Partition and Measure: a new technique for analyzis of Branch and Bound algorithms

N. Bourgeois (ESSEC),
B. Escoffier, V. Paschos (LAMSADE, Dauphine)

8 février 2012

A generic problem Π

Two parameters : p_1, p_0 with $p_0 + p_1 = N$

Three possible situations : A ,B, $\bar{A} \wedge \bar{B}$

An hypothesis :

$$p_1 \geq p_0 \Longrightarrow A = \mathtt{TRUE}.$$

A basic algorithm

Branching : $I \mapsto I', I'$

$$
\begin{aligned}
A = \texttt{TRUE} \quad &\Longrightarrow \quad p_1(I') \leq p_1(I) - 4, \\
& \quad p_0(I') \leq p_0(I). \\
A = \texttt{FALSE}, B = \texttt{TRUE} \quad &\Longrightarrow \quad p_1(I') \leq p_1(I) - 2, \\
& \quad p_0(I') \leq p_0(I) + 2. \\
A, B = \texttt{FALSE} \quad &\Longrightarrow \quad p_1(I') \leq p_1(I), \\
& \quad p_0(I') \leq p_0(I) - 1.
\end{aligned}
$$

Standard analyzis is a bit helpless

Since $N = p_0 + p_1$,

$$A = \texttt{TRUE} \implies T(N) \leq 2T(N-4)$$

$$A = \texttt{FALSE}, B = \texttt{TRUE} \implies T(N) \leq 2T(N) \quad ???$$

$$A, B = \texttt{FALSE} \implies T(N) \leq 2T(N-1)$$

.

Let's do Measure and Conquer instead

Fix $k = w_0 p_0 + w_1 p_1$,

$$A = \texttt{TRUE} \implies T(k) \leq 2T(k - 4w_1)$$

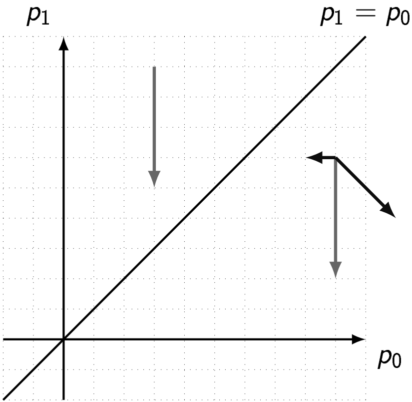
$$A = \texttt{FALSE}, B = \texttt{TRUE} \implies T(k) \leq 2T(k - 2w_1 + 2w_0)$$

$$A, B = \texttt{FALSE} \implies T(k) \leq 2T(k - w_0)$$

.

Optimal is $w_0 = 2w_1/3$, $T(N) \leq 2^{3N/2}$

Can we do something better ?



Another analyzis is possible (1)

- use different measures k_0, k_1 for $p_1 \leq p_0$ and $p_1 \geq p_0$.
- compute a solution T_0 for $p_1 \leq p_0$:
  Optimal is still $w_0 = 2w_1/3$, $T_0(k_0) \leq 2^{3k_0/2}$

Another analyzis is possible (2)

- use T_0 as a lower bound for $p_0 = p_1$.
- compute a solution T_1 for $p_1 \geq p_0$:
  In this part, optimal is $w_0 = w_1$, $T_1(k_1) \leq 2^{5k_1/4}$

$$\max\left\{\max_{p_1 \leq p_0}(2^{3k_0/2}), \max_{p_1 \geq p_0}(2^{5k_1/4})\right\} \leq 2^{5N/4}$$

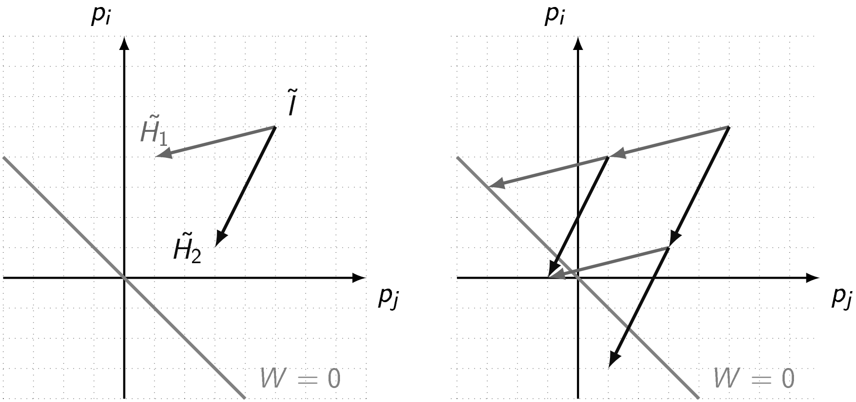
The quotient space of instances

Given a set of parameters $p_1, \ldots, p_s$ such that the problem is polynomial for $\sum p_i = 0$,
Given two instances I, I',

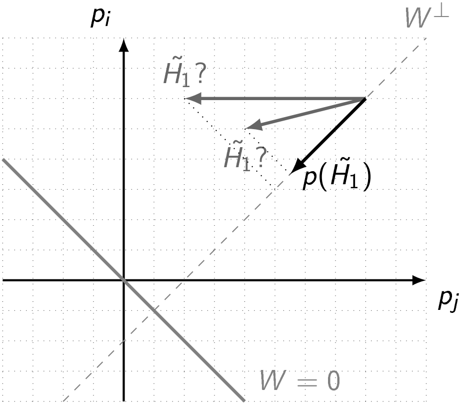
$$I \sim I' \iff \forall j, p_j(I) = p_j(I')$$

We work in $\mathcal{G}_n / \sim$

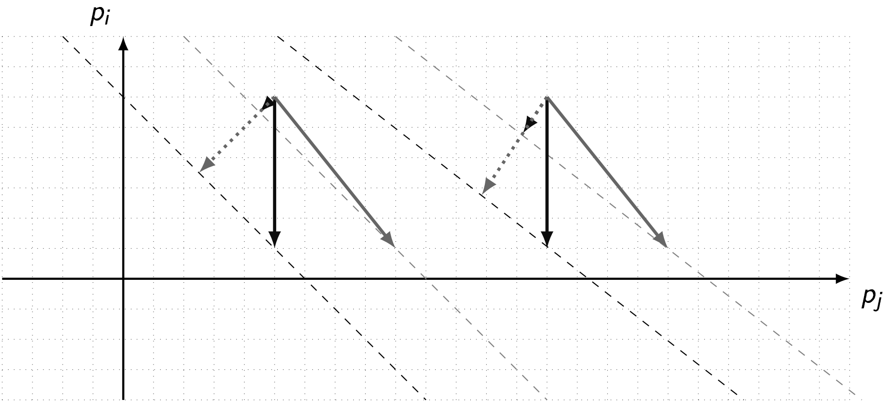
A Branching Rule



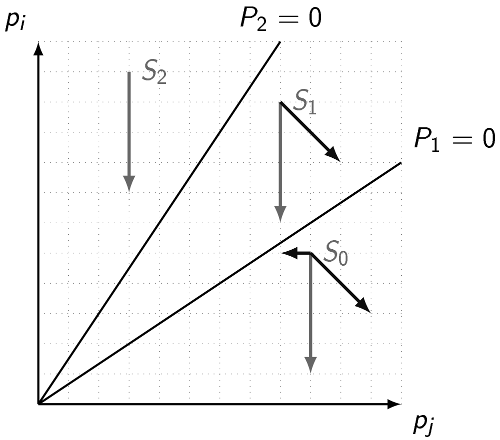
Projection allows to compare



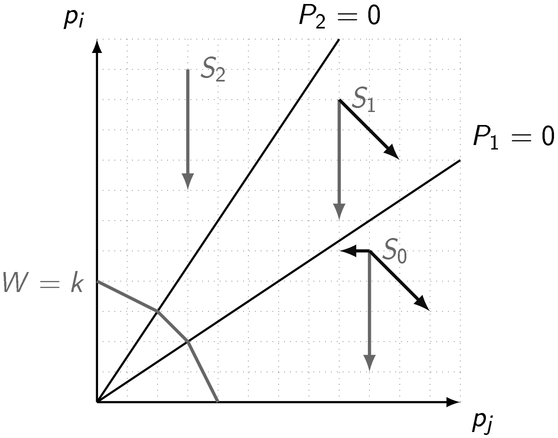
Fixing the objective hyperplan



Division of the space (1)



Division of the space (2)



Sets of recurrences

For a branching B_t, in the subspace S_i : $v_{i,t,j} =< \overrightarrow{\tilde{I}\tilde{H}_j}, W_i >$

$$T_i(k) \leq \max_t \left\{ \sum_j T_i(k - v_{i,t,j}) \right\},$$

which leads to :

$$1 \leq \max_t \left\{ \sum_j 2^{-\alpha_i v_{i,t,j}} \right\}.$$

Border conditions

Let $i' > i$ such that $S_{i'}$ and $S_i \setminus S_{i+1}$ have a border β, Let $\tilde{I} \in \beta$.

$$T_{W,i'}(\tilde{I}) \geq T_{W,i}(\tilde{I})$$

The latest inequations can be rewritten :

$$\alpha_{i'} \sum w_{i',j} p_j \geq \alpha_i \sum w_{i,j} p_j$$

Transversal division

- $(J^d)_{1 \leq d \leq \Delta}$ is a sequence of subspaces of $\mathcal{I}_n / \sim$ of increasing dimension. More precisely :

$$\begin{aligned} J^d &= \mathrm{Vect}(e_1, \ldots, e_{j^d}) = \{e_1, \ldots, e_{j^d}\}^n \\ j^{d+1} &> j^d \end{aligned}$$

- for each J^d, we define a partition (S_i^d) exactly like before.

Set Cover

- A ground set $\mathcal{C} = \{c_1, \cdots, c_n\}$
- A set system $\mathcal{S} = \{S_1, \cdots, S_m\}$
- Goal : a minimum size subset $\mathcal{S}' \subseteq \mathcal{S}$ that covers all elements in $\mathcal{C}$.

Basic algorithm

If no easy case occurs, branch on a set of maximal size : either take it or not.

Measure and conquer analysis

- Weight w_i to sets of size i, weight v_j of sets of frequency j.
- Analysis of the branching. Δ_{IN} and Δ_{OUT} (decreasing of the total weight when taking S or not) depending on :
  $\rightarrow$ The size d of the set S we branch on
  $\rightarrow$ The frequencies of the d elements in S.

  Set of recurrences :
  r_j : number of elements of frequency j in S.
  $\forall d, \forall r_j \ s.t. \sum_j r_j = d, \ 2^{-\alpha\Delta_{IN}} + 2^{-\alpha\Delta_{OUT}} \leq 1$.
  Global complexity $2^{\alpha(w_{max}|\mathcal{S}| + v_{max}|\mathcal{C}|)}$

Measure and conquer analysis

- Weight w_i to sets of size i, weight v_j of sets of frequency j.
- Analysis of the branching. Δ_{IN} and Δ_{OUT} (decreasing of the total weight when taking S or not) depending on :
  $\rightarrow$ The size d of the set S we branch on
  $\rightarrow$ The frequencies of the d elements in S.
  Set of recurrences :
  r_j : number of elements of frequency j in S.
  $\forall d, \forall r_j \; s.t. \sum_j r_j = d, \; 2^{-\alpha\Delta_{IN}} + 2^{-\alpha\Delta_{OUT}} \leq 1.$
  Global complexity $2^{\alpha(w_{max}|\mathcal{S}|+v_{max}|\mathcal{C}|)}$

Measure and conquer analysis

- Weight w_i to sets of size i, weight v_j of sets of frequency j.
- Analysis of the branching. Δ_{IN} and Δ_{OUT} (decreasing of the total weight when taking S or not) depending on :
  $\rightarrow$ The size d of the set S we branch on
  $\rightarrow$ The frequencies of the d elements in S.
  Set of recurrences :
  r_j : number of elements of frequency j in S.
  $\forall d, \forall r_j \ s.t. \sum_j r_j = d,\ 2^{-\alpha\Delta_{IN}} + 2^{-\alpha\Delta_{OUT}} \leq 1.$
  Global complexity $2^{\alpha(w_{max}|\mathcal{S}|+v_{max}|\mathcal{C}|)}$

Idea of Partition and Measure

1. In R_1 : n_d is small or m_2 is large $\rightarrow$ same analysis, but better complexity bound.
2. In R_2 : there exists one set of size d with no element of frequency 2 $\rightarrow$ restricted set of recurrences ($r_2 = 0$).

Generalization with several hyperplans for each d ;
Computation : on-going work !

Idea of Partition and Measure

1. In R_1 : n_d is small or m_2 is large $\rightarrow$ same analysis, but better complexity bound.
2. In R_2 : there exists one set of size d with no element of frequency 2 $\rightarrow$ restricted set of recurrences ($r_2 = 0$).

Generalization with several hyperplans for each d ;
Computation : on-going work !

A wide range of possibilities...

- Problems where branch and bound technique is efficient.
- Graphs where better branching exists when maximal degree is high.
- Typical other example : maximum independent set.

... with some computation cost

- r parameters, s subspaces $\Longrightarrow p \times s$ variables.
- leading measure and conquer algorithms are often quite tedious to analyze (4.8×10^6 recurrences for maximum independent set).