal

How hard is the problem?

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Q x + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z \subseteq \{0,1\}^n \end{aligned}$$

- $\mathcal{NP}$-hard in general, e.g., OLS, $Q = I +$ Rank-one

In online settings:

- MIP solvers? Assume 1000 time stamps and utilizing the most recent 200 observations to make inference $\Rightarrow$ around 1000 MIQPs each with 200 binary vars!

Goal

How hard is the problem?

$$
\begin{aligned}
\min_{x,z}\ & \frac{1}{2}x^\top Q x + d^\top x + c^\top z \\
\text{s.t. }\ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\
& z \in Z \subseteq \{0,1\}^n
\end{aligned}
$$

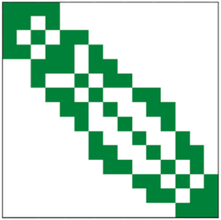
- $\mathcal{NP}$-hard in general, e.g., OLS, $Q = I +$ Rank-one

Goal: get a **real-time** solution to (MIQP) in **online** settings (that is, data d is revealed at each time stamp)

Goal

How hard is the problem?

$$\min_{x,z} \frac{1}{2}x^\top Q x + d^\top x + c^\top z$$
$$\text{s.t. } x_i(1 - z_i) = 0 \quad \forall i \in [n]$$
$$z \in Z \subseteq \{0,1\}^n$$



- $\mathcal{NP}$-hard in general, e.g., OLS, $Q = I +$ Rank-one

Goal: get a **real-time** solution to (MIQP) in **online** settings (that is, data d is revealed at each time stamp)

Assumption: Q has a small bandwidth k , i.e., $Q_{ij} = 0$ if $|i - j| > k$

Agenda

Historical Origin

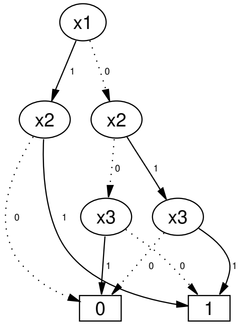
Decision diagrams encode Boolean functions

- Lee (1959), Akers (1978), Bryant (1986)
- Historically used for circuit design and verification

Example

$$f(x) = (\neg x_1 \wedge \neg x_2 \wedge \neg x_3) \vee (x_1 \wedge x_2) \vee (x_2 \wedge x_3)$$

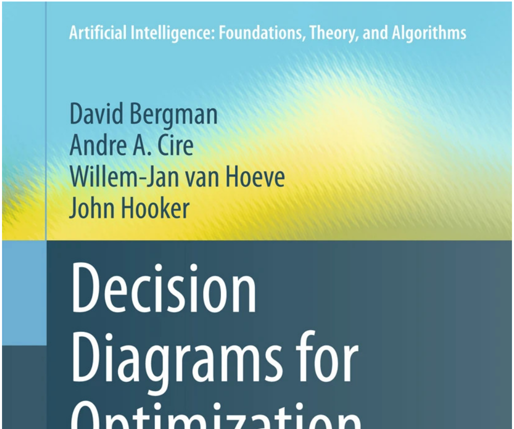
x_1	x_2	x_3	$f(x)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



Decision Diagram for Binary Linear Optimization

Using DD to solve binary linear optimization problems was pioneered by CMU scholars

$$\min f(z) \stackrel{\text{def}}{=} c^\top z$$
$$\text{s.t. } z \in Z \subseteq \{0,1\}^n$$

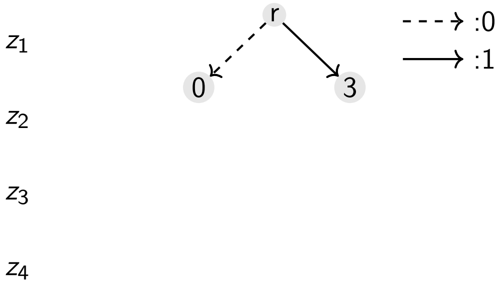


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z\in\{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Arc $\Leftrightarrow$ assigment of z_i
- Each node $\Leftrightarrow$ a state: total weights of the selected items

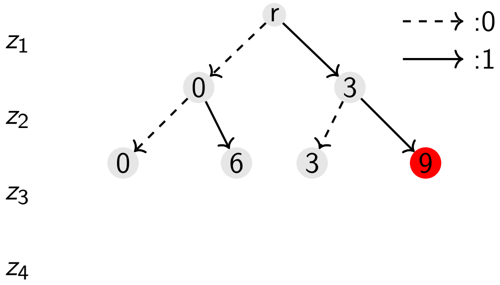


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Remove infeasible nodes



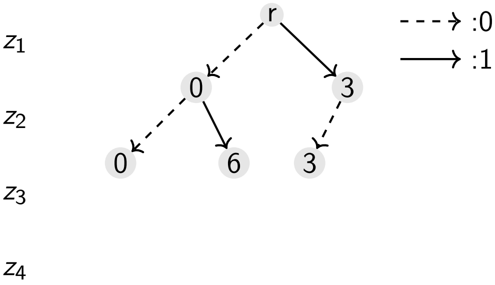
Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$

$$\text{s.t.} \; 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Remove infeasible nodes

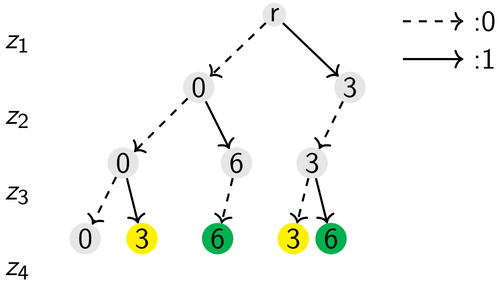


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \le 6$$

- Merge two ndoes with the same states in the same layer

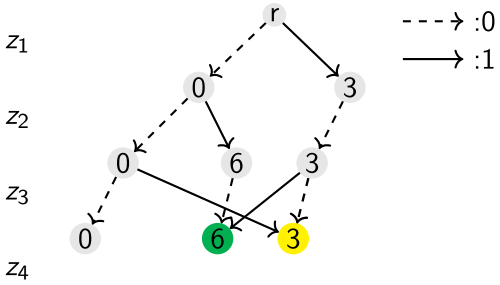


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z\in\{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Merge two ndoes with the same states in the same layer

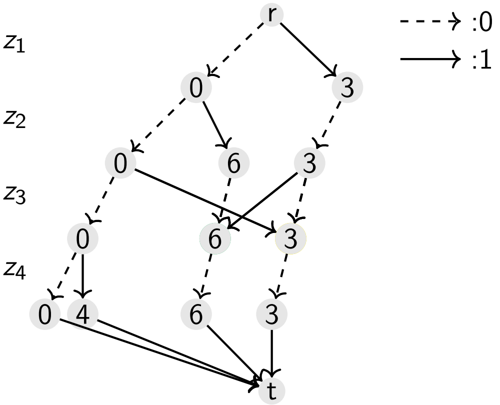


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \le 6$$

- Each (r-t) path $\Leftrightarrow$ a feasible solution z

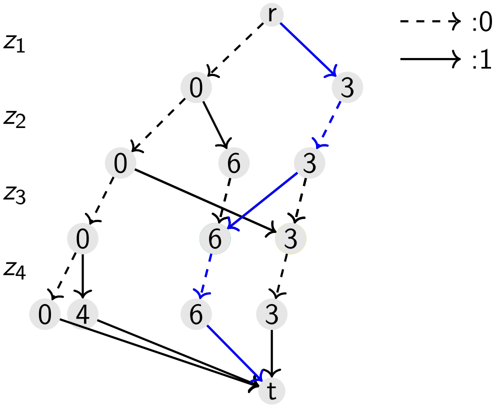


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Each (r-t) path ⇔ a feasible solution z

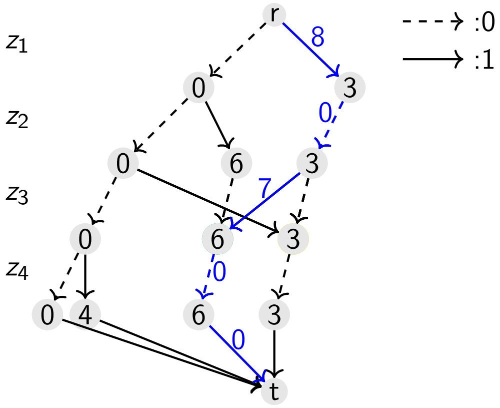


Decision Diagram for Binary Linear Optimization

To illustrate, consider a knapsack problem

$$\max_{z \in \{0,1\}^4} \; 8z_1 + 14z_2 + 7z_3 + 6z_4$$
$$\text{s.t. } 3z_1 + 6z_2 + 3z_3 + 4z_4 \leq 6$$

- Arc length = obj coef
- Path length = obj of a feasible sol
- Binary program ⇔ shortest/longest path problem for an acyclic directed graph



Decision Diagram for Binary Linear Optimization

More Comments

- Decision diagram is one way to express dynamic programming
  - state space $\{s^\ell\}$
  - transition function $\phi(s^\ell, \hat{z}^\ell)$
  - cost function/arc length ℓ_a
- Decision diagram is an effective tool to explore comninatorial structures

  The 0-1 inequality

$$300z_0 + 300z_1 + 285z_2 + 285z_3 + 265z_4 + 265z_5 + 230z_6 + 230z_7 + 190z_8 + 200z_9 + 400z_{10} + 200z_{11} + 400z_{12} + 200z_{13} + 400z_{14} + 200z_{15} + 400z_{16} + 200z_{17} + 400z_{18} \leq 2700$$

  has 117,520 minimal feasible solutions (or minimal covers). But its reduced BDD has only 152 nodes...
- Relaxed/restricted DD, variable ordering, etc...

Agenda

DD Construction for MIQP

We first assume $Z = \{0, 1\}^n$

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2} x^\top Q x + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in \{0, 1\}^n \end{aligned} \tag{MIQP}$$

Question: how to construct a decision diagram for problems involving <u>continuous</u> variables and <u>nonseparable</u> objectives?

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x : x \circ (1-z) = 0} \frac{1}{2} x^\top Q x + d^\top x = -\frac{1}{2} d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2} \langle (Q \circ zz^\top)^\dagger, d^\top d \rangle,$$

where $\left[(Q \circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i, j \in S$ and 0 otherwise.

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x: x\circ(1-z)=0} \frac{1}{2}x^\top Qx + d^\top x = -\frac{1}{2}d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2}\langle (Q \circ zz^\top)^\dagger, d^\top d\rangle,$$

where $\left[(Q \circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i,j \in S$ and 0 otherwise.

Example Consider $Q = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 2 \end{pmatrix}$ and $z = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$. Then

- $S = \{2,3\}$
- $Q_{SS}^{-1} = \begin{pmatrix} 3 & -1 \\ -1 & 2 \end{pmatrix}^{-1} = \begin{pmatrix} 2/5 & 1/5 \\ 1/5 & 3/5 \end{pmatrix}$
- $(Q \circ zz^\top)^\dagger = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2/5 & 1/5 \\ 0 & 1/5 & 3/5 \end{pmatrix}$

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x: x\circ(1-z)=0} \frac{1}{2}x^\top Qx + d^\top x = -\frac{1}{2}d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2}\langle (Q\circ zz^\top)^\dagger, d^\top d\rangle,$$

where $\left[(Q\circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i,j \in S$ and 0 otherwise.

State Space Assume v_ℓ is one node at layer ℓ corresponding to the partial solution $z^\ell \in \{0,1\}^{\ell-1}$. Define the state of v_ℓ as

$$s_{v_\ell} = \left[Q \circ \hat{z}^\ell \left(\hat{z}^\ell\right)^\top\right]^\dagger,$$

where $\hat{z}^\ell \in \mathbb{R}^n$ is defined by $\hat{z}_i^\ell = z_i^\ell$ if $i \leq \ell - 1$ and 0 otherwise.

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x: x \circ (1-z) = 0} \frac{1}{2} x^\top Q x + d^\top x = -\frac{1}{2} d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2} \langle (Q \circ zz^\top)^\dagger, d^\top d \rangle,$$

where $\left[(Q \circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i, j \in S$ and 0 otherwise.
State Space Assume v_ℓ is one node at layer ℓ corresponding to the partial solution $z^\ell \in \{0,1\}^{\ell-1}$. Define the state of v_ℓ as

$$s_{v_\ell} = \left[Q \circ \hat{z}^\ell \left(\hat{z}^\ell \right)^\top \right]^\dagger,$$

where $\hat{z}^\ell \in \mathbb{R}^n$ is defined by $\hat{z}_i^\ell = z_i^\ell$ if $i \leq \ell - 1$ and 0 otherwise.
Transition Function $s_{v_{\ell+1}} - s_{v_\ell}$ can be computed using rank-one updates

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x : x \circ (1-z) = 0} \frac{1}{2} x^\top Q x + d^\top x = -\frac{1}{2} d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2} \langle (Q \circ zz^\top)^\dagger, d^\top d \rangle,$$

where $\left[(Q \circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i, j \in S$ and 0 otherwise.

State Space Assume v_ℓ is one node at layer ℓ corresponding to the partial solution $z^\ell \in \{0,1\}^{\ell-1}$. Define the state of v_ℓ as

$$s_{v_\ell} = \left[Q \circ \hat{z}^\ell \left(\hat{z}^\ell \right)^\top \right]^\dagger,$$

where $\hat{z}^\ell \in \mathbb{R}^n$ is defined by $\hat{z}_i^\ell = z_i^\ell$ if $i \le \ell - 1$ and 0 otherwise.

Transition Function $s_{v_{\ell+1}} - s_{v_\ell}$ can be computed using rank-one updates

Arc Length $\ell_{v_\ell v_{\ell+1}} = g(\hat{z}^{\ell+1}) - g(\hat{z}^\ell) + c_\ell \hat{z}_\ell^\ell$ which is linear in $dd^\top$

DD Construction for MIQP

Observation For a fixed support $z \in \{0,1\}^n$, denote $S = \{i : z_i = 1\}$. Then

$$g(z) \triangleq \min_{x : x \circ (1-z) = 0} \frac{1}{2} x^\top Q x + d^\top x = -\frac{1}{2} d_S^\top Q_{SS}^{-1} d_S = -\frac{1}{2} \langle (Q \circ zz^\top)^\dagger, d^\top d \rangle,$$

where $\left[(Q \circ zz^\top)^\dagger\right]_{ij} = \left[Q_{SS}^{-1}\right]_{ij}$ if $i, j \in S$ and 0 otherwise.

State Space Assume v_ℓ is one node at layer ℓ corresponding to the partial solution $z^\ell \in \{0,1\}^{\ell-1}$. Define the state of v_ℓ as

$$s_{v_\ell} = \left[Q \circ \hat{z}^\ell \left(\hat{z}^\ell \right)^\top \right]^\dagger,$$

where $\hat{z}^\ell \in \mathbb{R}^n$ is defined by $\hat{z}_i^\ell = z_i^\ell$ if $i \le \ell - 1$ and 0 otherwise.

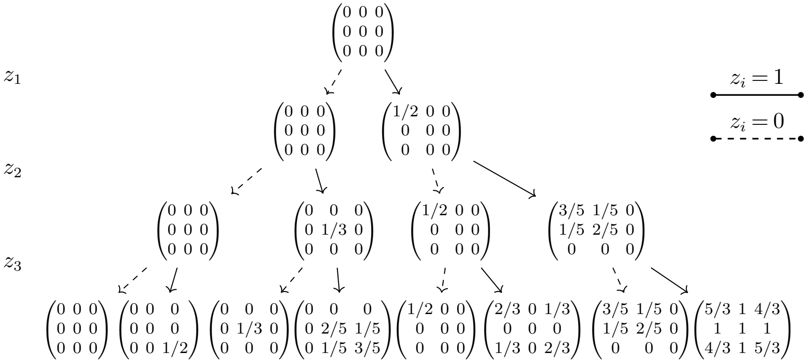
Transition Function $s_{v_{\ell+1}} - s_{v_\ell}$ can be computed using rank-one updates

Arc Length $\ell_{v_\ell v_{\ell+1}} = g(\hat{z}^{\ell+1}) - g(\hat{z}^\ell) + c_\ell \hat{z}_\ell^\ell$ which is linear in $dd^\top$

Remark: The architecture of *DD* does not depend on d $\Rightarrow$ **only need to construct DD once in the online setting**

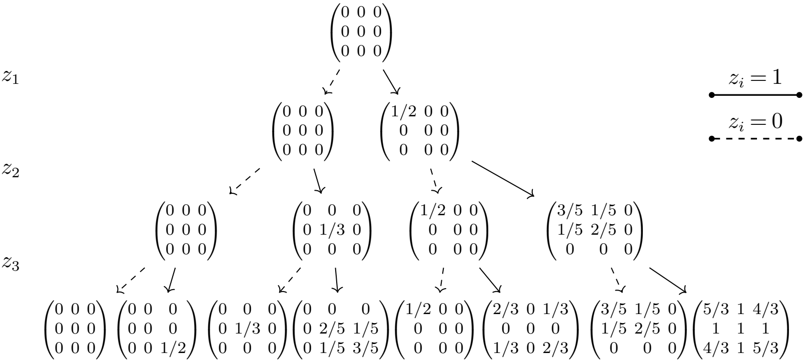
Illustrating Example

Example Consider $Q = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 2 \end{pmatrix}$. Then the decision diagram is



Illustrating Example

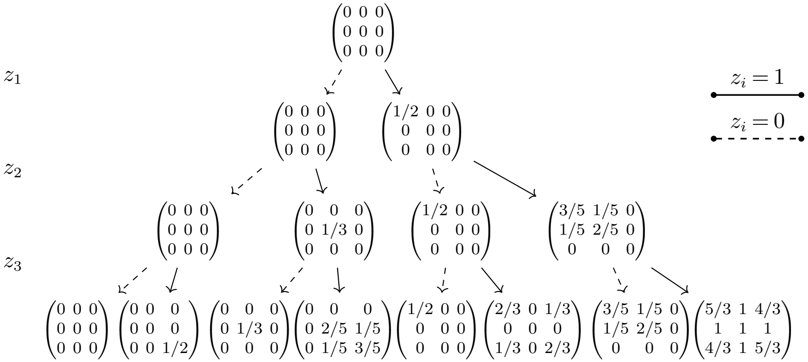
Example Consider $Q = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 2 \end{pmatrix}$. Then the decision diagram is



A second thought: we are doing enumeration...

Illustrating Example

Example Consider $Q = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 2 \end{pmatrix}$. Then the decision diagram is



A second thought: we are doing enumeration... Can we improve?

One observation

Consider $\boldsymbol{Q} = \begin{pmatrix} 5 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 5 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 5 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 5 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 5 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 5 & * \\ 0 & 0 & 0 & 0 & 0 & * & * \end{pmatrix}_{7\times 7}$, $\bar{\boldsymbol{z}}^{\ell} = \begin{pmatrix} \color{red}{1} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}$ and $\hat{\boldsymbol{z}}^{\ell} = \begin{pmatrix} \color{red}{0} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}$.

Then

$$A = \left(\begin{array}{c|ccc} & 0.00 & \mathbf{0.00008} & 0 \\ & 0.00 & \mathbf{0.00040} & 0 \\ & 0.01 & \mathbf{0.00190} & 0 \\ \bigstar & 0.05 & \mathbf{0.00909} & 0 \\ & 0.22 & \mathbf{0.04356} & 0 \\ & 0.04 & \mathbf{0.20871} & 0 \\ & 0 & \mathbf{0} & 0 \end{array} \right)_{7\times 7}, \quad B = \left(\begin{array}{c|ccc} & 0 & \mathbf{0} & 0 \\ & 0.00 & \mathbf{0.00038} & 0 \\ & 0.01 & \mathbf{0.00189} & 0 \\ \bigstar & 0.05 & \mathbf{0.00909} & 0 \\ & 0.22 & \mathbf{0.04356} & 0 \\ & 0.04 & \mathbf{0.20871} & 0 \\ & 0 & \mathbf{0} & 0 \end{array} \right)_{7\times 7},$$

where $\star$ is the submatrix unrelated to the transition/cost function, $A = \left(\boldsymbol{Q} \circ \bar{\boldsymbol{z}}^{\ell}(\bar{\boldsymbol{z}}^{\ell})^{\top}\right)^{\dagger}$, $B = \left(\boldsymbol{Q} \circ \hat{\boldsymbol{z}}^{\ell}(\hat{\boldsymbol{z}}^{\ell})^{\top}\right)^{\dagger}$.

Observation: $\max_{i,j \text{ essential}} |A_{ij} - B_{ij}| < 8 \times 10^{-5} \stackrel{\text{def}}{=} \epsilon$, i.e., the two states are essentially indistinguishable up to numerical precision $\epsilon \Rightarrow \epsilon$-exact decision diagram

ϵ-exact Decision Diagrams

Definition An ϵ-exact decision diagram is any decision diagram produced layer by layer according to the original construction rule and then merging those ϵ-indistinguishable states.

ϵ-exact Decision Diagrams

Definition An ϵ-exact decision diagram is any decision diagram produced layer by layer according to the original construction rule and then merging those ϵ-indistinguishable states.

A Fully Polynomial Time Approximation Scheme (FPTAS)

Theorem (Informal)

Given a matrix with bandwidth k, with a proper merging rule, one can construct a decision diagram $\mathcal{D}_{approx}$ such that

$$\#\text{ of arcs in } DD \le c_1 n \left(\frac{\|d\|_\infty^2 n}{\epsilon} \right)^{c_2},$$

- *where c_1 and c_2 only depend on k and the condition number of Q;*
- *ϵ is the optimality gap.*

ϵ-exact Decision Diagrams

Definition An ϵ-exact decision diagram is any decision diagram produced layer by layer according to the original construction rule and then merging those ϵ-indistinguishable states.

A Fully Polynomial Time Approximation Scheme (FPTAS)

Theorem (Informal)

Given a matrix with bandwidth k, with a proper merging rule, one can construct a decision diagram $\mathcal{D}_{approx}$ such that

$$\#\ \textit{of arcs in DD} \leq c_1 n \left(\frac{\|d\|_\infty^2 n}{\epsilon} \right)^{c_2},$$

- *where c_1 and c_2 only depend on k and the condition number of Q;*
- *ϵ is the optimality gap.*

Remark. In practical implementation, taking $\epsilon = 10^{-5}$ is sufficient to obtain exact optimal solutions within machine precision.

Agenda

More constraints...

What if we have more constraints over (x, z)?

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & , \end{aligned} \tag{MIQP}$$

More constraints...

What if we have more constraints over (x, z)?

If $Z \notin \{0,1\}^n$, delete nodes due to infeasibility in DD

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2} x^\top Q x + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z, \end{aligned} \tag{MIQP}$$

- E.g. $Z = \{z \in \{0,1\}^n : \sum_{i=1}^n z_i \leq k\}$
  The number of nodes is reduced $\Rightarrow$ ϵ-exact DD remains a FPTAS

More constraints...

What if we have more constraints over (x, z)?

If $Z \notin \{0,1\}^n$ and $P \notin \mathbb{R}^n$,

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z, \ x \in P \end{aligned} \tag{MIQP}$$

- This will destroy the exact small-bandwidth structure
  $\Rightarrow$ we cannot expect the same complexity anymore

More constraints...

What if we have more constraints over (x, z)?

If $Z \notin \{0,1\}^n$ and $P \notin \mathbb{R}^n$,

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Q x + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z, \ x \in P \end{aligned} \tag{MIQP}$$

- This will destroy the exact small-bandwidth structure
  $\Rightarrow$ we cannot expect the same complexity anymore

How to exploit the small-bandwidth structure of Q in solving (MIQP)?

More constraints...

What if we have more constraints over (x, z)?

If $Z \notin \{0,1\}^n$ and $P \notin \mathbb{R}^n$,

$$\begin{aligned} \min_{x,z}\ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.}\ & x_i(1 - z_i) = 0 \quad \forall i \in [n] \\ & z \in Z,\ x \in P \end{aligned} \tag{MIQP}$$

- This will destroy the exact small-bandwidth structure
  $\Rightarrow$ we cannot expect the same complexity anymore

How to exploit the small-bandwidth structure of Q in solving (MIQP)?

$\Rightarrow$ Convexification

Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

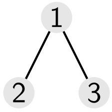
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Enumeration Branch & bound algorithm

- Solve a convex relaxation at each node of the tree



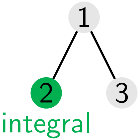
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP ⇔ enumeration + convexification

Enumeration Branch & bound algorithm

- Solve a convex relaxation at each node of the tree



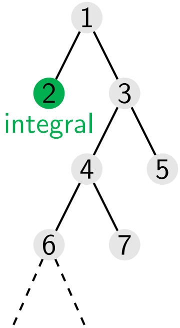
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Enumeration Branch & bound algorithm

- Solve a convex relaxation at each node of the tree
- Branch on variables with fractional value



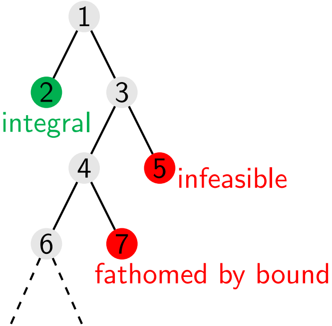
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Enumeration Branch & bound algorithm

- Solve a convex relaxation at each node of the tree
- Branch on variables with fractional value
- Prune by integrality, infeasibility and bounds



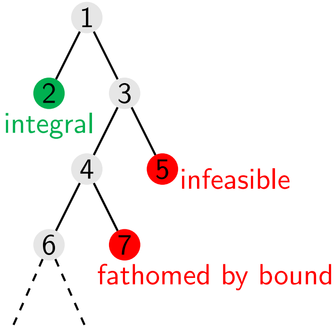
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Enumeration Branch & bound algorithm

- Solve a **convex relaxation** at each node of the tree
- Branch on variables with fractional value
- Prune by integrality, infeasibility and bounds



- Constructing strong convex relaxations is an art!

Role of Convexification in MIP

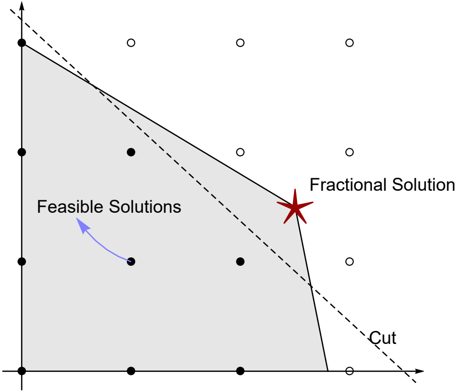
Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Convexification In mixed-integer linear optimization, convexification refers to various kinds of cutting planes

- Gomory cuts (1950s)
- Mixed-integer rounding cuts
- Flow cover cuts
- ...

Over 70-year development

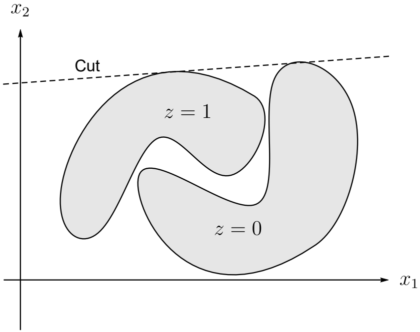


Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Convexification In mixed-integer nonlinear optimization, cutting planes could be ineffective



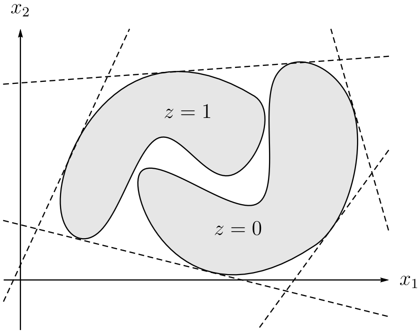
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Convexification In mixed-integer nonlinear optimization, cutting planes could be ineffective

- Need infinite number of linear cuts to ensure feasibility



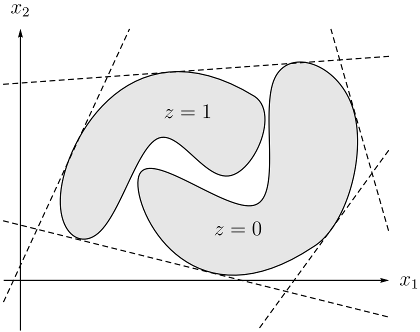
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Convexification In mixed-integer nonlinear optimization, cutting planes could be ineffective

- Need infinite number of linear cuts to ensure feasibility
- Study the convex hull of **structured** mixed-integer nonlinear sets



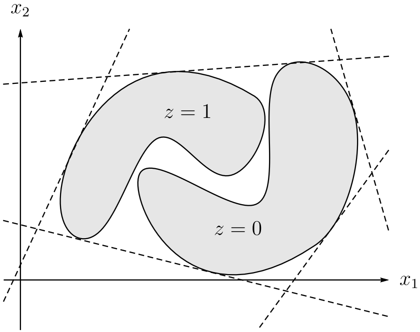
Role of Convexification in MIP

Recipe for solving a general mixed-integer program (MIP)

solving a MIP $\Leftrightarrow$ enumeration + convexification

Convexification In mixed-integer nonlinear optimization, cutting planes could be ineffective

- Need infinite number of linear cuts to ensure feasibility
- Study the convex hull of **structured** mixed-integer nonlinear sets
- Need new convexification techniques



Epigraphical reformulation

Get back.... Note that

$$
\begin{aligned}
\min_{x,z}\ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\
\text{s.t. }\ & x_i(1-z_i) = 0 \quad \forall i \in [n] \\
& z \in Z,\ x \in P
\end{aligned}
\tag{MIQP}
$$

is equivalent to

$$
\begin{aligned}
\min_{x,z}\ & t + d^\top x + c^\top z \\
\text{s.t. }\ & t \geq \frac{1}{2}x^\top Qx \\
& x_i(1-z_i) = 0, z_i \in \{0,1\} \quad \forall i \in [n] \\
& z \in Z \\
& x \in P
\end{aligned}
$$

Epigraphical reformulation

Get back.... Note that

$$\begin{aligned} \min_{x,z} \ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.} \ & x_i(1-z_i) = 0 \quad \forall i \in [n] \\ & z \in Z, \ x \in P \end{aligned} \tag{MIQP}$$

is equivalent to

$$\begin{aligned} \min_{x,z} \ & t + d^\top x + c^\top z \\ \text{s.t.} \ & \boxed{\begin{aligned} & t \geq \frac{1}{2}x^\top Qx \\ & x_i(1-z_i) = 0, z_i \in \{0,1\} \quad \forall i \in [n] \\ & z \in Z \end{aligned}} \overset{\text{def}}{=} X_{Q,Z} \\ & x \in P \end{aligned}$$

Epigraphical reformulation

Get back.... Note that

$$\begin{aligned} \min_{x,z}\ & \frac{1}{2}x^\top Qx + d^\top x + c^\top z \\ \text{s.t.}\ & x_i(1-z_i) = 0 \quad \forall i \in [n] \\ & z \in Z,\ x \in P \end{aligned} \tag{MIQP}$$

is equivalent to

$$\begin{aligned} \min_{x,z}\ & t + d^\top x + c^\top z \\ \text{s.t.}\ & \boxed{\begin{aligned} & t \geq \frac{1}{2}x^\top Qx \\ & x_i(1-z_i) = 0, z_i \in \{0,1\} \quad \forall i \in [n] \\ & z \in Z \end{aligned}} \overset{\text{def}}{=} X_{Q,Z} \\ & x \in P \end{aligned}$$

- Replace $X_{Q,Z}$ with $\text{conv}(X_{Q,Z}) \Rightarrow$ a strong convex relaxation
- $X_{Q,Z}$ doesn't involve $d \Rightarrow$ only need to compute $\text{conv}(X_{Q,Z})$ once

Convexification

Define

$$X_{Q,Z} \triangleq \left\{(t,x,z) \in \mathbb{R}^n \times \mathbb{R} \times Z : t \geq x^\top Qx,\ x_i(1-z_i) = 0\ \forall i\right\}.$$

With a DD at hand, one can show

Theorem (Convex Hull Description)

Point $(t,x,z) \in conv(X_{Q,Z})$ iff the followig **SOCP-r** *system is consistent*

$$x_0 \geq \sum_{a\in A} \frac{w_a^2}{r_a},\ x = \sum_{a\in A} u_a w_a,\ z = \sum_{a\in A:\nu_a=1} e_{\ell(a)} r_a,\ r \in P$$

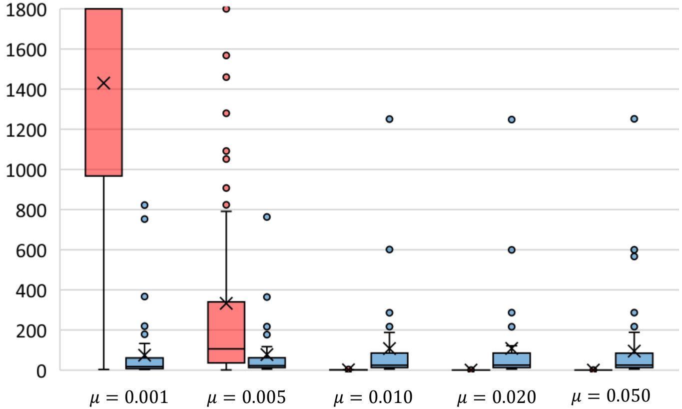
where

$$P = \left\{ r : \begin{array}{c} r \geq 0,\ \sum_{a\in A:\ell(a)=1} r_a = 1,\ \sum_{a\in A:\ell(a)=n} r_a = 1, \\ \sum_{a\in A:h_a=v} r_a = \sum_{a\in A:t_a=v} r_a\ \forall v \in N : \ell(v) \leq n \end{array} \right\}$$

is the path polytope associated with the decision diagram.

Agenda

Computational Results in Offline Settings



Distribution of runtimes of Mosek (red) vs Decision diagram (blue) for $n = 200$ as a function sparsity parameter μ. Each boxplot represents an average over 5 different signals $\boldsymbol{y}$ with n $k \in \{2, 3\}$ and $\lambda \in \{0.25, 0.50, 1.0, 2.00, 5.00\}$.

Computational Results in Online Settings

Online instances, each one requiring the sequential solution of 6,823 MIOs (31) with $n = 200$ (corresponding, for each point, to the most recent 200 observations).

$\boldsymbol{k}$	$\boldsymbol{\lambda}$	**Setup time**		**Online time**	
		$\lvert\boldsymbol{A}\rvert$	**time_dd (s)**	**time_sp(s)**	**time_total(s)**
2	0.25	10,965	7	0.001	7
	0.5	16,749	11	0.002	11
	1.0	30,963	24	0.004	30
	2.0	51,923	32	0.006	43
	5.0	88,491	62	0.013	88
3	0.25	56,789	40	0.007	48
	0.5	107,591	81	0.016	107
	1.0	233,917	184	0.035	239
	2.0	478,889	409	0.079	539
	5.0	963,643	864	0.185	1,261

Take Home Message

Summary

- Develop a real-time solution method for solving MIQPs with small bandwidth using decision diagrams
- Construct approximate DDs whose size is polynomial in the number of decision variables and $\frac{1}{\text{OPT GAP}} \Rightarrow$ FPTAS
- Establish the convex hull results for the mixed-integer epigraph using constructed DD
- Amazing performance in practice!

Take Home Message

Summary

- Develop a real-time solution method for solving MIQPs with small bandwidth using decision diagrams
- Construct approximate DDs whose size is polynomial in the number of decision variables and $\frac{1}{\text{OPT GAP}} \Rightarrow$ FPTAS
- Establish the convex hull results for the mixed-integer epigraph using constructed DD
- Amazing performance in practice!

Our paper is available at: `https://arxiv.org/pdf/2405.03051`

Thanks for your listening!

Dunn, J., Runge, R., and Snyder, M. (2018). Wearables and the medical revolution. Personalized medicine, 15(5):429–448.

Yan, H., Paynabar, K., and Shi, J. (2017). Anomaly detection in images with smooth background via smooth-sparse decomposition. Technometrics, 59(1):102–114.