MDD Propagation for Sequence Constraints

Andre A. Cire and Willem-Jan van Hoeve

Tepper School of Business, Carnegie Mellon University

Sequence constraint

Example: Employee must work between 4 and 7 days every 9 consecutive days

sun	mon	tue	wed	thu	fri	sat	sun	mon	tue	wed	thu
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}

$4 \le x_1+x_2+ \dots +x_9 \le 7 \quad \wedge$

$4 \le x_2+x_3+ \dots +x_{10} \le 7 \quad \wedge$

$4 \le x_3+x_4+ \dots +x_{11} \le 7 \quad \wedge$

$4 \le x_4+x_5+ \dots +x_{12} \le 7$

=: *Sequence*($[x_1,x_2,\dots,x_{12}]$, q=9, S={1}, l=4, u=7)

$$\textit{Sequence}(X, q, S, l, u) := \bigwedge_{|X'|=q} \; l \le \textstyle\sum_{x \in X'} (x \in S) \le u$$

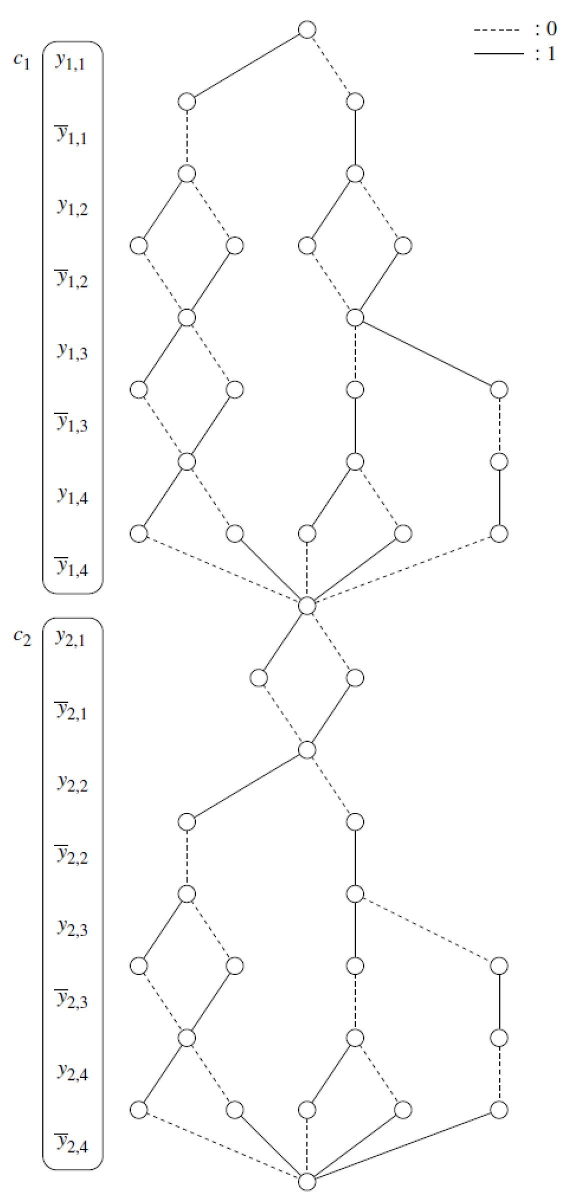
MDD-Based CP

- Maintain limited-width MDD
  - Serves as relaxation
  - Typically start with width 1 (initial variable domains)
  - Dynamically adjust MDD, based on constraints
- Constraint Propagation
  - Edge filtering: Remove provably inconsistent edges (those that do not participate in any solution)
  - Node refinement: Split nodes to separate edge information
- Search
  - As in classical CP, but can be guided by MDD

Hardness of MDD Consistency

Goal: Given an arbitrary MDD and a *Sequence* constraint, remove *all* inconsistent edges from the MDD

Theorem: Establishing MDD consistency for *Sequence* on an arbitrary MDD is NP-hard



Proof: Reduction from 3-SAT

- Literal x_j in clause c_i represented by variable y_{ij}
- MDD size $O(6(2mn+1))$
- Ensure that a variable takes the same value in each clause:

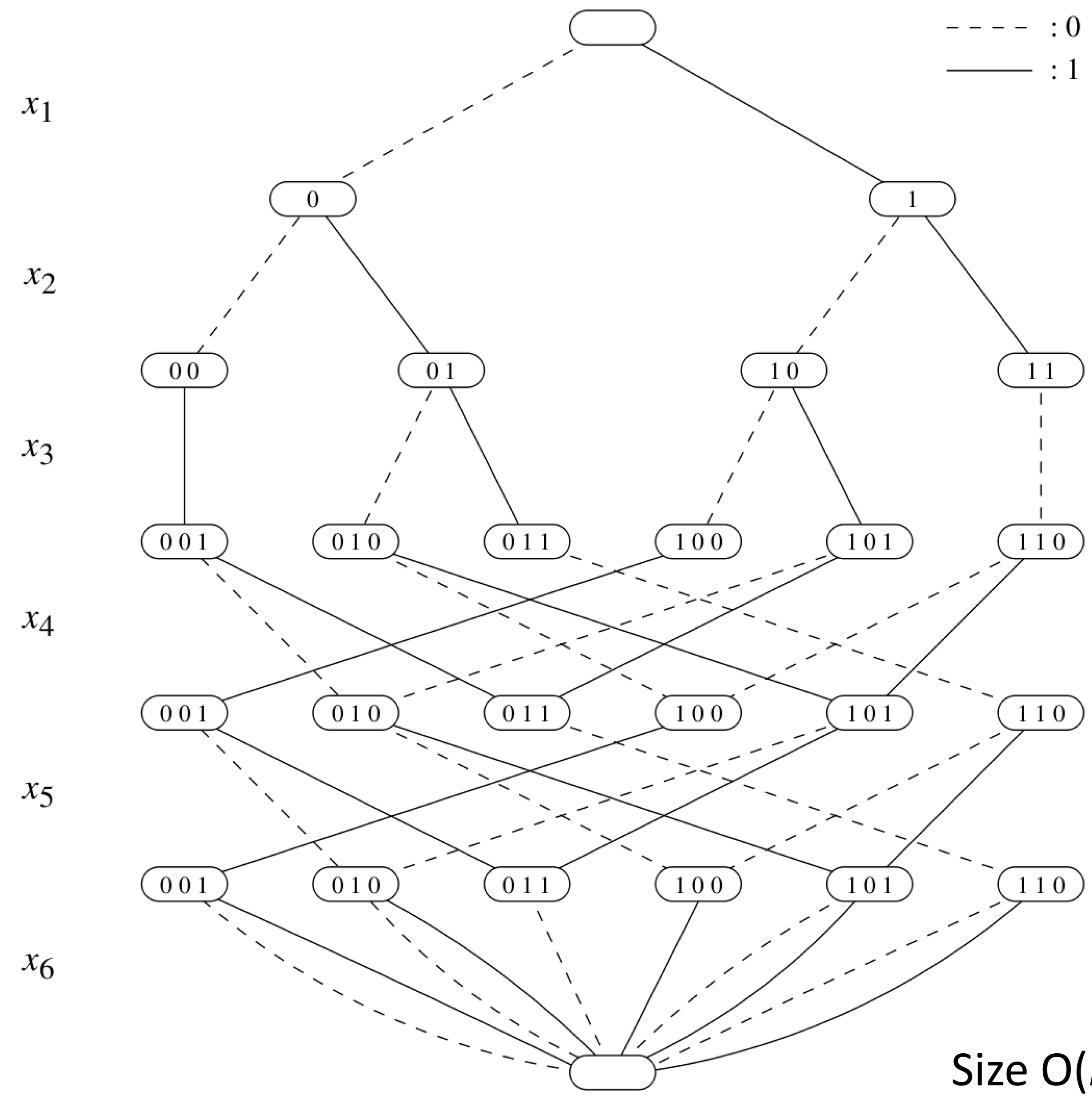
Sequence(Y, q=2n, S={1}, l=n, u=n)

Example: $c_1 = (x_1 \vee \overline{x}_3 \vee x_4)$

$c_2 = (x_2 \vee x_3 \vee \overline{x}_4)$

Exact MDD Representation

Exact MDD for *Sequence*(X, q=3, S={1}, l=1, u=2)



Size $O(n2^q)$

Partial MDD Propagation

Based on decomposition:

- *Sequence*(X, q, S, l, u) with $X = x_1, x_2, \dots, x_n$
- Introduce a 'cumulative' variable y_i representing the sum of the first i variables in X

  $y_0 = 0$

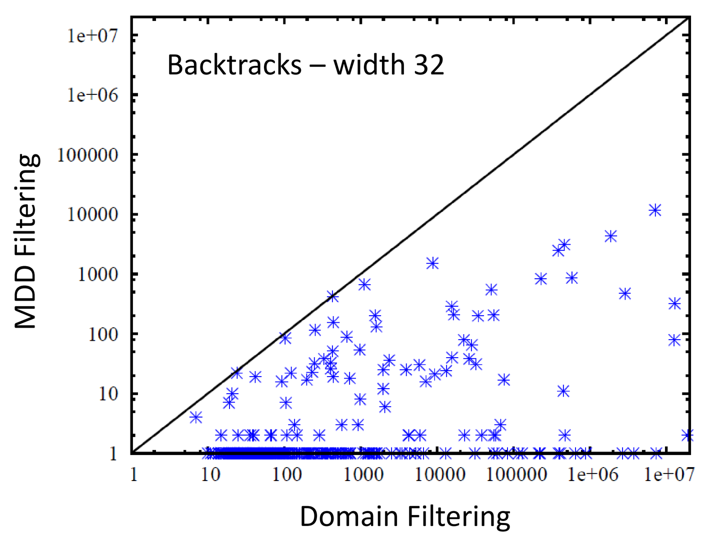
  $y_i = y_{i-1} + (x_i \in S)$ for i=1..n
- Then the sub-constraint on $[x_{i+1},\dots, x_{i+q}]$ is equivalent to

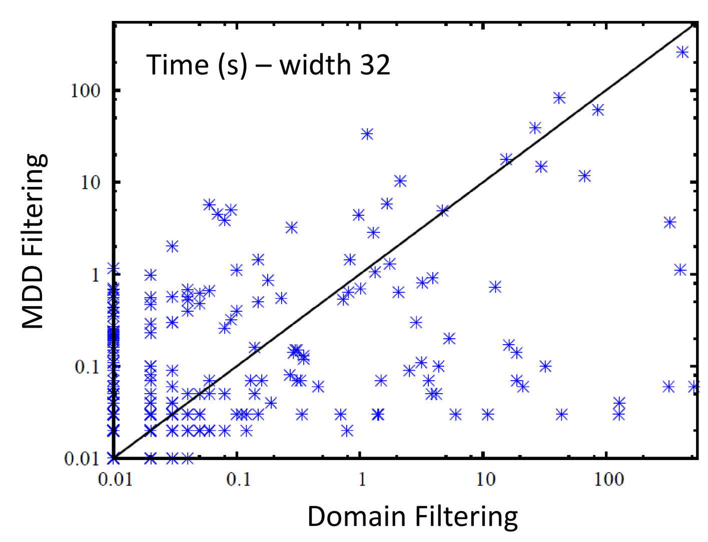
  $l \le y_{i+q} - y_i$

  $y_{i+q} - y_i \le u$ for i = 0..n-q
- Variables y represented at the *nodes* of the MDD

Experimental Results

Random instances





Shift scheduling problems

Instance		Domain filtering		MDD - width 1		MDD - width 2		MDD - width 8	
	n	backtracks	time	backtracks	time	backtracks	time	backtracks	time
Type P-I	40	17,054	0.36	17,054	0.61	1,213	0.07	0	0.00
	50	17,054	0.42	17,054	0.75	1,213	0.09	0	0.00
	60	17,054	0.54	17,054	0.90	1,213	0.11	0	0.01
	70	17,054	0.58	17,054	1.04	1,213	0.12	0	0.01
	80	17,054	0.66	17,054	1.26	1,213	0.15	0	0.01
Type P-II	40	126,406	2.00	126,406	4.66	852	0.08	0	0.00
	50	126,406	2.36	126,406	5.90	852	0.09	0	0.00
	60	126,406	2.86	126,406	7.43	852	0.11	0	0.00
	70	126,406	3.04	126,406	8.38	852	0.13	0	0.01
	80	126,406	3.48	126,406	9.46	852	0.15	0	0.01