Filtering AtMostNValue with difference constraints:

Application to the Shift Minimisation Personnel Task Scheduling Problem

Jean-Guillaume Fages[1] and Tanguy Lapègue[2]
Ecole des Mines de Nantes, LUNAM Université, France
{jean-guillaume.fages,tanguy.lapegue}@mines-nantes.fr
[1] LINA (TASC), [2] IRCCyN (SLP)

September 17, 2013

Outline

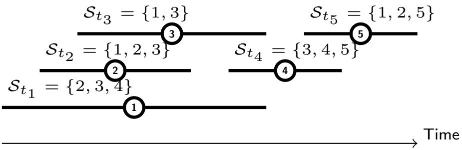
Problem definition

The Shift Minimisation Personnel Task Scheduling Problem

Assign a shift to each task :

- Tasks are fixed in time
- Overlapping tasks must be assigned to different shifts
- Tasks require particular skills

$\Rightarrow$ Minimise the number of shifts



Input data

Task	Shift
t_1	2
t_2	3
t_3	1
t_4	3
t_5	1

Optimum (3 shifts)

SMPTSP CP model

$\mathcal{S} = \{1, .., m\}$: Shifts
$\mathcal{T} = \{t_1, .., t_n\}$: Tasks
$\mathcal{C}$: Sets of overlapping tasks

z : Objective variable
$\mathcal{X}$: Assignment variables ($\mathcal{T} \rightarrow \mathcal{S}$)

CP model

minimise z (1)

subject to : $\texttt{AllDifferent}(\{x_i \,|\, t_i \in \mathcal{K}\}) \quad , \forall \mathcal{K} \in \mathcal{C}$ (2)

$\texttt{AtMostNValue}(\mathcal{X}, z)$ (3)

$Dom(z) = [LB_{\neq}, m]$ (4)

$Dom(x_i) = \mathcal{S}_{t_i} \quad , \forall t_i \in \mathcal{T}$ (5)

$$LB_{\neq} = \max_{\mathcal{K} \in \mathcal{C}}(|\mathcal{K}|)$$

Filtering AtMostNValue

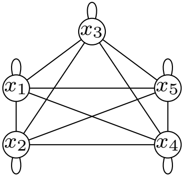
State-of-the-art propagator [Bessière *et. al.*, 2006]

1. Compute the **intersection graph** $G_{\mathcal{I}} = (V, E_{\mathcal{I}})$ of $\mathcal{X}$
2. Find an **independent set** A in $G_{\mathcal{I}}$ with **min degree** (MD)
3. Filter $\mathcal{X}$ and z from A and **rules** $\mathcal{R}_1$ and $\mathcal{R}_2$

$\mathcal{R}_1$ $\quad \underline{z} \leftarrow \max(\underline{z}, |A|)$

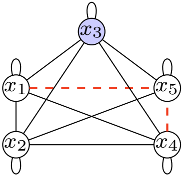
$\mathcal{R}_2$ $\quad |A| = \overline{z} \Rightarrow \forall i \in V, Dom(x_i) \leftarrow Dom(x_i) \cap \bigcup_{a \in A} Dom(x_a)$

$z = [1, 1]$
$x_1 = \{2, 3, 4\}$
$x_2 = \{1, 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, 4, 5\}$
$x_5 = \{1, 2, 5\}$



Intersection graph

$z = [1, 1]$
$x_1 = \{\not 2, 3, \not 4\}$
$x_2 = \{1, \not 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, \not 4, \not 5\}$
$x_5 = \{1, \not 2, \not 5\}$



Propagation

Filtering AtMostNValue : Formalism

A generic family propagator for $\texttt{AtMostNValue}(\mathcal{X}, z)$

$\texttt{AMNV}\langle G|H|\mathcal{R}\rangle$ propagators

1. Compute a graph $G(\mathcal{X})$
2. Compute independent sets $\mathcal{A}$ in G with a heuristic H
3. Filter $\mathcal{X}$ and z from $\mathcal{A}$ and a set of rules $\mathcal{R}$

Hence the state-of-the-art propagator is $\texttt{AMNV}\langle G_{\mathcal{I}}|\texttt{MD}|\mathcal{R}_{1,2}\rangle$

Outline

Embedding inequalities into the intersection graph

Core observation

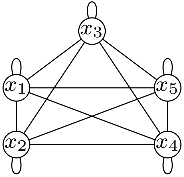
If constraint $x_i \neq x_j$ is present, for any solution, $(i, j) \notin G_{\mathcal{I}}$

Let's remove such edges !

Constrained intersection graph

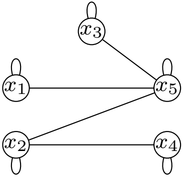
$G_{\mathcal{CI}} = (V, E_{\mathcal{I}} \setminus \{(i, j) |$ constraint $x_i \neq x_j$ exists$\})$

$z = [1, 3]$
$x_1 = \{2, 3, 4\}$
$x_2 = \{1, 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, 4, 5\}$
$x_5 = \{1, 2, 5\}$



$G_{\mathcal{I}}$

$z = [1, 3]$
$x_1 = \{2, 3, 4\}$
$x_2 = \{1, 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, 4, 5\}$
$x_5 = \{1, 2, 5\}$



$G_{\mathcal{CI}}$

Embedding inequalities into the intersection graph

Core observation

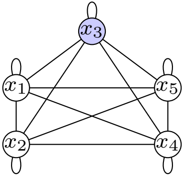
If constraint $x_i \neq x_j$ is present, for any solution, $(i, j) \notin G_{\mathcal{I}}$

Let's remove such edges !

Constrained intersection graph

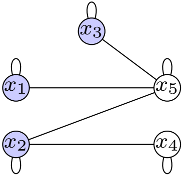
$G_{\mathcal{CI}} = (V, E_{\mathcal{I}} \setminus \{(i, j)|$ constraint $x_i \neq x_j$ exists$\})$

$z = [1, 3]$
$x_1 = \{2, 3, 4\}$
$x_2 = \{1, 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, 4, 5\}$
$x_5 = \{1, 2, 5\}$



$G_{\mathcal{I}}$: No filtering

$z = [3, 3]$
$x_1 = \{2, 3, 4\}$
$x_2 = \{1, 2, 3\}$
$x_3 = \{1, 3\}$
$x_4 = \{3, 4, \not{5}\}$
$x_5 = \{1, 2, \not{5}\}$



$G_{\mathcal{CI}}$: Filtering

Tightening filtering rules

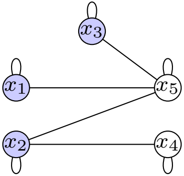
Improving $\mathcal{R}_2$ with vertex *neighbourhood*

$\mathcal{R}_3$ $\quad |A| = \overline{z} \Rightarrow \forall i \in V \backslash A,$

- $Dom(x_i) \leftarrow Dom(x_i) \cap \bigcup_{a \in A_i} Dom(x_a)$
- $A_i = \{j\} \Rightarrow Dom(x_j) \leftarrow Dom(x_j) \cap Dom(x_i)$

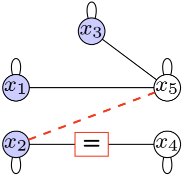
Where, $A_i = \{j \in A | (i,j) \in E_{\mathcal{CI}}\}$

$z = [3,3]$
$x_1 = \{2,3,4\}$
$x_2 = \{1,2,3\}$
$x_3 = \{1,3\}$
$x_4 = \{3,4,\not{5}\}$
$x_5 = \{1,2,\not{5}\}$



$\mathtt{AMNV}\langle G_{\mathcal{CI}} | \mathtt{MD} | \mathcal{R}_{1,2} \rangle$

$z = [3,3]$
$x_1 = \{2,3,4\}$
$x_2 = \{\not{1},\not{2},3\}$
$x_3 = \{1,3\}$
$x_4 = \{3,\not{4},\not{5}\}$
$x_5 = \{1,2,\not{5}\}$



$\mathtt{AMNV}\langle G_{\mathcal{CI}} | \mathtt{MD} | \mathcal{R}_{1,3} \rangle$

Diversifying filtering

About independent sets

Filtering depends on independent sets

- Finding **large** independent sets is important
- Finding **different** independent sets is also important

MD is very efficient (good approximation ratio) but deterministic

The R^k heuristic

Computes k random independent sets

$\Rightarrow$ Use R^k in addition to MD

Outline

Experimental settings

Instance sets

Data_137

- 137 instances
- 60-2100 tasks
- 23-417 workers

Data_100

- 100 instances
- 70-1600 tasks
- 60-950 workers

Solvers

- CP : CHOCO-3.1.0
- MIP : CPLEX-12.4 (default settings)

Root node evaluation

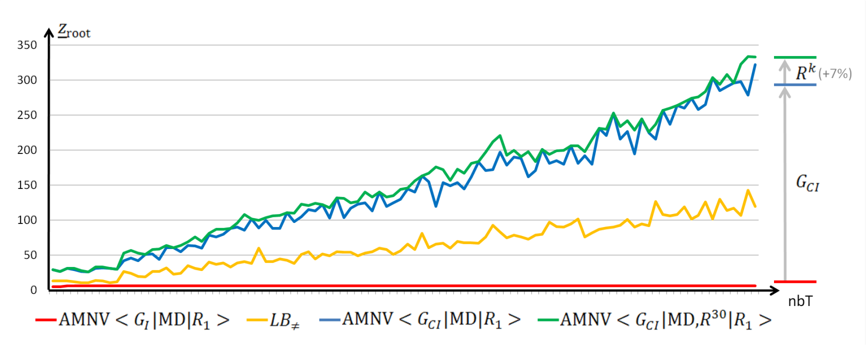


FIGURE – Root node lower bounds on Data_100

Scalability : CP vs. MIP

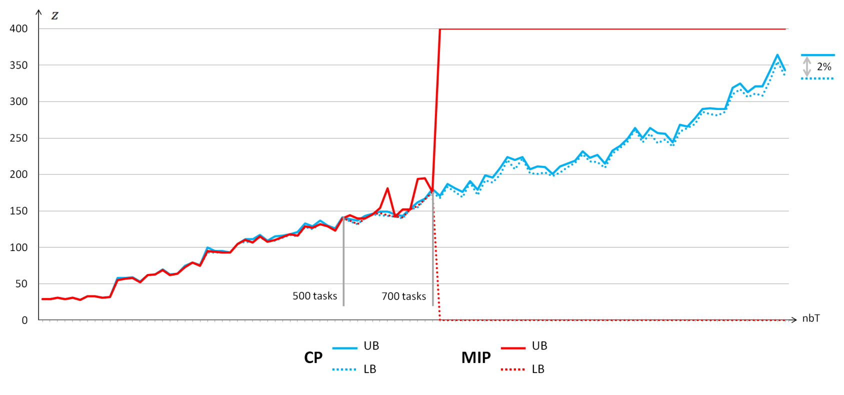


FIGURE – Lower and Upper bounds on Data_100, after 5 minutes

Overall evaluation

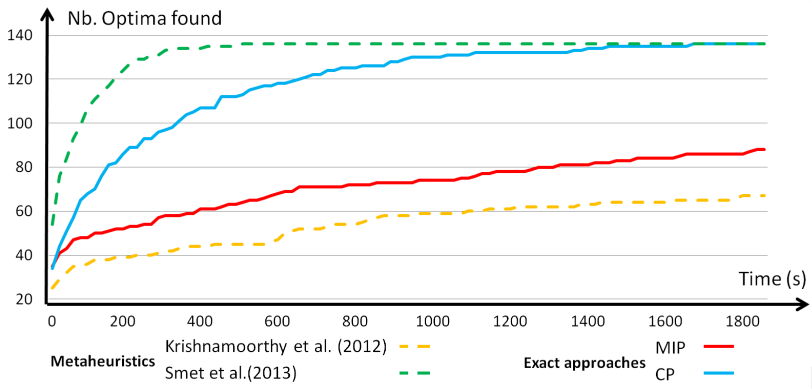


FIGURE – Positioning *w.r.t.* state-of-the-art approaches on `Data_137`

Outline

Conclusion

A CP model for the SMPTSP

- Very competitive
- Scalable

And more generally...

Propagation for `AtMostNValue` and inequalities

- New formalism : $\texttt{AMNV}\langle G|H|\mathcal{R}\rangle$
  - A constrained graph
  - A diversified heuristic
  - Refined filtering rules
- Significant improvements
- Simple to implement

Thank you for your attention

Any question ?

Filtering AtMostNValue with difference constraints:

Application to the Shift Minimisation Personnel Task Scheduling Problem

Jean-Guillaume Fages[1] and Tanguy Lapègue[2]
Ecole des Mines de Nantes, LUNAM Université, France
{jean-guillaume.fages,tanguy.lapegue}@mines-nantes.fr
[1] LINA (TASC), [2] IRCCyN (SLP)

September 17, 2013