Search, propagation, and learning in sequencing and scheduling problems

Mohamed Siala

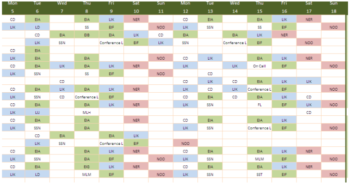
http://ucc.insight-centre.org/msiala
mohamed.siala@insight-centre.org

Christian Artigues	LAAS-CNRS Toulouse
Fahiem Bacchus	University of Toronto
Christian Bessiere	LIRMM Montpellier
Hadrien Cambazard	G-SCOP & Grenoble INP
Emmanuel Hebrard	LAAS-CNRS Toulouse
George Katsirelos	INRA Toulouse
Christine Solnon	INSA Lyon

INSA TOULOUSE
Insight
LAAS-CNRS
UCC Coláiste na hOllscoile Corcaigh, Éire University College Cork, Ireland

Context

Sequencing and Scheduling: the organization in time of of operations subject to capacity and resource constraints.



Mon 5	Tue 6	Wed 7	Thu 8	Fri 9	Sat 10	Sun 11	Mon 12	Tue 13	Wed 14	Thu 15	Fri 16	Sat 17	Sun 18
CD	EIA		EIA	LIK	NER		CD	EIA		EIA	LIK	NER	
LIK	LD		SS	EIF		NOO	LIK	SSN		SS	EIF		NOO
	CD	EIA	EIB	EIA	LIK	CD	EIA		EIA	LIK	NER		
	LIK	SSN		Conference L	EIF	LIK	SSN		Conference L	EIF		NOO	
CD	EIA												
LIK	EIA		EIA			NOO	CD	LIK		EIA	LIK	NER	
CD	EIA	LIK	EIA	LIK	NER		LIK		LIK	On Call	EIF		NOO
LIK	SSN		SS	EIF		NOO		CD					
		CD					CD	LIK	CD	EIA	LIK	LIK	
CD	EIA	LIK	EIA	LIK	NER		LIK	CD	LIK	Conference L	EIF		NOO
LIK	SSN	CD	Conference L	EIF			CD	EIA	CD	EIA	LIK	CD	
CD	EIA		EIA	LIK	NER		LIK	SSN		FL	EIF	LIK	NOO
LIK	LIJ		MLH									CD	
CD	EIA		EIA		NER		CD	EIA		EIA	LIK		
LIK	SSN		EIA				LIK	SSN		Conference L	EIF		NOO
	CD	EIA		EIA	LIK								
	LIK	SSN		Conference L	EIF		NOO						
CD	EIA		EIA	LIK	NER		CD	EIA		EIA	LIK	NER	
LIK	SSN		EIA	EIF		NOO	LIK	SSN		MLM	EIF		NOO
CD	EIA		EIG	LIK	NER		CD	EIA		EIA	LIK	NER	
LIK	LD		MLM	EIF		NOO	LIK	SSN		SST	EIF		NOO

PhD Context

- Combinatorial (optimization) problems
- Constraint satisfaction and optimization

- Laboratory: LAAS-CNRS, Toulouse
- Research Team: ROC (Operations Research, Combinatorial Optimization and Constraints)
- Supervision: Christian Artigues, and Emmanuel Hebrard
- Funding:

Thesis overview

Constraint Programming: Search $\oplus$ Propagation

Thesis overview

Constraint Programming: Search $\oplus$ Propagation $\oplus$ **Learning**

Thesis overview

Constraint Programming: Search $\oplus$ Propagation $\oplus$ **Learning**

All these aspects are important and must all be taken into account in order to design efficient solution methods

Outline

Definition

A constraint is a finite relation

Definition

A constraint is a finite relation

Definition

A `constraint network` (CN) is defined by a triplet $\mathcal{P} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ where

- $\mathcal{X} = [x_1, \ldots, x_n]$: finite set of variables
- $\mathcal{D}$: a domain for $\mathcal{X}$
- $\mathcal{C}$: finite set of constraints

Definition

A constraint is a finite relation

Definition

A $\texttt{constraint network}$ $(\texttt{CN})$ is defined by a triplet $\mathcal{P} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ where

- $\mathcal{X} = [x_1, \ldots, x_n]$: finite set of variables
- $\mathcal{D}$: a domain for $\mathcal{X}$
- $\mathcal{C}$: finite set of constraints

- Constraint Satisfaction Problem $(\texttt{CSP})$: deciding whether a constraint network has a solution or not
- CSP is NP-Hard in general

Definition

A constraint is a finite relation

Definition

A `constraint network` (`CN`) is defined by a triplet $\mathcal{P} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ where

- $\mathcal{X} = [x_1, \ldots, x_n]$: finite set of variables
- $\mathcal{D}$: a domain for $\mathcal{X}$
- $\mathcal{C}$: finite set of constraints

- Constraint Satisfaction Problem (`CSP`): deciding whether a constraint network has a solution or not
- CSP is NP-Hard in general

- Complete backtracking algorithms

Search & Propagation

Search & Propagation

Search

- Search: decisions to explore the search tree

Search & Propagation

Search

- Search: decisions to explore the search tree
- Search in $\mathrm{CP}=$ variable ordering $+$ value ordering

Search & Propagation

Search

- Search: decisions to explore the search tree
- Search in $\mathrm{CP}=$ variable ordering $+$ value ordering
- Standard or customized

Search & Propagation

Search

- Search: decisions to explore the search tree
- Search in $\mathrm{CP}=$ variable ordering $+$ value ordering
- Standard or customized

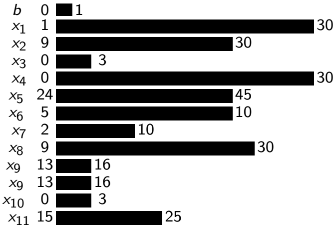
Propagation

- Propagation: inferences based on the current state
- Constraint $\leftrightarrow$ propagator
- The level of pruning $\leftrightarrow$ local consistency

Learning

Learning

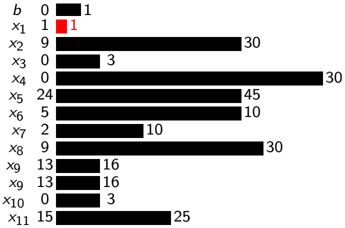
$$
\begin{aligned}
&x_1 + x_7 \geq 4 \wedge \\
&x_2 + x_{10} \geq 11 \wedge \\
&x_3 + x_9 = 16 \wedge \\
&x_5 \geq x_8 + x_9 \wedge \\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
&b \rightarrow (x_6 \geq 7) \wedge \\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
&x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



Learning

$$x_1 + x_7 \geq 4 \wedge$$
$$x_2 + x_{10} \geq 11 \wedge$$
$$x_3 + x_9 = 16 \wedge$$
$$x_5 \geq x_8 + x_9 \wedge$$
$$b \leftrightarrow (x_9 - x_4 = 14) \wedge$$
$$b \rightarrow (x_6 \geq 7) \wedge$$
$$b \rightarrow (x_6 + x_7 \leq 9) \wedge$$
$$x_{11} \geq x_9 + x_{10}$$

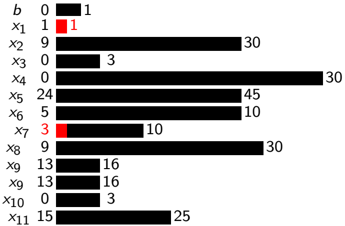
$[\![x_1 = 1]\!]$



Learning

$$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$$

$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$

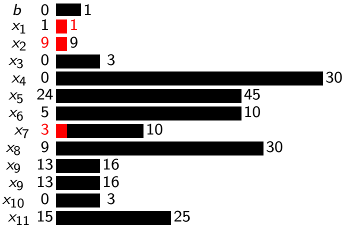


Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

$[\![x_2 = 9]\!]$

$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$

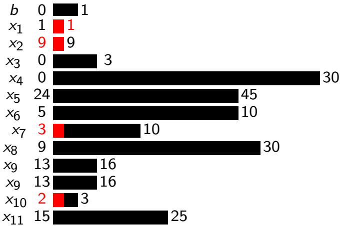


Learning

$$
\begin{aligned}
&x_1 + x_7 \geq 4 \wedge \\
&x_2 + x_{10} \geq 11 \wedge \\
&x_3 + x_9 = 16 \wedge \\
&x_5 \geq x_8 + x_9 \wedge \\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
&b \rightarrow (x_6 \geq 7) \wedge \\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
&x_{11} \geq x_9 + x_{10}
\end{aligned}
$$

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

$[\![x_2 = 9]\!] \longrightarrow [\![x_{10} \geq 2]\!]$



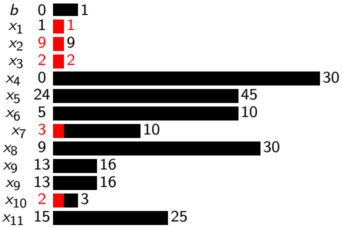
Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

$[\![x_2 = 9]\!] \longrightarrow [\![x_{10} \geq 2]\!]$

$[\![x_3 = 2]\!]$

$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



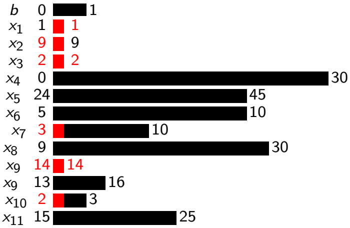
Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

$[\![x_2 = 9]\!] \longrightarrow [\![x_{10} \geq 2]\!]$

$[\![x_3 = 2]\!] \longrightarrow [\![x_9 = 14]\!]$

$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



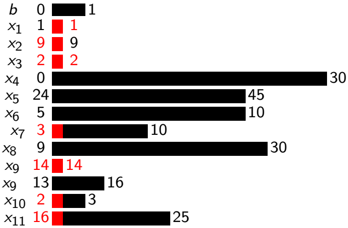
Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

$[\![x_2 = 9]\!] \longrightarrow [\![x_{10} \geq 2]\!]$

$[\![x_3 = 2]\!] \longrightarrow [\![x_9 = 14]\!] \rightarrow [\![x_{11} \geq 16]\!]$

$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

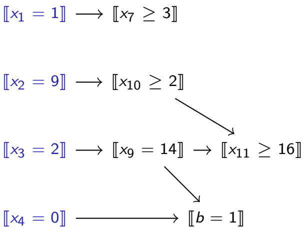
$[\![x_2 = 9]\!] \longrightarrow [\![x_{10} \geq 2]\!]$

$[\![x_3 = 2]\!] \longrightarrow [\![x_9 = 14]\!] \rightarrow [\![x_{11} \geq 16]\!]$

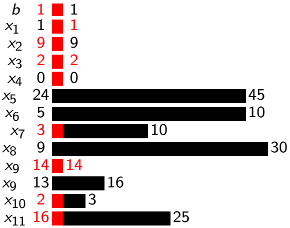
$[\![x_4 = 0]\!]$

$$
\begin{aligned}
&x_1 + x_7 \geq 4 \wedge \\
&x_2 + x_{10} \geq 11 \wedge \\
&x_3 + x_9 = 16 \wedge \\
&x_5 \geq x_8 + x_9 \wedge \\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
&b \rightarrow (x_6 \geq 7) \wedge \\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
&x_{11} \geq x_9 + x_{10}
\end{aligned}
$$

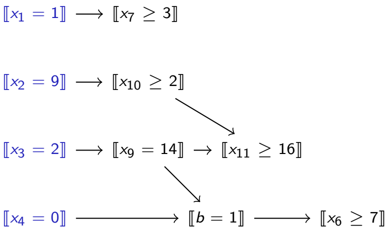
Learning



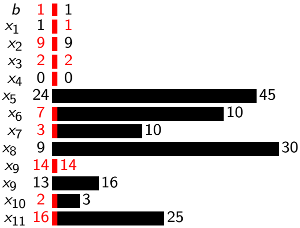
$$
\begin{aligned}
&x_1 + x_7 \geq 4 \wedge \\
&x_2 + x_{10} \geq 11 \wedge \\
&x_3 + x_9 = 16 \wedge \\
&x_5 \geq x_8 + x_9 \wedge \\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
&b \rightarrow (x_6 \geq 7) \wedge \\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
&x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



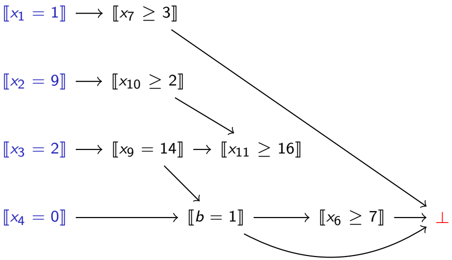
Learning



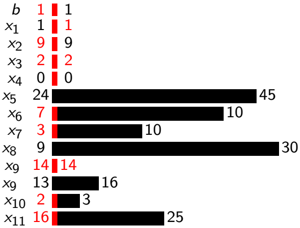
$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



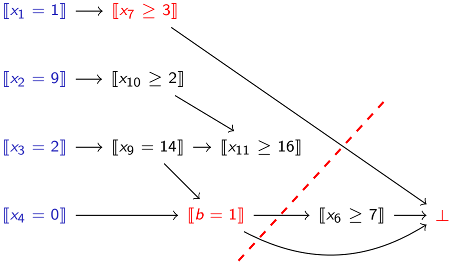
Learning



$x_1 + x_7 \geq 4 \wedge$
$x_2 + x_{10} \geq 11 \wedge$
$x_3 + x_9 = 16 \wedge$
$x_5 \geq x_8 + x_9 \wedge$
$b \leftrightarrow (x_9 - x_4 = 14) \wedge$
$b \rightarrow (x_6 \geq 7) \wedge$
$b \rightarrow (x_6 + x_7 \leq 9) \wedge$
$x_{11} \geq x_9 + x_{10}$

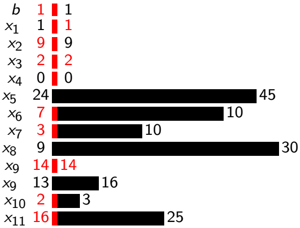


Learning

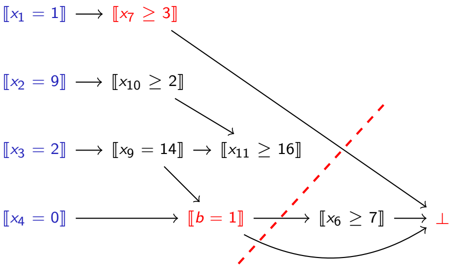


- Conflict analysis: $[\![b = 1]\!] \wedge [\![x_7 \geq 3]\!] \Rightarrow \bot$

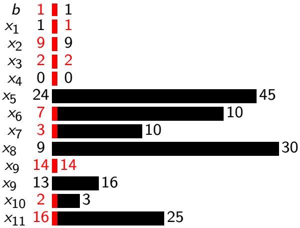
$$
\begin{aligned}
& x_1 + x_7 \geq 4 \wedge \\
& x_2 + x_{10} \geq 11 \wedge \\
& x_3 + x_9 = 16 \wedge \\
& x_5 \geq x_8 + x_9 \wedge \\
& b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
& b \rightarrow (x_6 \geq 7) \wedge \\
& b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
& x_{11} \geq x_9 + x_{10}
\end{aligned}
$$



Learning



$x_1 + x_7 \geq 4 \wedge$
$x_2 + x_{10} \geq 11 \wedge$
$x_3 + x_9 = 16 \wedge$
$x_5 \geq x_8 + x_9 \wedge$
$b \leftrightarrow (x_9 - x_4 = 14) \wedge$
$b \rightarrow (x_6 \geq 7) \wedge$
$b \rightarrow (x_6 + x_7 \leq 9) \wedge$
$x_{11} \geq x_9 + x_{10}$



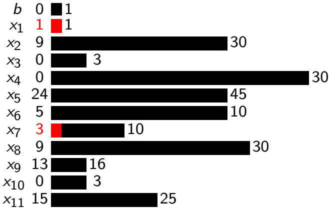
- Conflict analysis: $[\![b = 1]\!] \wedge [\![x_7 \geq 3]\!] \Rightarrow \perp$
- New clause: $[\![b \neq 0]\!] \vee [\![x_7 \leq 2]\!]$

Learning

$$\begin{aligned}
&x_1 + x_7 \geq 4 \wedge\\
&x_2 + x_{10} \geq 11 \wedge\\
&x_3 + x_9 = 16 \wedge\\
&x_5 \geq x_8 + x_9 \wedge\\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge\\
&b \rightarrow (x_6 \geq 7) \wedge\\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge\\
&x_{11} \geq x_9 + x_{10}
\end{aligned}$$

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!]$

- Conflict analysis: $[\![b = 1]\!] \wedge [\![x_7 \geq 3]\!] \Rightarrow \perp$
- New clause: $[\![b \neq 0]\!] \vee [\![x_7 \leq 2]\!]$
- Backtrack to level 1

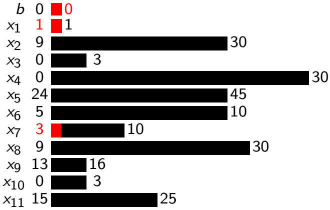


Learning

$$\begin{aligned}
&x_1 + x_7 \geq 4 \wedge \\
&x_2 + x_{10} \geq 11 \wedge \\
&x_3 + x_9 = 16 \wedge \\
&x_5 \geq x_8 + x_9 \wedge \\
&b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
&b \rightarrow (x_6 \geq 7) \wedge \\
&b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
&x_{11} \geq x_9 + x_{10}
\end{aligned}$$

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!] \longrightarrow [\![b = 0]\!]$

- Conflict analysis: $[\![b = 1]\!] \wedge [\![x_7 \geq 3]\!] \Rightarrow \bot$
- New clause: $[\![b \neq 0]\!] \vee [\![x_7 \leq 2]\!]$
- Backtrack to level 1
- Propagate the learnt clause

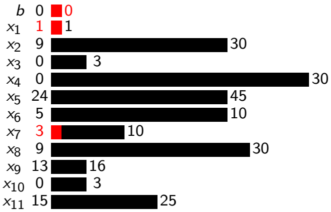


Learning

$[\![x_1 = 1]\!] \longrightarrow [\![x_7 \geq 3]\!] \longrightarrow [\![b = 0]\!]$

- Conflict analysis: $[\![b = 1]\!] \wedge [\![x_7 \geq 3]\!] \Rightarrow \bot$
- New clause: $[\![b \neq 0]\!] \vee [\![x_7 \leq 2]\!]$
- Backtrack to level 1
- Propagate the learnt clause
- Continue exploration

$$
\begin{array}{l}
x_1 + x_7 \geq 4 \wedge \\
x_2 + x_{10} \geq 11 \wedge \\
x_3 + x_9 = 16 \wedge \\
x_5 \geq x_8 + x_9 \wedge \\
b \leftrightarrow (x_9 - x_4 = 14) \wedge \\
b \rightarrow (x_6 \geq 7) \wedge \\
b \rightarrow (x_6 + x_7 \leq 9) \wedge \\
x_{11} \geq x_9 + x_{10}
\end{array}
$$



Learning in CP

- Hybrid CP/SAT
- Conflict Driven Clause Learning (CDCL) [Moskewicz et al., 2001]
- Based on the notion of **explanation**

Summary of the thesis

Summary of the thesis

Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Summary of the thesis

Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Contributions

- Propagation in a class of sequencing problems

Summary of the thesis

Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Contributions

- Propagation in a class of sequencing problems
- Search in car-sequencing

Summary of the thesis

Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Contributions

- Propagation in a class of sequencing problems
- Search in car-sequencing
- Learning in car-sequencing

Summary of the thesis

Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Contributions

- Propagation in a class of sequencing problems
- Search in car-sequencing
- Learning in car-sequencing
- Revisiting lazy generation

Summary of the thesis

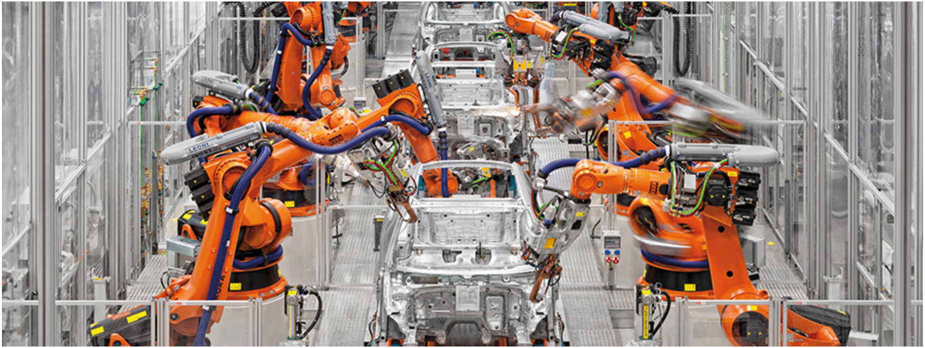
Modern CP-Solvers may not underestimate any of the three aspects: search, propagation, and learning

Contributions

- Propagation in a class of sequencing problems
- Search in car-sequencing
- Learning in car-sequencing
- Revisiting lazy generation
- Learning in disjunctive scheduling

Outline

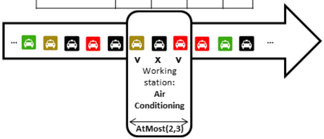
Car-Sequencing



- ROADEF'05 challenge [Solnon et al., 2008]
- RENAULT

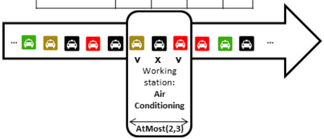
Problem Definition

Air Conditioning	V	X	V	X
Sunroof	X	V	X	V
MP3 Device	V	V	X	X



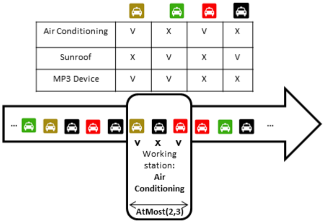
Problem Definition

Air Conditioning	V	X	V	X
Sunroof	X	V	X	V
MP3 Device	V	V	X	X



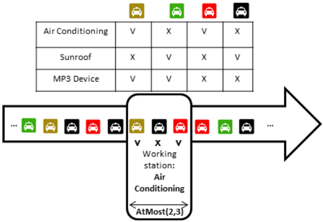
- A class of vehicles is defined by a set of options

Problem Definition



- A class of vehicles is defined by a set of options
- Each class is associated to a demand

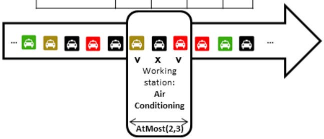
Problem Definition



- A class of vehicles is defined by a set of options
- Each class is associated to a demand
- `Capacity` constraints: no subsequence of size q may contain more than p vehicles requiring a given option

Problem Definition

Air Conditioning	V	X	V	X
Sunroof	X	V	X	V
MP3 Device	V	V	X	X



- A class of vehicles is defined by a set of options
- Each class is associated to a demand
- `Capacity` constraints: no subsequence of size q may contain more than p vehicles requiring a given option
- Is there a sequence of cars satisfying both demand and capacity constraints?

Outline

Propagation via AtMostSeqCard

Propagation via AtMostSeqCard

Definition

AtMostSeqCard$(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Propagation via AtMostSeqCard

Definition

AtMostSeqCard$(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} (\sum_{l=1}^{q} x_{i+l} \leq p) \wedge (\sum_{i=1}^{n} x_i = d)$$

Example AtMostSeqCard$(2, 5, 4, [x_1, \ldots, x_9])$

x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9

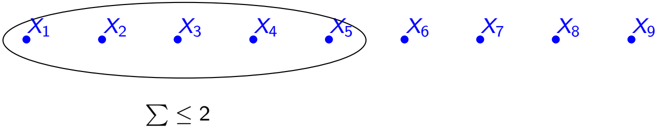
Propagation via AtMostSeqCard

Definition

$\text{AtMostSeqCard}(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p \right) \wedge \left(\sum_{i=1}^{n} x_i = d \right)$$

Example $\text{AtMostSeqCard}(2, 5, 4, [x_1, \ldots, x_9])$



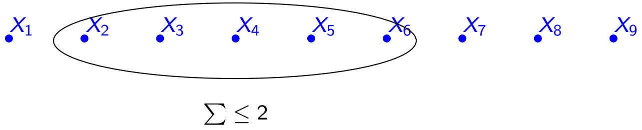
Propagation via AtMostSeqCard

Definition

$\textsc{AtMostSeqCard}(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example $\textsc{AtMostSeqCard}(2, 5, 4, [x_1, \ldots, x_9])$



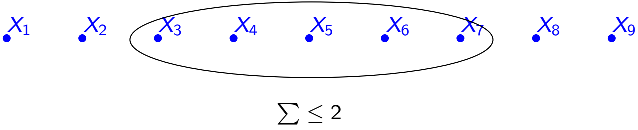
Propagation via AtMostSeqCard

Definition

$\text{AtMostSeqCard}(p, q, d, [x_1, \dots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example $\text{AtMostSeqCard}(2, 5, 4, [x_1, \dots, x_9])$



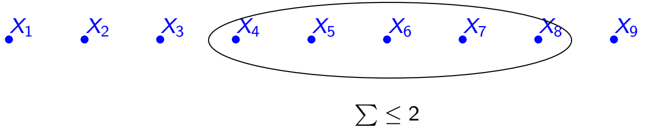
Propagation via AtMostSeqCard

Definition

$\textsc{AtMostSeqCard}(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example $\textsc{AtMostSeqCard}(2, 5, 4, [x_1, \ldots, x_9])$



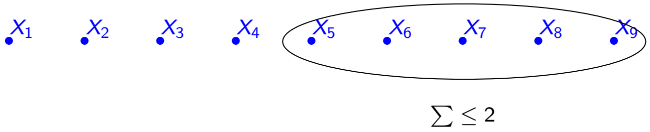
Propagation via ATMOSTSEQCARD

Definition

ATMOSTSEQCARD$(p, q, d, [x_1, \dots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} (\sum_{l=1}^{q} x_{i+l} \leq p) \wedge (\sum_{i=1}^{n} x_i = d)$$

Example ATMOSTSEQCARD$(2, 5, 4, [x_1, \dots, x_9])$



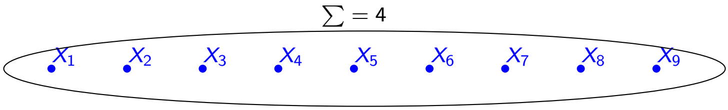
Propagation via AtMostSeqCard

Definition

AtMostSeqCard$(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example AtMostSeqCard$(2, 5, 4, [x_1, \ldots, x_9])$



Propagation via AtMostSeqCard

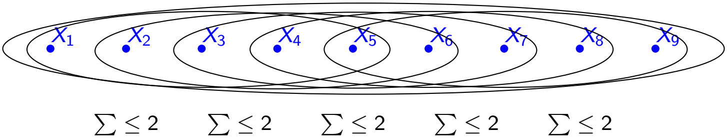
Definition

AtMostSeqCard$(p, q, d, [x_1, \dots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example AtMostSeqCard$(2, 5, 4, [x_1, \dots, x_9])$

$\sum = 4$



Propagation via AtMostSeqCard

Definition

AtMostSeqCard$(p, q, d, [x_1, \ldots, x_n]) \Leftrightarrow$

$$\bigwedge_{i=0}^{n-q} \left(\sum_{l=1}^{q} x_{i+l} \leq p\right) \wedge \left(\sum_{i=1}^{n} x_i = d\right)$$

Example AtMostSeqCard$(2, 5, 4, [x_1, \ldots, x_9])$

x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9

- Car sequencing
- Crew-Rostering/Timetabling

Arc Consistency on AtMostSeqCard

Definition

A constraint C is Arc Consistent (AC) iff for every x in the scope of C, for every value $v \in \mathcal{D}(x)$ there exists an assignment w in $\mathcal{D}$ satisfying C in which v is assigned to x

Arc Consistency on ATMOSTSEQCARD

Definition

A constraint C is Arc Consistent (AC) iff for every x in the scope of C, for every value $v \in \mathcal{D}(x)$ there exists an assignment w in $\mathcal{D}$ satisfying C in which v is assigned to x

- ATMOSTSEQ $\oplus$ CARDINALITY is not enough

Arc Consistency on AtMostSeqCard

Definition

A constraint C is Arc Consistent (AC) iff for every x in the scope of C, for every value $v \in \mathcal{D}(x)$ there exists an assignment w in $\mathcal{D}$ satisfying C in which v is assigned to x

- AtMostSeq $\oplus$ Cardinality is not enough

AtMostSeqCard as a particular case?

- cost-Regular: $O(2^q n)$ [van Hoeve et al., 2009]
- Gen-Sequence: $O(n^3)$ [van Hoeve et al., 2009]
- Gen-Sequence: $O(n^2.log(n))$ down a branch $\oplus$ initial compilation of $O(q.n^2)$. [Maher et al., 2008].

Arc Consistency on AtMostSeqCard

An example with $\textsc{AtMostSeqCard}(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 ■ 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 ▮ 1

1 0 **1** **1** **1** 0 0 0 0 1 0

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 ▮ 1

1 0 **1** **1** **1** 0 0 0 0 1 0 max added= 4

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 1

1 0 **1** **1** **1** 0 0 0 0 1 0 max added= 4

1 **1** 0 0 0 0 **1** **1** **1** 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 1

1 0 **1** **1** **1** 0 0 0 0 1 0 max added= 4

max added 5 **1** **1** 0 0 0 0 **1** **1** **1** 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \dots, x_{22}])$

. 0 . . . . 0 1 0 1

. 0 . . . . 0 1 0 ■ 1

1 0 **1** **1** **1** 0 0 0 0 1 0 max added= 4

max added 5 **1** **1** 0 0 0 0 **1** **1** **1** 1

Maximum possible 9 < residual demand

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 1

1 0 **1 1 1** 0 0 0 0 1 0 max added= 4

max added 5 **1 1** 0 0 0 0 **1 1 1** 1

Maximum possible 9 $<$ residual demand

. 0 . . . . . . 0 1 0 1 1

Arc Consistency on AtMostSeqCard

An example with AtMostSeqCard$(4, 8, 12, [x_1, \ldots, x_{22}])$

. 0 . . . . . . 0 1 0 1

. 0 . . . . . . 0 1 0 ■ 1

1 0 **1 1 1** 0 0 0 0 1 0 max added= 4

max added 5 **1 1** 0 0 0 0 **1 1 1** 1

Maximum possible 9 < residual demand

. 0 . . . . . . 0 1 0 1 1

- Arc Consistency in $O(n)$ time (optimal)
- Extremely efficient in practice (Car-Sequencing + Crew Rostering)

Outline

Hybrid CP/SAT Models

- Models based on AtMostSeqCard

Hybrid CP/SAT Models

- Models based on AtMostSeqCard
- We have to explain AtMostSeqCard

Hybrid CP/SAT Models

- Models based on AtMostSeqCard
- We have to explain AtMostSeqCard

Explaining AtMostSeqCard?

- Explain AtMostSeq and Cardinality
- Explaining the extra filtering: consider the naive explanation, then try to reduce it.

$$\mathcal{D}: 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \ldots\ 1$$

Hybrid CP/SAT Models

- Models based on AtMostSeqCard
- We have to explain AtMostSeqCard

Explaining AtMostSeqCard?

- Explain AtMostSeq and Cardinality
- Explaining the extra filtering: consider the naive explanation, then try to reduce it.

$$\mathcal{D}: 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \ldots\ 1$$

Hybrid CP/SAT Models

- Models based on AtMostSeqCard
- We have to explain AtMostSeqCard

Explaining AtMostSeqCard?

- Explain AtMostSeq and Cardinality
- Explaining the extra filtering: consider the naive explanation, then try to reduce it.

$$\mathcal{D}: 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \ldots\ 1$$

Hybrid CP/SAT Models

- Models based on AtMostSeqCard
- We have to explain AtMostSeqCard

Explaining AtMostSeqCard?

- Explain AtMostSeq and Cardinality
- Explaining the extra filtering: consider the naive explanation, then try to reduce it.

$\mathcal{D}$: 1 1 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 1 0 0 . . . 1

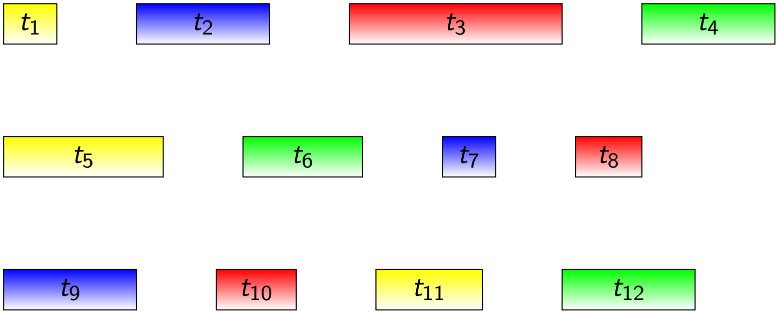
$\widehat{\mathcal{D}}$: 1 1 . . . . . . 1 1 . . . . 0 0 0 0 . 0 0 . . . 1

Experimental Results

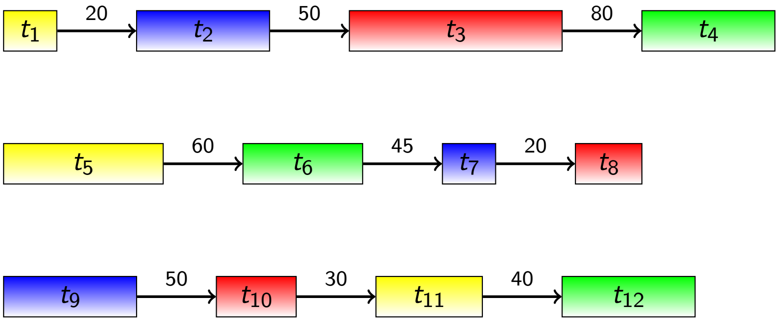
- CP, SAT, Hybrid CP/SAT models
- Finding solutions quickly: Propagation is very important to find solutions quickly when they exist.
- For proving unsatisfiability: Clause learning is by far the most critical factor.

Outline

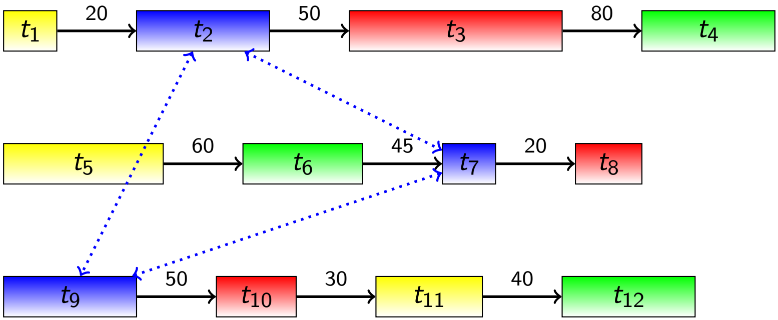
Disjunctive Scheduling



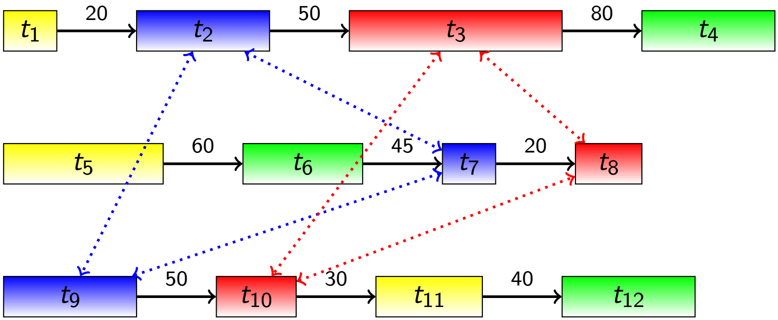
Disjunctive Scheduling



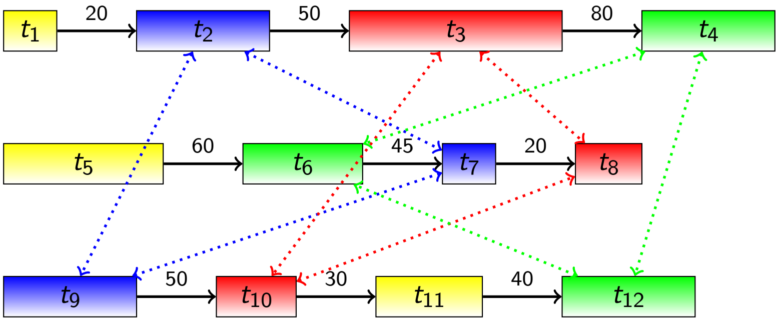
Disjunctive Scheduling



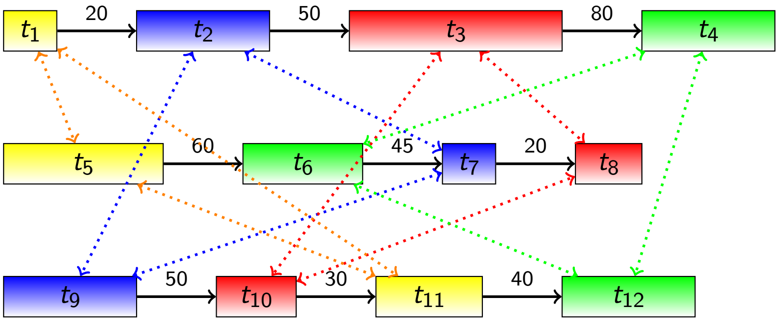
Disjunctive Scheduling



Disjunctive Scheduling



Disjunctive Scheduling



Formulation

Unary Resource Constraint

Formulation

Unary Resource Constraint

- Decomposition using the following DISJUNCTIVE constraints:

$$\delta_{kij} = \begin{cases} 0 & \Leftrightarrow \quad t_{ik} + p_{ik} \leq t_{jk} \\ 1 & \Leftrightarrow \quad t_{jk} + p_{jk} \leq t_{ik} \end{cases} \tag{1}$$

Disjunctive-based Learning

Disjunctive-based Learning

Conflict analysis in two phases:

1. Standard 1-UIP cut
2. Apply resolution for every bound literal until having a nogood with only reified Boolean variables

Disjunctive-based Learning

Disjunctive-based Learning

Conflict analysis in two phases:

1. Standard 1-UIP cut
2. Apply resolution for every bound literal until having a nogood with only reified Boolean variables

- ⊕ No domain encoding
- ⊕ Scheduling horizon does not manner in size
- ⊖ Language of literals is restricted compared to standard approaches

DISJUNCTIVE-based Learning

DISJUNCTIVE-based Learning

Conflict analysis in two phases:

1. Standard 1-UIP cut
2. Apply resolution for every bound literal until having a nogood with only reified Boolean variables

- ⊕ No domain encoding
- ⊕ Scheduling horizon does not manner in size
- ⊖ Language of literals is restricted compared to standard approaches

tai13		tai21		tai23		tai25		tai26		tai29		tai30	
new	old	new	old	new	old	new	old	new	old	new	old	new	old
1305	1282	1613	1573	1514	1474	1543	1518	1561	1558	1573	1525	1508	1485

Outline

Summary

- Contributions to each of the three aspects of constraint programming that are ‘search’, ‘propagation’ and ‘learning’ for efficiently solving sequencing and scheduling problems.

Summary

- Contributions to each of the three aspects of constraint programming that are ‘search’, ‘propagation’ and ‘learning’ for efficiently solving sequencing and scheduling problems.
- Case study: car-sequencing

Summary

- Contributions to each of the three aspects of constraint programming that are 'search', 'propagation' and 'learning' for efficiently solving sequencing and scheduling problems.
- Case study: car-sequencing
- Clause Learning in CP

Summary

- Contributions to each of the three aspects of constraint programming that are 'search', 'propagation' and 'learning' for efficiently solving sequencing and scheduling problems.
- Case study: car-sequencing
- Clause Learning in CP

Modern constraint programming solvers may not underestimate any of these three aspects

Summary

- Contributions to each of the three aspects of constraint programming that are 'search', 'propagation' and 'learning' for efficiently solving sequencing and scheduling problems.
- Case study: car-sequencing
- Clause Learning in CP

Modern constraint programming solvers may not underestimate any of these three aspects

Future Research

- (Car-Sequencing) Application to 'real' industrial situations?
- More extensions for ATMOSTSEQCARD?
- Hand crafted learning?

Thank you for your attention!

Special thanks to my co-authors..

- Christian Artigues
- Emmanuel Hebrard
- Marie-Jose Huguet
- Valentin Mayer-Eichberger
- Nina Narodytska
- Thierry Petit
- Toby Walsh

References I

Maher, M. J., Narodytska, N., Quimper, C., and Walsh, T. (2008).
Flow-based propagators for the SEQUENCE and related global constraints.
In *Proceedings of the 14th International Conference on Principles and Practice of Constraint Programming, CP'08, Sydney, NSW, Australia*, pages 159–174.

Moskewicz, M. W., Madigan, C. F., Zhao, Y., Zhang, L., and Malik, S. (2001).
Chaff: Engineering an Efficient SAT Solver.
In *Proceedings of the 38th Annual Design Automation Conference, DAC'01, Las Vegas, Nevada, USA*, pages 530–535.

Solnon, C., Cung, V., Nguyen, A., and Artigues, C. (2008).
The car sequencing problem: Overview of state-of-the-art methods and industrial case-study of the roadef'2005 challenge problem.
European Journal of Operational Research, 191(3):912–927.

van Hoeve, W. J., Pesant, G., Rousseau, L., and Sabharwal, A. (2009).
New filtering algorithms for combinations of among constraints.
Constraints, 14(2):273–292.