Exact resolution of the sparse spectral unmixing problem

Application of branch-and-bound algorithm

M.Latif[1], S.Bourguignon[1], L.Drumetz[2] & J.Ninin[3]

[1]LS2N / Centrale Nantes, [2]Lab-STICC / IMT Atlantique, [3]Lab-STICC / ENSTA-Bretagne.

July 6, 2022

The short λ's resume

2015 - 2018 Bachelor's degree in computer science and mathematics (Université de Nantes).

2018 - 2020 Master's degree in computer science (Université de Nantes & Université Libre de Bruxelles);
Major in Optimization in Operations Research;
Master's thesis : Branch-and-bound design for ℓ_0-*norm* optimization, application to the spectral unmixing problem;
Advisors : S.Bourguignon, R.Ben Mhenni

2020 - 2021 Project engineer (LS2N): algorithm implementation and scientific article redaction.

2021 - 2024 PhD student : Tomographic image reconstruction for a new uncommon 3-photons imaging system (CHU, Centrale Nantes, LS2N);
Advisors : Simon Stute, Jérôme Idier & Thomas Carlier

Table of contents

Table of contents

Hyperspectral imaging

Let us consider a scene,

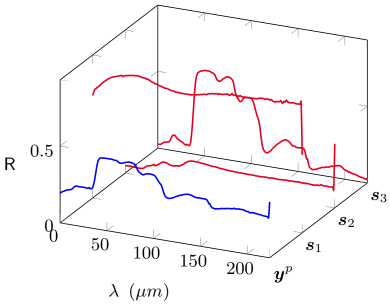
- Extracts information about the **present electromagnetic spectra** over a **large range of spectral bands**;
- A pixel $\Rightarrow$ the observed reflectance spectrum is a **mixture** formed by **different contributions** of the materials **spectral signatures**.

Spectral unmixing problem (SU)

- A **blind source separation** problem i.e. estimate both **spectral signatures** - aka. *atoms* or *endmembers* - and **their proportions** aka. *abundances*;
- **Supervised SU :** the mixture is searched in a known **dictionary** of reference spectra.

Assumption : Linear Mixing Model (LMM)

The **observed mixture** for a pixel can be expressed as the **linear combination of endmembers weighted by the fractional** abundances of each atom.



$\Rightarrow$ For a given pixel observed on N_λ spectral bands :

$$\boldsymbol{y} = \mathbf{S}\,\boldsymbol{a} + \boldsymbol{\varepsilon} \;\Leftrightarrow\; \boldsymbol{y} = \sum_{q \in [\![1,Q]\!]} a_q\,\boldsymbol{s}_q + \boldsymbol{\varepsilon}$$

where

- $\boldsymbol{y} \in \mathbb{R}^{N_\lambda}$: the reflectance spectrum;
- $\mathbf{S} \in \mathbb{R}^{N_\lambda \times Q}$: the dictionary of Q atoms;
- $\boldsymbol{a} \in \mathbb{R}^{Q}$: the abundance vector;
- $\boldsymbol{\varepsilon} \in \mathbb{R}^{N_\lambda}$: additional noise vector;
- $\boldsymbol{s}_q, a_q$: the q-th atom and its abundance.

Sum-to-one & nonnegativity constraints

$$\boldsymbol{y} = \sum_{q \in [\![1,Q]\!]} a_q \, \boldsymbol{s}_q + \boldsymbol{\varepsilon} \qquad \text{(LMM)}$$

Motivation :

- To express these abundances $\boldsymbol{a}$ as percentages;
- An active endmember $\Rightarrow$ a nonnegative explanatory contribution to the observed spectrum;
- The reflectance spectrum is only a part of the emitted light (absorption by materials) $\sim$ Normalization of the observed abundances.

Abundance nonnegativity constraint

$$\boldsymbol{a} \geq 0 \iff a_q \geq 0 \quad \forall q \in [\![1,Q]\!]$$

Abundance sum-to-one constraint

$$\mathbf{1}_Q^\intercal \, \boldsymbol{a} = 1 \iff \textstyle\sum_{q \in [\![1,Q]\!]} a_q = 1$$

Remark :

- This SU formulation is well identified in the literature as **Fully Constrained Least Squares** problems [Heinz et al., 2001] which naturally produce solutions where **some abundances are zero**.
- in **complex problems**, the estimated abundances are **spread over a substantial number** of components.

Sparsity constraint

Motivations

- **Physical constraint** : the observed scene is composed of a **limited number** of materials;
- The FCLS formulation may **fail in locating** the true endmembers;
- **Enforcing the number of nonzero abundances** may merely **ensure a correct mixture interpretation**.

The support of $\boldsymbol{a}$

$$\mathbf{supp}(\boldsymbol{a}) := \{q \in [\![1, Q]\!] \,|\, a_q \neq 0\}$$

The counting function : the ℓ_0-"norm"

$$\|\,\boldsymbol{a}\,\|_0 := \mathbf{Card}\,(\mathbf{supp}(\boldsymbol{a}))$$

K-sparse solution

$$\boldsymbol{a} \in \mathbb{R}^Q \quad \text{s.t.} \quad \|\,\boldsymbol{a}\,\|_0 \leq K, \;\; K \in \mathbb{Z}_+^*$$

Sparse spectral unmixing problem (ℓ_0-SU)

$$\min_{\boldsymbol{a}\in[0,1]^Q} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}\,\boldsymbol{a} \right\|_2^2 \qquad \text{s.t.} \quad \|\,\boldsymbol{a}\,\|_0 \leq K, \quad \mathbf{1}_Q^\intercal\,\boldsymbol{a} = 1 \qquad (\mathcal{P}_{2/0})$$

where $K \in \mathbb{Z}_+^*$ the sparsity coefficient **set *a priori***.

It's a complex problem

- ℓ_0-"norm" is not differentiable, not continuous $\Rightarrow$ $\mathcal{P}_{2/0}$ is **not convex**;
- ℓ_0-"norm" minimization problems are $\mathcal{NP}$-Hard [Natarajan, 1995];
- Can be **only considered** for problems with a **limited amount of atoms**.

But in practice

- The **number of active atoms** in a mixture is relatively **small** e.g. $K \in [\![1, 6]\!]$;
- The **dictionary** is typically composed of $10 \sim 500$ spectral signatures;
- The **sum-to-one & nonnegativity constraints restrict** the space of feasible solutions.

A short state of the art

Approximation: greedy algorithms using heuristics, convex or nonconvex relaxation [Tropp and Wright, 2010]

- **Reduce** the complexity of an **exhaustive exploration**;
- **Rapidly** provide sparse solution;
- The obtained results are **often suboptimal**.

Standard ℓ_0 problems using MIP techniques:
[Bourguignon et al., 2016], [Bertsimas et al., 2016];

Dedicated branch-and-bound algorithm to tackle ℓ_0 problems:
[Bienstock, 1999],[Bertsimas and Shioda, 2009], [Ben Mhenni, 2020], [Hazimeh et al., 2021];

Our approach

Implementing an **exact optimization algorithm** for the ℓ_0-SU

Exact optimisation of the ℓ_0-SU as a MIP [Ben Mhenni et al., 2018]

$$\min_{\boldsymbol{a}\in[0,1]^Q} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}\,\boldsymbol{a} \right\|_2^2 \qquad \text{s.t.} \quad \|\,\boldsymbol{a}\,\|_0 \le K, \quad \mathbf{1}_Q^\intercal\,\boldsymbol{a} = 1 \tag{$\mathcal{P}_{2/0}$}$$

MIP formulation

Introduce a binary decision variables: $\boldsymbol{b} \in \{0,1\}^Q$ s.t. $a_q = 0 \Leftrightarrow b_q = 0 \;\; \forall q \in [\![1, Q]\!]$;

Link $\boldsymbol{a}$ and $\boldsymbol{b}$: $|\,\boldsymbol{a}\,| \le M\,\boldsymbol{b}$ with $M \in \mathbb{R}_+^*$ aka. *bigM assumption*;

Trivially with $\boldsymbol{a} \in [0,1]$: $M = 1$ and $\boldsymbol{a} \le \boldsymbol{b}$

Rewrite the ℓ_0-"norm": $\|\boldsymbol{a}\|_0 = \sum_{q\in[\![1,Q]\!]} b_q$

$$\min_{\substack{\boldsymbol{a}\in[0,1]^Q \\ \boldsymbol{b}\in\{0,1\}^Q}} \quad \frac{1}{2}\left\|\boldsymbol{y} - \mathbf{S}\boldsymbol{a}\right\|_2^2 \qquad \text{s.t.} \quad \sum_{q\in[\![1,Q]\!]} b_q \le K, \quad \mathbf{1}_Q^\intercal\,\boldsymbol{a} = 1, \quad \boldsymbol{a} \le \boldsymbol{b} \tag{$\mathcal{P}_{2/0}$}$$

$\Rightarrow$ Just use IBM Cplex solver . . .

Exact optimisation of the ℓ_0-SU - Our approach

Our approach

- Implementing a **dedicated** and **exact optimization algorithm** for the ℓ_0-SU
  $\Rightarrow$ Branch-and-bound algorithm;
- Try to be ~~*harder*~~, *better*, *faster*, *stronger* than a commercial solver [Ben Mhenni, 2020];
- Provide an **open-source** software to the SU community.

Table of contents

Branch-and-bound algorithm [Land and Doig, 1960]

OR's Mantra

When an exhaustive search of all feasible solutions is inconceivable, take a break and implement a branch-and-bound.

Main principle

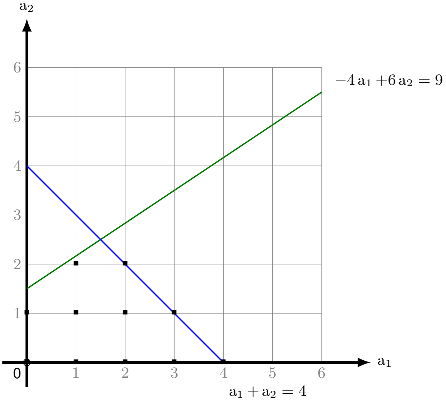
- Divide & conquer approach;
- Use of bounds on the optimal value to avoid the exploration of some regions of the search space; for a minimization problem :

$$\underline{z} \leq z^* \leq \overline{z}$$

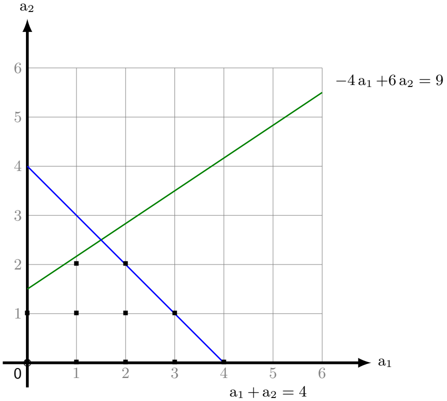
  The dual bound $\underline{z} \in \mathbb{R}$: a **lower bound** computed by a **relaxation**, often **easier to compute**;
  The primal bound $\overline{z} \in \mathbb{R}$: an **upper bound** given by any **feasible** solution.
- Build an **exploration tree** by repeating two steps: the **Branch** & the **Bound** operations.

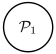
Let us consider:

$$(\mathcal{P}) : \min \left\{ f(\boldsymbol{a}) = \mathrm{a}_1 - 2\,\mathrm{a}_2 \;\middle|\; -4\,\mathrm{a}_1 + 6\,\mathrm{a}_2 \leq 9;\;\; \mathrm{a}_1 + \mathrm{a}_2 \leq 4;\;\; \boldsymbol{a} \in \mathbb{Z}_+^2 \right\}$$



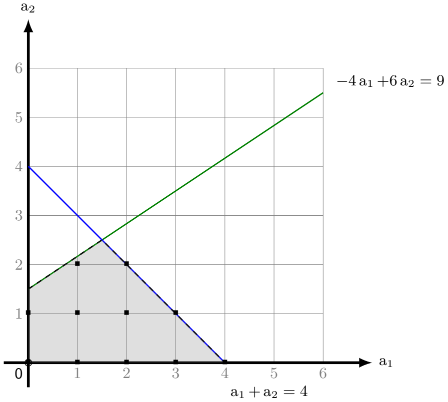
Our objective : find $\hat{\boldsymbol{a}} := \textbf{Argmin}(\mathcal{P})$

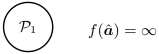




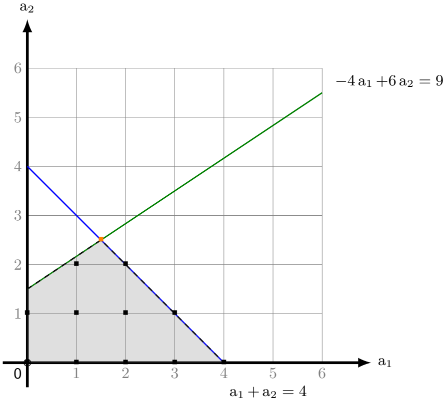
$f(\hat{\boldsymbol{a}}) = \infty$

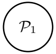
$$(\mathcal{P}_1) : \min \left\{ f(\boldsymbol{a}) = \mathrm{a}_1 - 2\,\mathrm{a}_2 \;\middle|\; -4\,\mathrm{a}_1 + 6\,\mathrm{a}_2 \leq 9;\;\; \mathrm{a}_1 + \mathrm{a}_2 \leq 4;\;\; \boldsymbol{a} \in \mathbb{R}_+^2 \right\}$$





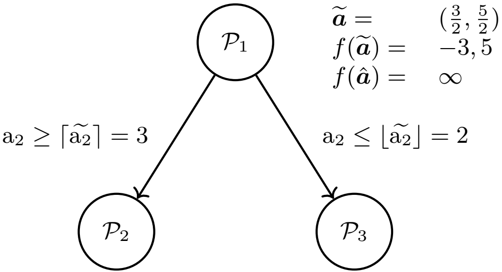
$$(\mathcal{P}_1) : \min \left\{ f(\boldsymbol{a}) = \mathrm{a}_1 - 2\,\mathrm{a}_2 \;\middle|\; -4\,\mathrm{a}_1 + 6\,\mathrm{a}_2 \leq 9;\;\; \mathrm{a}_1 + \mathrm{a}_2 \leq 4;\;\; \boldsymbol{a} \in \mathbb{R}_+^2 \right\}$$



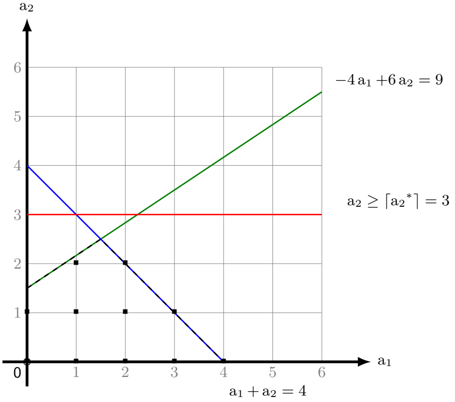


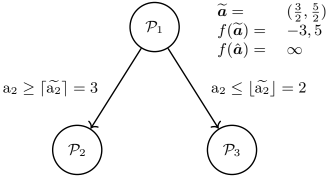
$\widetilde{\boldsymbol{a}} = \;(\frac{3}{2}, \frac{5}{2})$
$f(\widetilde{\boldsymbol{a}}) = \;-3,5$
$f(\hat{\boldsymbol{a}}) = \;\infty$

Branch : Use the optimum value to define inequalities and create a b-partition of the search space (exhaustive but not necessarily mutually exclusive);
e.g. Most infeasible branching $\Leftrightarrow$ most fractional component



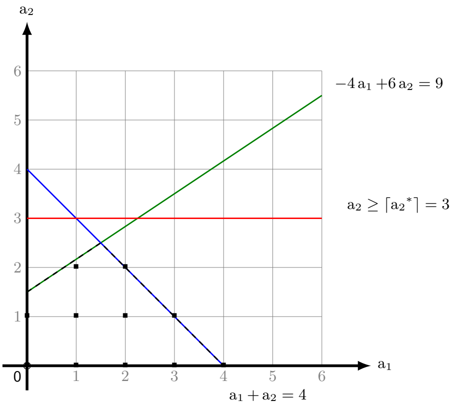
$$\mathcal{P}_2 := \mathcal{P}_1 \cap \{a_2 \geq 3\}$$

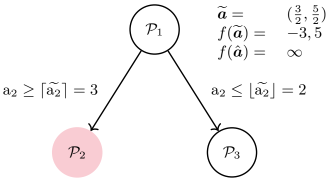




Branch-and-bound algorithm by the example

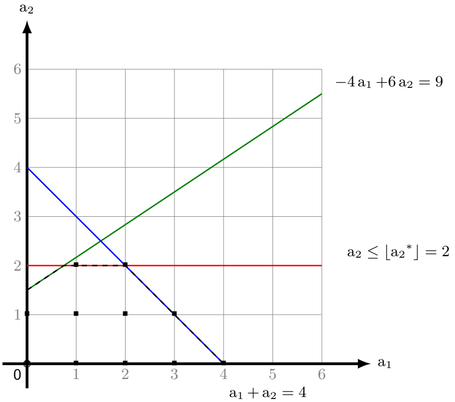
$$\mathcal{P}_2 := \mathcal{P}_1 \cap \{a_2 \geq 3\}$$

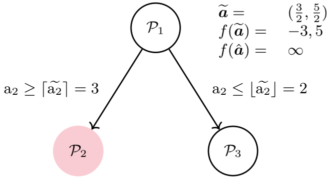




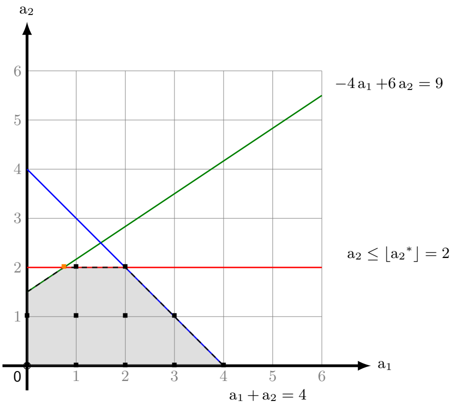
$\Rightarrow$ Pruning $\mathcal{P}_2$ by **nonfeasibility** !

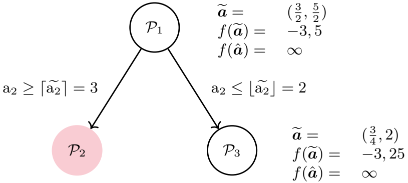
$$\mathcal{P}_3 := \mathcal{P}_1 \cap \{a_2 \leq 2\}$$





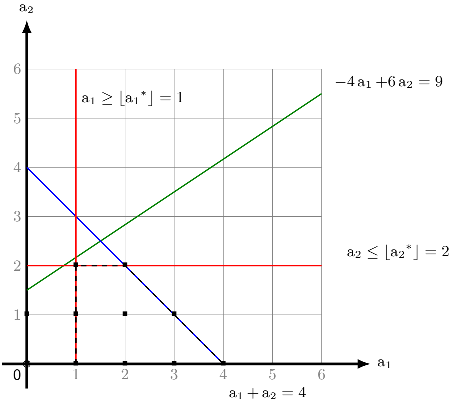
$$\mathcal{P}_3 := \mathcal{P}_1 \cap \{a_2 \le 2\}$$

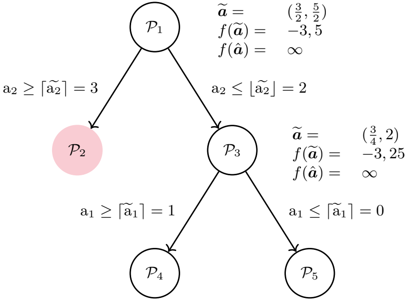




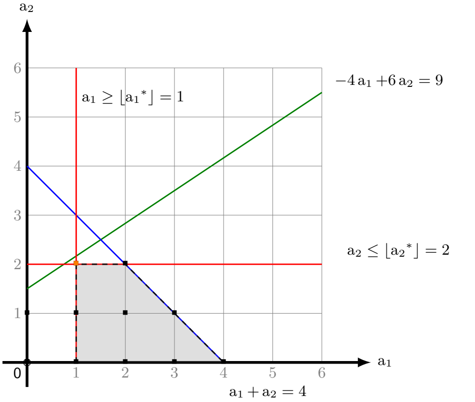
$\Rightarrow$ Nothing can be concluded: **Branch** on $\mathcal{P}_3$!

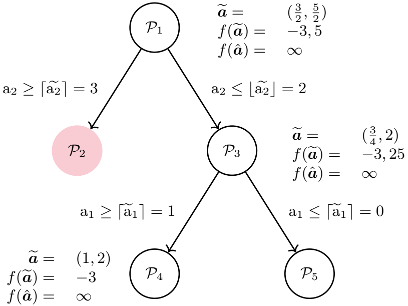
$$\mathcal{P}_4 := \mathcal{P}_1 \cap \{a_2 \leq 2\} \cap \{a_1 \geq 1\}$$





$$\mathcal{P}_4 := \mathcal{P}_1 \cap \{a_2 \leq 2\} \cap \{a_1 \geq 1\}$$

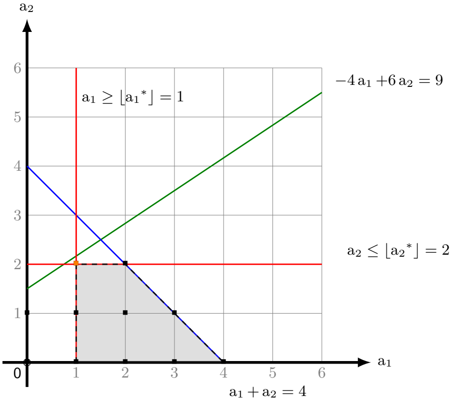


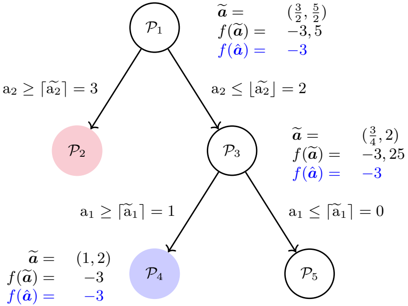


$\Rightarrow$ Prune $\mathcal{P}_4$ by **optimality** i.e. $\exists\, \widetilde{\boldsymbol{a}}$ $\mathbb{Z}_+$-feasible s.t. $f(\widetilde{\boldsymbol{a}}) < f(\hat{\boldsymbol{a}})$;
$\Rightarrow$ Update the global upper bound $f(\hat{\boldsymbol{a}})$

Branch-and-bound algorithm by the example

$$\mathcal{P}_4 := \mathcal{P}_1 \cap \{a_2 \leq 2\} \cap \{a_1 \geq 1\}$$



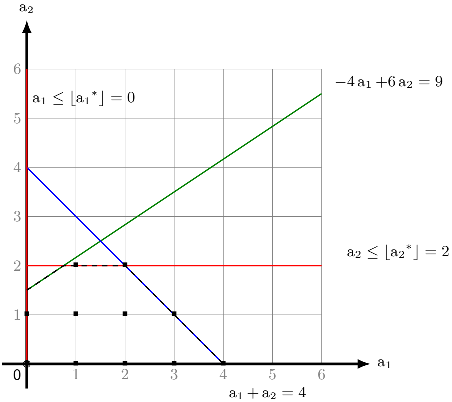


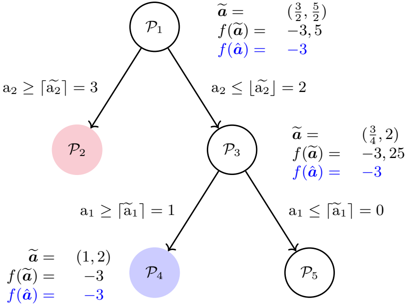
$\Rightarrow$ Prune $\mathcal{P}_4$ by **optimality** i.e. $\exists \widetilde{\boldsymbol{a}}$ $\mathbb{Z}_+$-feasible s.t. $f(\widetilde{\boldsymbol{a}}) < f(\hat{\boldsymbol{a}})$;

$\Rightarrow$ Update the global upper bound $f(\hat{\boldsymbol{a}})$

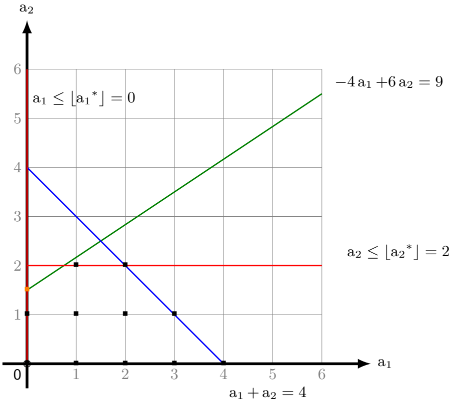
Branch-and-bound algorithm by the example

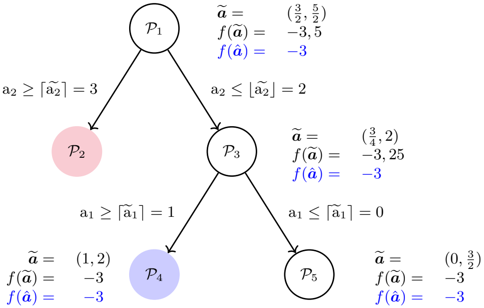
$$\mathcal{P}_5 := \mathcal{P}_1 \cap \{a_2 \leq 2\} \cap \{a_1 \leq 0\}$$



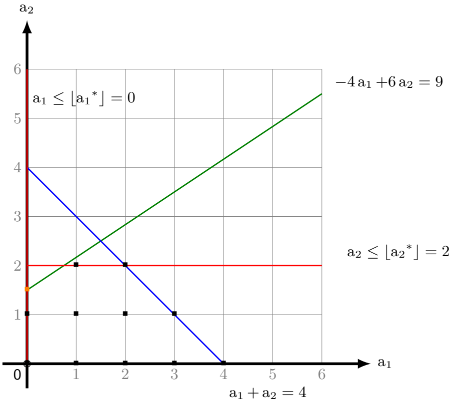


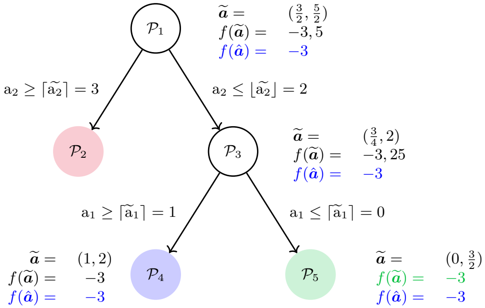
$$\mathcal{P}_5 := \mathcal{P}_1 \cap \{a_2 \le 2\} \cap \{a_1 \le 0\}$$





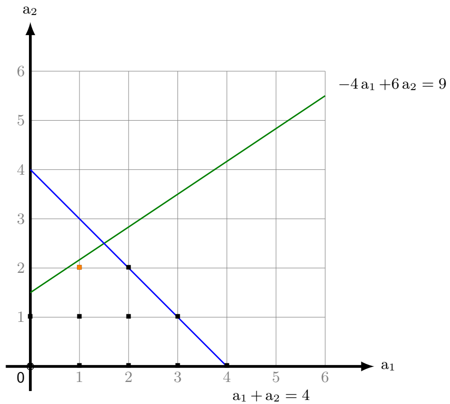
$$\mathcal{P}_5 := \mathcal{P}_1 \cap \{a_2 \le 2\} \cap \{a_1 \le 0\}$$

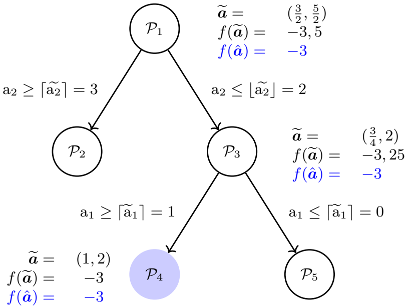




$\Rightarrow$ Prune $\mathcal{P}_5$ by **dominance** i.e. $\forall \widetilde{a}, f(\widetilde{a}) \ge f(\hat{a})$;

$$(\mathcal{P}) : \min \left\{ f(\boldsymbol{a}) = \mathrm{a}_1 - 2\,\mathrm{a}_2 \;\middle|\; -4\,\mathrm{a}_1 + 6\,\mathrm{a}_2 \leq 9;\;\; \mathrm{a}_1 + \mathrm{a}_2 \leq 4;\;\; \boldsymbol{a} \in \mathbb{Z}_+^2 \right\}$$





$\Rightarrow$ **Founded** and **certified** optimality : $\hat{\boldsymbol{a}} = (1, 2)$, $f(\hat{\boldsymbol{a}}) = -3$.

Branch-and-bound algorithm [Land and Doig, 1960]

Computational complexity: $\mathcal{O}(Tb^Q)$ with

b: *branching factor*

T: a fixed bound on time needed to explore the search (sub)spaces;

Q: the length of the longest path from the root to a leaf.

How to tune this algorithm ?

Branching strategy:

- Which value for the $b \in \mathbb{N}^*$?
- How to choose the branching variable a_i ?

Pruning rules:

- How to compute the lower bounds ?
- Add other valid inequality, cutting planes, dominance rules, ...

Search strategy:

- How to explore the tree ? e.g. DFS, BFS, ...

Table of contents

$$\min_{\boldsymbol{a}\in[0,1]^Q} \quad \frac{1}{2}\left\|\, \boldsymbol{y} - \mathbf{S}\,\boldsymbol{a}\,\right\|_2^2 \quad \text{s.t.} \quad \|\,\boldsymbol{a}\,\|_0 \leq K, \quad \mathbf{1}_Q^\intercal\,\boldsymbol{a} = 1 \qquad (\mathcal{P}_{2/0})$$

The branch-and-bound will solve many subproblems on different subspaces :

Set notations

Let $\mathbb{S} := [\![1, Q]\!]$,

- The discarded variables: $\mathcal{Z} := \{q \in \mathbb{S} \,|\, a_q \notin \mathbf{supp}(\boldsymbol{a})\} \subset \mathbb{S}$;
- The remaining variables on $[0, 1]$: $\overline{\mathcal{Z}} := \mathbb{S} \setminus \mathcal{Z}$ with $\bar{n} := \mathbf{Card}(\overline{\mathcal{Z}})$
- The choosen variables: $\mathcal{C} \subset \overline{\mathcal{Z}}$

The **general formulation** of the subproblems :

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}}\in[0,1]^{\bar{n}}} \quad \frac{1}{2}\left\|\, \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}}\boldsymbol{a}_{\overline{\mathcal{Z}}}\right\|_2^2 \quad \text{s.t.} \quad \|\boldsymbol{a}_{\overline{\mathcal{Z}}}\|_0 \leq \widehat{K} - \mathbf{Card}(\mathcal{C}), \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$

ℓ_0-*norm* convex relaxation:

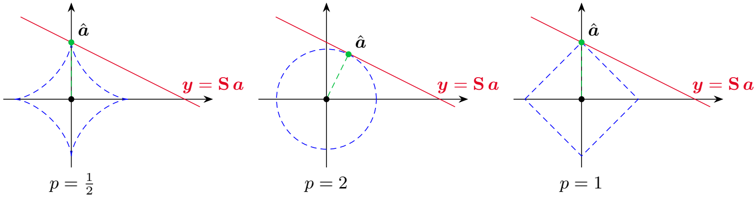


Figure: ℓ_p-norm $\forall p \in \{\frac{1}{2}, 2, 1\}$

Which one to choose?

Constraint : preserve the sparsity of the solutions;

- ℓ_p-*norm* with $p \in]0,1[$: sparse solution **but** nonconvex norm **and** we need to **ensure the optimality** of the solution
- ℓ_2-norm: nonsparse solution **but** convex norm;
- ℓ_1-norm: sparse solution **and** convex norm;

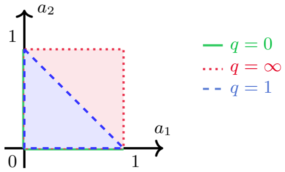
ℓ_0-*norm* convex relaxation:

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \|\boldsymbol{a}_{\overline{\mathcal{Z}}}\|_0 \leq K - \mathbf{Card}(\mathcal{C}), \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\intercal} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$

Rewrite the sparsity constraint:

Using $\mathcal{B}_p^K := \left\{ \boldsymbol{a} \in \mathbb{R}^Q \mid \| \boldsymbol{a} \|_p \leq K \right\}$, $K \in \mathbb{Z}_+^*$: $\quad \boldsymbol{a} \in \left([0,1]^{\overline{n}} \cap \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})} \right)$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\intercal} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$



Property : the ℓ_1 relaxation

Given $K \in \mathbb{N}^*$ and under bound constraints:

$$\mathbf{conv}\left([0,1]^Q \cap \mathcal{B}_0^K\right) = \left([0,1]^Q \cap \mathcal{B}_1^K\right)$$

ℓ_0-*norm* convex relaxation:

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\mathsf{T}} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_1^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\mathsf{T}} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/1})$$

Beware of sparsity constraint !!

When $K > \mathbf{Card}(\mathcal{C})$, the relaxed sparsity constraint is **dominated** by the sum-to-one constraint; it can be **withdrawn** of $\mathcal{P}_{2/1}$.

$\rightsquigarrow$ $\mathbf{1}_{\overline{\mathcal{Z}}}^{\mathsf{T}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \leq K - \mathbf{Card}(\mathcal{C}) \ \wedge \ K > \mathbf{Card}(\mathcal{C}) \implies K - \mathbf{Card}(\mathcal{C}) > 1 \ \wedge \ \mathbf{1}_{\overline{\mathcal{Z}}}^{\mathsf{T}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \leq 1$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \cancel{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_1^{K - \mathbf{Card}(\mathcal{C})}}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\mathsf{T}} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/1})$$

$\Rightarrow$ Just another FCLS formulation.

ℓ_0-"norm" subproblem:

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\intercal} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \tag{$\mathcal{P}_{2/0}$}$$

ℓ_1-"norm" subproblem:

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}}^{\intercal} \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \tag{$\mathcal{P}_{2/1}$}$$

Main idea

To select an atom to add in the support of $\mathcal{P}_{2/0}$:

- Compute $\hat{\boldsymbol{a}} := \mathbf{Argmin}(\mathcal{P}_{2/1})$
- Choose the q-th endmember

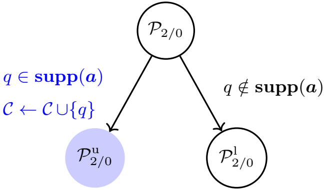
$$q := \mathbf{argmax}_{i \in \overline{\mathcal{Z}} \setminus \mathcal{C}} (\hat{a}_i)$$

- Apply a binary branching strategy on the q-th endmember, i.e. $\forall\, q \in \overline{\mathcal{Z}} \setminus \mathcal{C}$

$$q \in \mathbf{supp}(\boldsymbol{a}) \quad \textbf{or} \quad q \notin \mathbf{supp}(\boldsymbol{a})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \hat{\boldsymbol{a}}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/1})$$



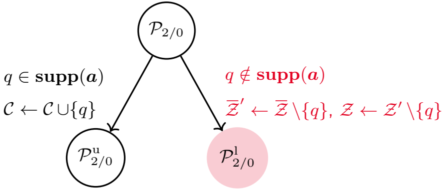
$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - (\mathbf{Card}(\mathcal{C})+1)}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0}^{\mathrm{u}})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\overline{n}}} \quad \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \hat{\boldsymbol{a}}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/1}^{\mathrm{u}})$$

$\Longrightarrow$ Same relaxations, **Nothing to do**.

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\bar{n}}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \boldsymbol{a}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/0})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}} \in [0,1]^{\bar{n}}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}} \, \boldsymbol{a}_{\overline{\mathcal{Z}}} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}}^\intercal \, \hat{\boldsymbol{a}}_{\overline{\mathcal{Z}}} = 1 \qquad (\mathcal{P}_{2/1})$$

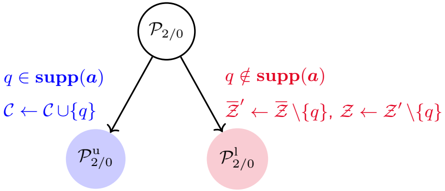


$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}'} \in [0,1]^{\bar{n}'}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}'} \boldsymbol{a}_{\overline{\mathcal{Z}}'} \right\|_2^2 \quad \text{s.t.} \quad \boldsymbol{a}_{\overline{\mathcal{Z}}'} \in \mathcal{B}_0^{K - \mathbf{Card}(\mathcal{C})}, \quad \mathbf{1}_{\overline{\mathcal{Z}}'}^\intercal \boldsymbol{a}_{\overline{\mathcal{Z}}'} = 1 \qquad (\mathcal{P}_{2/0}^{\mathrm{l}})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}'} \in [0,1]^{\bar{n}'}} \frac{1}{2}\left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}'} \, \boldsymbol{a}_{\overline{\mathcal{Z}}'} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}'}^\intercal \, \boldsymbol{a}_{\overline{\mathcal{Z}}'} = 1 \qquad (\mathcal{P}_{2/1}^{\mathrm{l}})$$

$\Longrightarrow$ Different relaxations, **A new FCLS problem**.

Additional assumption: $\hat{\boldsymbol{a}} := \mathbf{Argmin}(\mathcal{P}_{2/1})$ and $\| \hat{\boldsymbol{a}} \|_0 = K - 1$



$$\min_{\boldsymbol{a}_{\mathcal{C}} \in [0,1]^K} \frac{1}{2} \left\| \boldsymbol{y} - \mathbf{S}_{\mathcal{C}} \boldsymbol{a}_{\mathcal{C}} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\mathcal{C}}^{\intercal} \boldsymbol{a}_{\mathcal{C}} = 1 \qquad (\mathcal{P}^{\mathrm{u}}_{2/1})$$

$$\min_{\boldsymbol{a}_{\overline{\mathcal{Z}}'} \in [0,1]^{\overline{n}'}} \frac{1}{2} \left\| \boldsymbol{y} - \mathbf{S}_{\overline{\mathcal{Z}}'} \boldsymbol{a}_{\overline{\mathcal{Z}}'} \right\|_2^2 \quad \text{s.t.} \quad \mathbf{1}_{\overline{\mathcal{Z}}'}^{\intercal} \boldsymbol{a}_{\overline{\mathcal{Z}}'} = 1 \qquad (\mathcal{P}^{\mathrm{l}}_{2/1})$$

- $\mathcal{P}^{\mathrm{l}}_{2/1}$: to gather the best subset of atoms in the search subspace and get **feasible nonsparse solutions** i.e. **lower** bounds;
- $\mathcal{P}^{\mathrm{u}}_{2/1}$: to fit the optimal abundances of the selected $\mathbf{supp}(\boldsymbol{a})$ and get K**-sparse feasible solutions** i.e. **upper** bounds.

The dedicated branch-and-bound

Lower bound evaluations:

- Right child:
  $\Rightarrow$ Solve $\mathcal{P}^{l}_{2/1}$ to get lower bound,
  $\Rightarrow$ try to fathomed the node;
- Left child $\wedge$ $K = \mathbf{Card}(\mathcal{C})$:
  $\Rightarrow$ Solve $\mathcal{P}^{u}_{2/1}$ to get upper bound,
  $\Rightarrow$ fathomed the node;
- Left child $\wedge$ $K > \mathbf{Card}(\mathcal{C})$:
  $\Rightarrow$ Just branch on $q \in \overline{\mathcal{Z}} \setminus \mathcal{C}$.

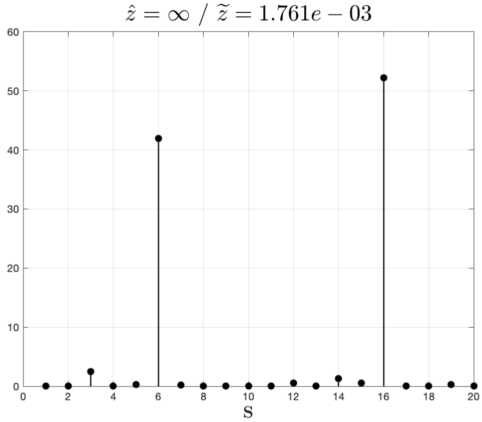
Search strategy:

DFS: to quickly find $\widehat{K}$-sparse solutions

Algorithme 1 : Dedicated branch-and-bound algorithm

Data : $\widehat{K}$, $\mathbf{S}$, $\boldsymbol{y}$,$\mathbb{S}$
Result : $(\hat{\boldsymbol{a}}, \hat{z})$
Initialization :
$\overline{\mathcal{Z}} \leftarrow \mathbb{S}$, $\mathcal{Z} \leftarrow \emptyset$, $\mathcal{C} \leftarrow \emptyset$, $n \leftarrow 0$, $L \leftarrow \emptyset$
$\hat{\boldsymbol{a}} \leftarrow \mathbf{0}_Q$, $\hat{z} \leftarrow \| \boldsymbol{y} \|_2^2$
push $\left(L, (\mathcal{C}, \overline{\mathcal{Z}}, \mathcal{Z}, n)\right)$
Main loop :
while $(L \neq \emptyset)$ **do**
  $(\mathcal{C}, \overline{\mathcal{Z}}, \mathcal{Z}, n) \leftarrow \mathbf{pop}(L)$
  $\text{terminal} \leftarrow \text{false}$
  Node evaluation procedure :
  $(\tilde{\boldsymbol{a}}, \tilde{z}) \leftarrow \mathbf{solve}(\mathcal{P}^f_{\text{n}}, \mathcal{C}, \overline{\mathcal{Z}}, \mathcal{Z})\ \forall f \in \{u, l, b\}$
  Bound procedure :
  if $(\tilde{z} \geq \hat{z})$ **then** // Fathomed by dominance
    $\text{terminal} \leftarrow \text{true}$
  else if $(\tilde{z} < \hat{z})$ **then** // Fathomed by optimality
    $\text{terminal} \leftarrow \text{true}$, $\hat{\boldsymbol{a}} \leftarrow \tilde{\boldsymbol{a}}$, $\hat{z} \leftarrow \tilde{z}$
  end
  Branch procedure :
  if $(\neg\, \textit{terminal} \wedge \textbf{\textit{Card}}(\bar{\mathbb{S}}) \neq \emptyset)$ **then**
    $q \leftarrow \mathbf{argmax}_{i \in \bar{\mathbb{S}}}(\hat{a}_i \,|\, \hat{a}_i > 0)$
    // Insert Right and Left children in L
    push $\left(L, \left(\mathcal{C}, \mathcal{Z} \cup \{q\}, \overline{\mathcal{Z}} \setminus \{q\}, 2n+2\right)\right)$
    push $\left(L, \left(\mathcal{C} \cup \{q\}, \mathcal{Z}, \overline{\mathcal{Z}} \setminus \{q\}, 2n+1\right)\right)$
  end
end

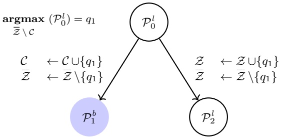
Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$

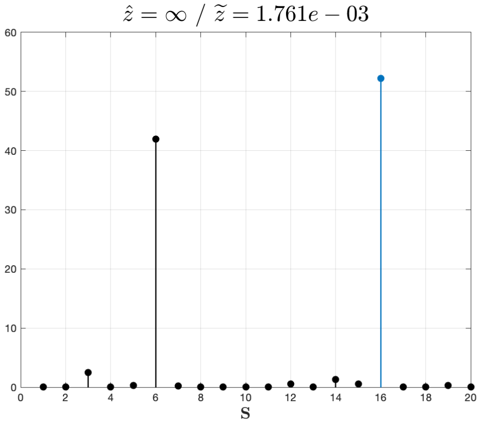


$\mathcal{C} = \emptyset$

The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$

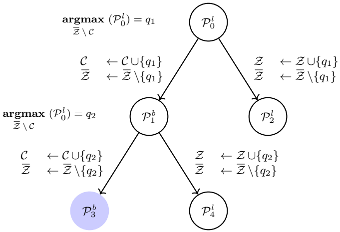


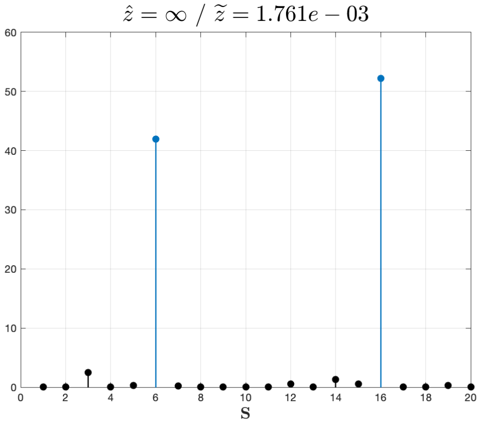


$\mathcal{C} = \{16\}$

The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$

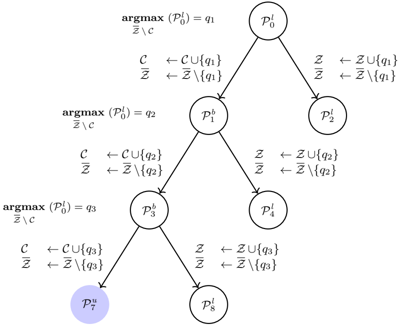


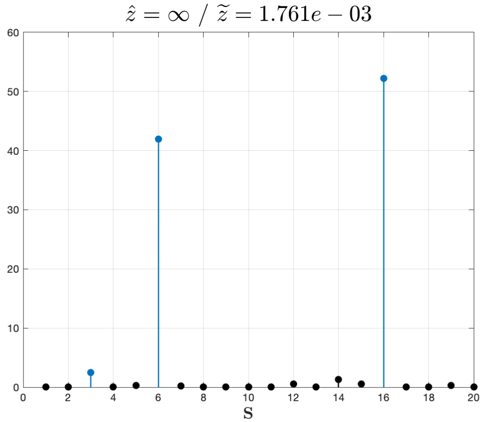


$\mathcal{C} = \{16, 6\}$

The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

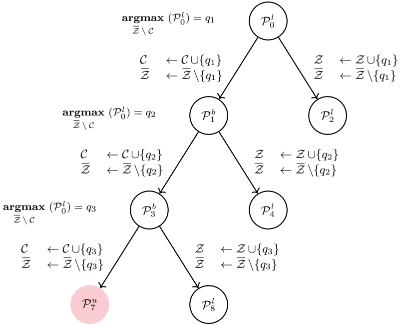
Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$





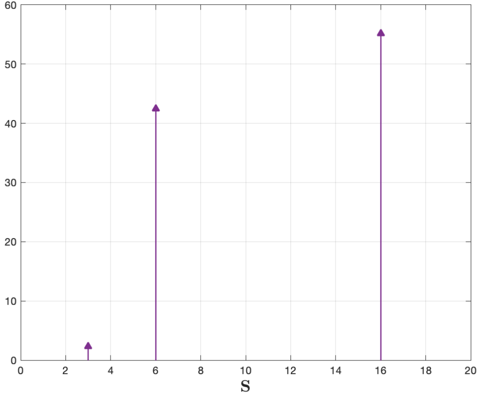
$$\mathcal{C} = \{16, 6, 3\} \quad \wedge \quad \textbf{Card}(\mathcal{C}) = K$$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



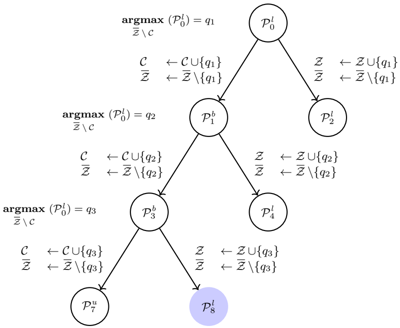
$\mathcal{C} = \{16, 6, 3\}$

$\hat{z} = \widetilde{z} = 1.789e - 03$

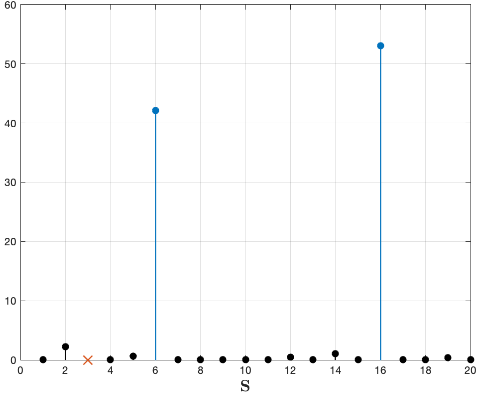


The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



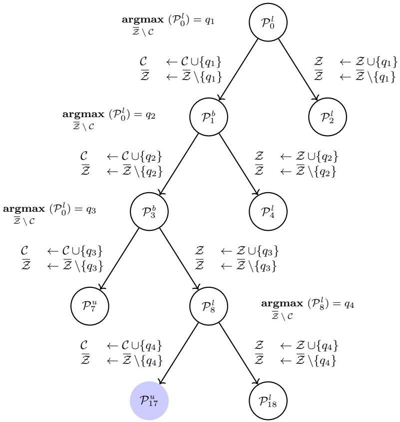
$\hat{z} = 1.789e-03 \ / \ \widetilde{z} = 1.781e-03 \ / \ \delta = 7.948e-06$

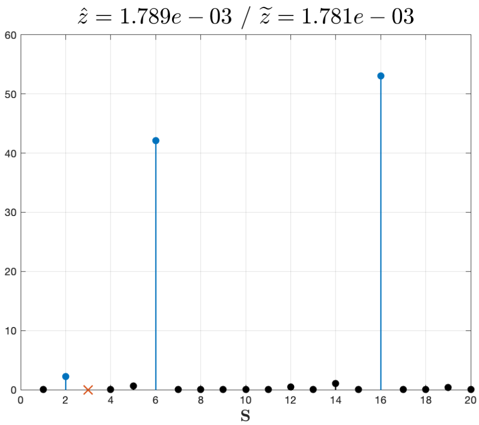


$\mathcal{C} = \{16, 6\}$

The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$

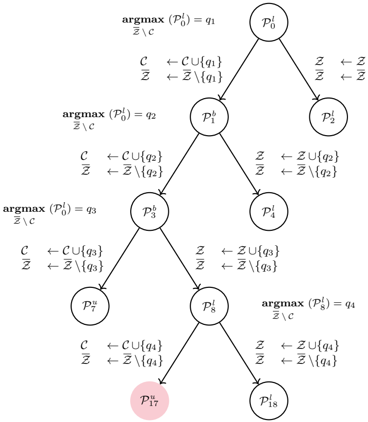




$$\mathcal{C} = \{16, 6, 2\} \quad \wedge \quad \textbf{Card}(\mathcal{C}) = K$$

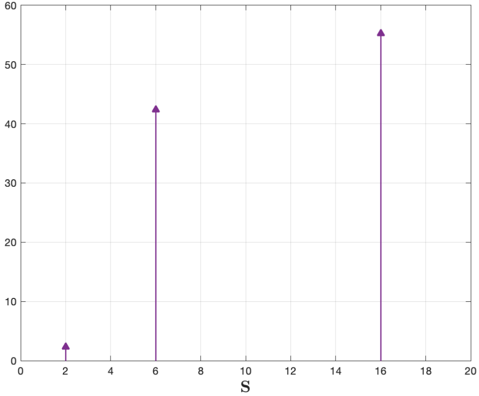
The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



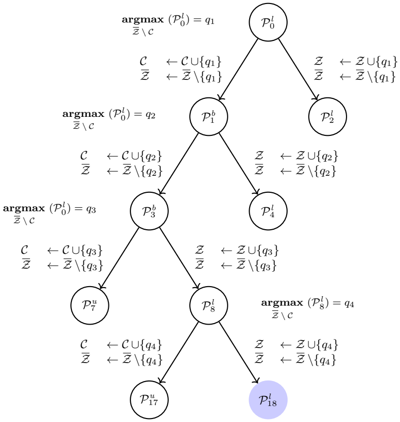
$\mathcal{C} = \{16, 6, 2\}$

$\hat{z} = 1.789e-03$ / $\widetilde{z} = 1.814e-03$ / $\delta = -2.500e-05$

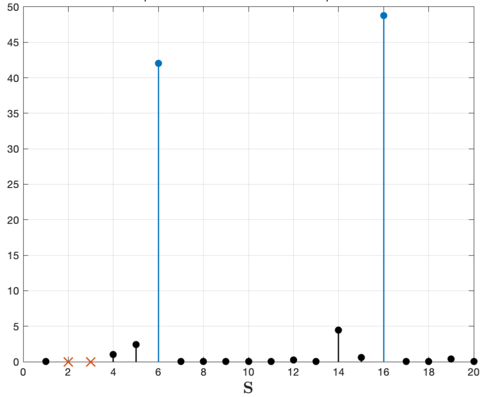


The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



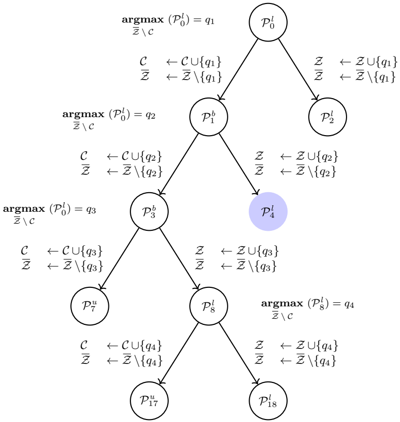
$\hat{z} = 1.789e - 03$ / $\widetilde{z} = 1.841e - 03$ / $\delta = -5.203e - 05$



$\mathcal{C} = \{16, 6\}$

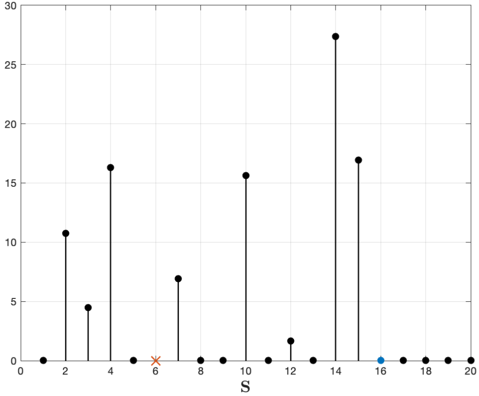
The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



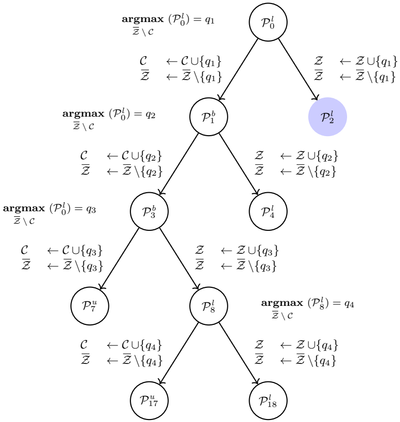
$\mathcal{C} = \{16\}$

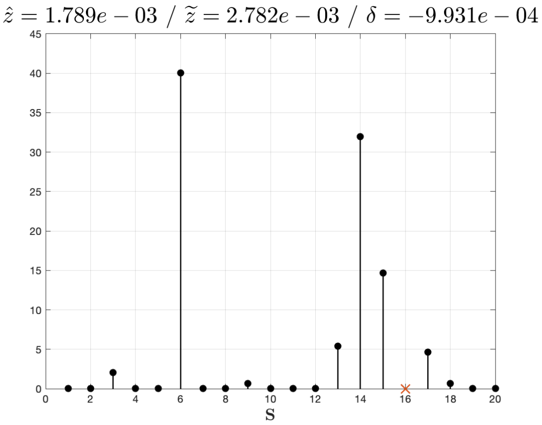
$\hat{z} = 1.789e - 03$ / $\widetilde{z} = 1.984e - 02$ / $\delta = -1.805e - 02$



The dedicated branch-and-bound: Example with $K = 3$, $Q = 20$

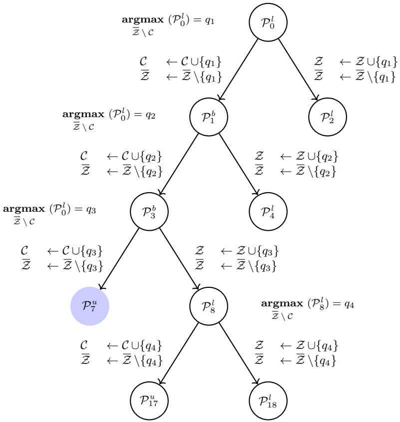
Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$





$\mathcal{C} = \emptyset$

Notations : $\hat{z}$ the best feasible solution, $\widetilde{z}$ the current solution, $\delta = \hat{z} - \widetilde{z}$



$\mathcal{C} = \{16, 6, 3\}$

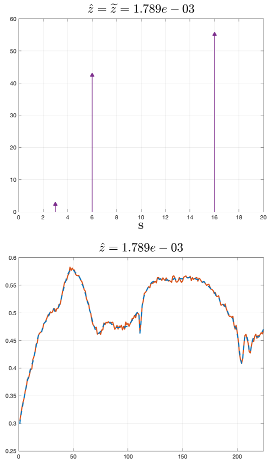


Table of contents

Spectral dictionary : USGS library [Clark et al., 2003]

Figure: Source : South Park - Season 18 (2014)

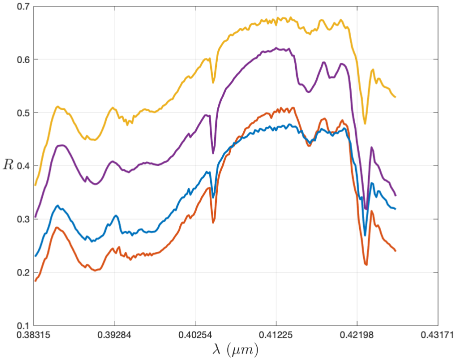


Figure: USGS atoms Calcite : AMX7, AMX18, AMX6, AMX43

USGS Digital Spectral Libraries

- $Q = 498$ pure spectral signatures (Minerals, Vegetation, Manmade materials, . . .)
- $N_\lambda = 224$ samples and the wavelength coverage is $[0.3, 2.7]$ μm

Simulated problems on USGS spectral library

Instances generation protocol

Two sets of designed problems such as:

Number ot atoms: $Q \in \{20, 50, 100, 200, 300, 400, 498\}$;

Sparsity level: $\widehat{K} \in [\![2, 6]\!]$;

Some noise: $\mathrm{SNR} \in \{40, 45, 50, 55, 60, \infty\}$ dB;

A minimum threshold: on the abundances $\tau = 0.1$;

Permutations of $\mathbf{S}$: 20.

Investigated methods

Time comparison:

B&B: our branch-and-bound with qpOASES solver [Ferreau et al., 2014] (C++)

MIP: using IBM Cplex solver (C++)

Quality comparison:

SDA: An iterative deflation algorithm [Greer, 2011] (Matlab)

$\widehat{K}$-FCLS: Two-phase algorithm using IBM Cplex solver (Matlab)

Simulated problems on USGS - Time comparison

$\mathrm{SNR} = \infty$		***B&B***	**MIP**	**Ratio**
$Q = 20$	$\widehat{K} = 2$	7,59e-04	8,23e-03	0,092
	$\widehat{K} = 4$	1,45e-03	6,17e-03	0,234
	$\widehat{K} = 6$	2,38e-03	6,70e-03	0,356
$Q = 100$	$\widehat{K} = 2$	1,70e-02	7,63e-02	0,223
	$\widehat{K} = 4$	1,86e-02	7,16e-02	0,26
	$\widehat{K} = 6$	3,64e-02	7,33e-02	0,497
$Q = 498$	$\widehat{K} = 2$	4,17e-01	6,95	0,06
	$\widehat{K} = 4$	1,03	9,28	0,11
	$\widehat{K} = 6$	1,93	1,91e+01	0,1

$\mathrm{SNR} = 60\mathrm{dB}$		***B&B***	**MIP**	**Ratio**
$Q = 20$	$\widehat{K} = 2$	6,46e-04	7,36e-03	0,088
	$\widehat{K} = 4$	1,45e-03	8,19e-03	0,177
	$\widehat{K} = 6$	2,17e-03	8,17e-03	0,265
$Q = 100$	$\widehat{K} = 2$	5,39e-03	7,61e-02	0,071
	$\widehat{K} = 4$	1,47e-02	7,68e-02	0,192
	$\widehat{K} = 6$	2,57e-02	1,63e-01	0,158
$Q = 498$	$\widehat{K} = 2$	2,79e-01	2,14	0,131
	$\widehat{K} = 4$	2,77	2,99	0,926
	$\widehat{K} = 6$	8,97	4,75	1,889

$\mathrm{SNR} = 50\mathrm{dB}$		***B&B***	**MIP**	**Ratio**
$Q = 20$	$\widehat{K} = 2$	5,76e-04	8,97e-03	0,064
	$\widehat{K} = 4$	1,15e-03	3,21e-02	0,036
	$\widehat{K} = 6$	1,64e-03	9,03e-03	0,182
$Q = 100$	$\widehat{K} = 2$	4,55e-03	9,41e-02	0,048
	$\widehat{K} = 4$	1,39e-02	7,67e-02	0,181
	$\widehat{K} = 6$	7,49e-02	1,84e-01	0,406
$Q = 498$	$\widehat{K} = 2$	5,29e-01	2,88	0,184
	$\widehat{K} = 4$	3,11e+01	1,74e+01	1,789
	$\widehat{K} = 6$	3,12e+02	1,51e+02	2,062

$\mathrm{SNR} = 40\mathrm{dB}$		***B&B***	**MIP**	**Ratio**
$Q = 20$	$\widehat{K} = 2$	6,91e-04	1,40e-02	0,049
	$\widehat{K} = 4$	1,37e-03	1,01e-02	0,136
	$\widehat{K} = 6$	2,18e-03	9,71e-03	0,225
$Q = 100$	$\widehat{K} = 2$	7,61e-03	1,10e-01	0,069
	$\widehat{K} = 4$	5,36e-02	1,35e-01	0,398
	$\widehat{K} = 6$	7,35e-01	4,18e-01	1,760
$Q = 498$	$\widehat{K} = 2$	7,95	4,91	1,619
	$\widehat{K} = 4$	4,39e+02	2,39e+02	1,834
	$\widehat{K} = 6$	9,08e+02	7,52e+02	1,208

- The exact method can be used to solve ℓ_0-SU problems
- The ratio of execution times seems to be favorable to our dedicated approach, except for the most difficult problems

Simulated problems on USGS - Quality comparison

Metric Exact Recovery Ratio (as %) : $\mathrm{ERR} := (\mathbf{supp}(\hat{\boldsymbol{a}}) \cap \mathbf{supp}(\mathring{\boldsymbol{a}}))\ /\ \widehat{K}$

$\mathrm{SNR} = \infty$		B&B	SDA	$\widehat{K}$-FCLS
$Q = 20$	$\widehat{K} = 2$	100	100	100
	$\widehat{K} = 4$	100	100	100
	$\widehat{K} = 6$	100	100	100
$Q = 100$	$\widehat{K} = 2$	100	100	100
	$\widehat{K} = 4$	100	100	100
	$\widehat{K} = 6$	100	100	100
$Q = 498$	$\widehat{K} = 2$	100	100	100
	$\widehat{K} = 4$	100	100	100
	$\widehat{K} = 6$	100	100	100

$\mathrm{SNR} = 60\mathrm{dB}$		B&B	SDA	$\widehat{K}$-FCLS
$Q = 20$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	100,0	100,0	100,0
$Q = 100$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	100,0	100,0	99,2
$Q = 498$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	100,0	99,6	98,3

$\mathrm{SNR} = 50\mathrm{dB}$		B&B	SDA	$\widehat{K}$-FCLS
$Q = 20$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	100,0	100,0	100,0
$Q = 100$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	100,0	99,6	97,9
$Q = 498$	$\widehat{K} = 2$	100,0	98,8	97,5
	$\widehat{K} = 4$	100,0	98,1	91,9
	$\widehat{K} = 6$	97,1	91,7	89,6

$\mathrm{SNR} = 40\mathrm{dB}$		B&B	SDA	$\widehat{K}$-FCLS
$Q = 20$	$\widehat{K} = 2$	100,0	100,0	100,0
	$\widehat{K} = 4$	100,0	100,0	100,0
	$\widehat{K} = 6$	98,8	99,2	97,1
$Q = 100$	$\widehat{K} = 2$	100,0	98,8	97,5
	$\widehat{K} = 4$	100,0	94,4	92,5
	$\widehat{K} = 6$	95,0	92,1	87,9
$Q = 498$	$\widehat{K} = 2$	100,0	93,8	86,3
	$\widehat{K} = 4$	85,6	76,9	65,0
	$\widehat{K} = 6$	63,8	55,0	52,1

- Giving time to exactly solve the ℓ_0-SU is quite interesting !

Table of contents

Conclusion - Main perspectives

To work with (semi)real datasets: Collaboration with Lucas Drumetz (IMT Atlantique)

- **Unsupervised** SU;
- The dictionary, composed of $Q = 15$ spectra, is learned from a data set;
- **Teaser**
  - Branch-and-bound algorithm is **very promising** in regards to **computation time**;
  - We are in a position to show **the interest of exact optimization**;
  - We have some problems with the **interpretability** of the data in the case of **high correlated** dictionary.

To publish the paper: Ongoing!

To implement other OR stuffs in the current branch-and-bound framework;

To tackle other specific problems: e.g. structured sparsity, . . .

Thank you for your attention

References I

Ben Mhenni, R. (2020). Méthodes de programmation en nombres mixtes pour l'optimisation parcimonieuse en traitement du signal. PhD thesis, ECOLE CENTRALE DE NANTES.

Ben Mhenni, R., Bourguignon, S., Ninin, J., and Schmidt, F. (2018). Spectral unmixing with sparsity and structuring constraints. pages 1–5.

Bertsimas, D., King, A., and Mazumder, R. (2016). Best subset selection via a modern optimization lens. The Annals of Statistics, 44(2):813 – 852.

Bertsimas, D. and Shioda, R. (2009). Algorithm for cardinality-constrained quadratic optimization. Computational Optimization and Applications, 43:1–22.

Bienstock, D. (1999). A computational study of a family of mixed-integer quadratic programming problems. Mathematical Programming, Series B, 74.

Bourguignon, S., Ninin, J., Carfantan, H., and Mongeau, M. (2016). Exact sparse approximation problems via mixed-integer programming: Formulations and computational performance. IEEE Transactions on Signal Processing, 64(6):1405–1419.

Clark, R., Swayze, G., Wise, R., Livo, K., Hoefen, T., Kokaly, R., and Sutley, S. (2003). USGS digital spectral library splib05a. US Geological Survey, Digital Data Series, 231.

Ferreau, H., Kirches, C., Potschka, A., Bock, H., and Diehl, M. (2014). qpOASES: A parametric active-set algorithm for quadratic programming. Mathematical Programming Computation, 6(4):327–363.

Greer, J. (2011). Sparse demixing of hyperspectral images. IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, 21:219–28.

Hazimeh, H., Mazumder, R., and Saab, A. (2021). Sparse regression at scale: Branch-and-bound rooted in first-order optimization.

Heinz, D. C. et al. (2001). Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. IEEE transactions on geoscience and remote sensing, 39(3):529–545.

Land, A. H. and Doig, A. G. (1960). An Automatic Method of Solving Discrete Programming Problems. Econometrica: Journal of the Econometric Society, 28(3):497–520.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. SIAM Journal on Computing, 24(2):227–234.

Singer, R. B. and McCord, T. B. (1979). Mars-large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In Lunar and Planetary Science Conference Proceedings, volume 10, pages 1835–1848.

Tropp, J. A. and Wright, S. J. (2010). Computational Methods for Sparse Solution of Linear Inverse Problems. Proceedings of the IEEE, 98(6).