SAT Encodings for the Car Sequencing Problem

Valentin Mayer-Eichberger and Toby Walsh

16/09/2013 Doctoral Program at CP

Car Sequencing

Source Wikipedia

- Cars require different options (air-conditioning, sun-roof, etc.)
- Is there a production sequence for cars on the assembly line satisfying the sliding capacity constraints?
- CSPLib Benchmark Nr. 1
- CP, IP, local search

Car Sequencing: Example

- Classes $C = \{1, 2, 3\}$ with demand $d_1 = 3, d_2 = 2, d_3 = 2$
- Options $O = \{a, b\}$ with capacity constraints 1/2 and 1/5
- Class 1 no restriction
- Class 2 requires option a
- Class 3 requires option a and b

Sequence of cars	3	1	2	1	2	1	3
Option a	1	-	1	-	1	-	1
Option b	1	-	-	-	-	-	1

Car Sequencing: Example

- Classes $C = \{1, 2, 3\}$ with demand $d_1 = 3, d_2 = 2, d_3 = 2$
- Options $O = \{a, b\}$ with capacity constraints $1/2$ and $1/5$
- Class 1 no restriction
- Class 2 requires option a
- Class 3 requires option a and b

Sequence of cars	3	1	2	1	2	1	3
Option a	1	-	1	-	1	-	1
Option b	1	-	-	-	-	-	1

Car Sequencing is NP-Complete

The SAT Approach: The Ultimate Decomposition

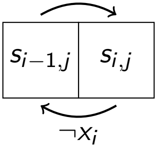
- ONE constraint, the clause (e.g. $a \vee b \vee \neg c$).
- ONE propagator, unit propagation: (e.g. a and $\neg a \vee b$ then propagate b).
- Using SAT solvers as blackboxes.
- Challenge: Finding good CNF representations.

The SAT Approach: The Ultimate Decomposition

- ONE constraint, the clause (e.g. $a \vee b \vee \neg c$).
- ONE propagator, unit propagation: (e.g. a and $\neg a \vee b$ then propagate b).
- Using SAT solvers as blackboxes.
- Challenge: Finding good CNF representations.
- Global constraint: $(\sum_{i=1}^{n} x_i = d) \wedge \bigwedge_{i=0}^{n-q} (\sum_{l=1}^{q} x_{i+l} \leq u)$
- Use cumulative sums:

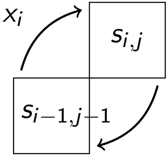
$$s_{i,j} \iff (j \leq \sum_{l=1}^{i} x_l)$$

Sequential Counter



$$\neg s_{i-1,j} \vee s_{i,j}$$

$$x_i \vee \neg s_{i,j} \vee s_{i-1,j}$$



$$\neg x_i \vee \neg s_{i-1,j-1} \vee s_{i,j}$$

$$\neg s_{i,j} \vee s_{i-1,j-1}$$

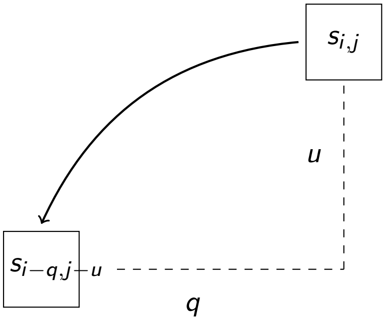
- This idea can translate all cardinality constraints

Demand Constraint + Capacity Constraint

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

Demand Constraint + Capacity Constraint

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



$$\neg s_{i,j} \vee s_{i-q,j-u}$$

Results on CSPLib

- 30+9 hard solved 28 within 20min
- Largest: 400 cars, 5 options, 23 classes: 200K Var, 1M Clauses
- Several variations of this encoding

	E1	E2	E3
#solved UNSAT	17	15	17
#fastest UNSAT	5	4	4
#solved SAT	11	11	11
#fastest SAT	0	4	7

- With another trick 36 decision problems in the CSPLib can be solved (3 left open)
- Decoder and encoder available `github.com/vale1410/car-sequencing`

Related Work

- Sinz: Sequential Counter CNF [6]
- Een and Soerensson: Translation through BDDs to CNF [3]
- Bacchus: Decomposition through DFAs to CNF [1]
- Brand et al: Decomposition to cumulative sums for CP [2]
- Siala et al: Linear time propagator for CP [5]

Conclusions

Conclusion

- SAT is strong on instances of the CSPLib
- Global Constraints motivate for encodings
- Choosing the right encoding of cardinality constraints is crucial
- SAT can be very competitive on CP benchmarks

Current and Future work

- Fair Comparison to CP, IP, ASP, LS ...
- Elegant proof of GAC and lower bound on size.
- Idea useful in rostering, planning, scheduling?
- Exponential encoding in the number of options?
- Treat the optimization problem.

Bibliography

- Fahiem Bacchus.
  GAC Via Unit Propagation.
  In *CP*, pages 133–147, 2007.

- Sebastian Brand, Nina Narodytska, Claude-Guy Quimper, Peter J. Stuckey, and Toby Walsh.
  Encodings of the Sequence Constraint.
  In *CP*, pages 210–224, 2007.

- Niklas Eén and Niklas Sörensson.
  Translating Pseudo-Boolean Constraints into SAT.
  Journal on Satisfiability, Boolean Modeling and Computation, 2(1-4):1–26, 2006.

- Ian P. Gent.
  Two Results on Car-sequencing Problems.
  In *Report APES-02-1998*, 1998.

- Mohamed Siala, Emmanuel Hebrard, and Marie-José Huguet.
  An Optimal Arc Consistency Algorithm for a Chain of Atmost

Backupslides

Capacity Constraints: Example

Capacity constraint $4/8$, demand $d = 12$ on a sequence of 22 variables:

13																					U	U	U
12																				U	?	?	L
11																			U	?	?	L	
10																		U	?	?	L		
9													U	U	U	U	U	?	?	L			
8												U	?	?	L	L	L	L	L				
7											U	?	?	L									
6										U	?	?	L										
5					U	U	U	U	U	?	?	L											
4				U	?	?	L	L	L	L	L												
3			U	?	?	L																	
2		U	?	?	L																		
1	U	?	?	L																			
0	L	L	L																				
$s_{i,j}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
x_i								**0**	**0**								**0**	**0**					

Capacity Constraints: Example

Partial Assignment: x_1 and x_{13} to true and x_{12}, x_{14} and x_{21} to false.

13																				U	U	U
12																			U	0	0	L
11																		U	?	1	L	
10																	U	?	1	L		
9													U	U	U	U	U	?	1	L			
8												U	0	1	L	L	L	L	L				
7											U	1	1	L									
6										U	1	1	L										
5					U	U	U	U	U	1	1	L											
4				U	?	1	L	L	L	L	L												
3			U	1	1	L																	
2		U	1	1	L																		
1	U	1	1	L																			
0	L	L	L																				
$s_{i,j}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
x_i		1	**1**	**1**			**0**	(0)	(0)	**1**	**1**	**1**	0	1	0	(0)	(0)					0	**1**

Sequential Counter: Comparison to Sinzs AtMost

$\forall i \in \{1 \dots n\}\ \forall j \in \{0 \dots d+1\}$:

$$\neg s_{i-1,j} \vee s_{i,j} \tag{1}$$

$$x_i \vee \neg s_{i,j} \vee s_{i-1,j} \tag{2}$$

$\forall i \in \{1 \dots n\} \forall j \in \{1 \dots d+1\}$:

$$\neg s_{i,j} \vee s_{i-1,j-1} \tag{3}$$

$$\neg x_i \vee \neg s_{i-1,j-1} \vee s_{i,j} \tag{4}$$

$$s_{0,0} \wedge \neg s_{0,1} \wedge \neg s_{n,d+1} \tag{5}$$

SAT instances

Inst(SAT)	E1	E2	E3	ASP	PB
4/72	0.14	0.17	**0.05**	6.78	-
16/81	0.38	**0.08**	0.14	13.25	-
41/66	0.06	**0.04**	0.04	2.55	123.41
26/82	0.95	**0.16**	0.21	80.06	654.80
200-01	30.43	47.70	**15.35**	1141.87	-
200-07	6.32	2.39	**1.46**	1478.01	-
300-01	143.76	10.62	**5.98**	810.23	-
300-07	59.86	28.39	**8.24**	-	-
400-05	623.30*	768.97	846.41	-	-
400-06	445.36	24.79	**16.29**	-	-
400-10	884.91	18.99	**13.50**	-	-

UNSAT instances

Inst(UNSAT)	E1	E2	E3	ASP	PBO
6/76	72.55	117.28	**57.55**	929.87	289.68
10/93	11.40	**6.48**	11.08	331.31	-
21/90	119.83	**74.01**	95.18	-	-
36/92	**16.97**	18.67	41.34	277.63	-
200-03	137.21	**24.02**	30.81	-	-
200-04	69.64	475.76	16.83	-	**1.84**
200-05	**254.03**	1337.39*	1172.38*	-	-
200-09	358.81	-	504.26	-	**3.77**
200-10	**2.10**	3.36	2.53	3.91	2.32
300-03	**99.03**	-	214.41	949.55	-
300-04	3.17	30.57	**2.03**	46.60	4.40
300-08	18.52	799.16	50.98	123.22	**13.54**
300-05	**0.37**	2.73	0.62	-	-
300-10	1.08	25.15	**0.96**	1282.09	904.31
400-03	37.05	**30.31**	31.47	-	-
400-04	13.03	185.32	**6.33**	130.33	14.47
400-09	25.75	470.60	32.90	557.60	**5.19**

lower bounds

	LB (pre)	LB (SAT)	sec	UB (SAT)	sec	LB* (known)	UB* (known)
4/72		0	0.07	0	0.07	0	0
6/76		6	209.77	6	0.10	1	6
10/93		1	18.93	3	0.53	1	3
16/81		0	-	0	0.07	0	0
19/71	2	-	-	2	1.50	2	2
21/90	2	1	93.80	2	0.11	1	2
36/92		1	38.55	1	0.07	1	2
41/66		0	-	0	0.06	0	0
26/82		0	-	0	0.14	0	0
200-01		0	-	0	7.46		0
200-02	2	-	-	2	0.86		2
200-03		3	1323.05	3	110.33		3
200-04	7	1	17.59	7	1.04		7
200-05		1	639.42	3	39.63		6
200-06	6	-	-	6	0.69		6
200-07		0	-	0	0.69		0
200-08	8	-	-	8	20.17		8
200-09	10	1	189.14	10	2.32		10
200-10	17	16	213.88	17	3.51		19
300-01		0	-	0	24.83		0
300-02		-	-	6	39.89		12
300-03	13	2	872.06	13	77.76		13
300-04	7	6	795.68	7	12.20		7
300-05	2	12	1145.39	16	1247.82		27
300-06	2	-	-	2	1559.76		2
300-07		0	-	0	6.33		0
300-08	8	1	102.26	8	1.01		8
300-09	7	-	-	7	141.56		7
300-10	3	9	863.15	13	115.67		21
400-01		-	-	-	-		1
400-02	15	-	-	15	112.36		15
400-03		10	1531.53				9
400-04	19	5	25.85	19	222.68		19
400-05		0	-	0	302.19		0

Size

Length	Variables					Clauses				
	E1	E2	E3	ASP	PB	E1	E2	E3	ASP	PB
100	27	21	27	90	14	91	83	99	243	54
200	73	60	73	335	33	256	259	291	946	127
300	136	118	136	747	48	495	524	573	2195	197
400	223	199	223	1308	63	827	907	972	3879	271

Link between Cars and Options

$\forall i \in \{1 \ldots n\}$:

$$\bigwedge_{\substack{k \in C \\ l \in O_k}} \neg c_i^k \vee o_i^l \tag{6}$$

$$\bigwedge_{\substack{k \in C \\ l \notin O_k}} \neg c_i^k \vee \neg o_i^l \tag{7}$$

$$\bigwedge_{l \in O} \left(\neg o_i^l \vee \bigvee_{k \in C_l} c_i^k \right) \tag{8}$$

Example for non GAC of E2

Example

Let $n = 5$, $d = 2$ with a capacity constraint of $1/2$, and let x_3 be true, then unit propagation does not force x_2 nor x_4 to false. Setting them to true will lead to a conflict through clauses (12) and (**??**) on positions 2, 3 and 4.

3				U	U	U
2		U	U	.	.	L
1	U	.	.	L	L	
0	L	L	L			
$s_{i,j}$	0	1	2	3	4	5
x_i		.	.	1	.	.

Sequential Counter

$\forall i \in \{1 \dots n\}\ \forall j \in \{0 \dots d+1\}$:

$$\neg s_{i-1,j} \vee s_{i,j} \tag{9}$$

$$x_i \vee \neg s_{i,j} \vee s_{i-1,j} \tag{10}$$

$\forall i \in \{1 \dots n\} \forall j \in \{1 \dots d+1\}$:

$$\neg s_{i,j} \vee s_{i-1,j-1} \tag{11}$$

$$\neg x_i \vee \neg s_{i-1,j-1} \vee s_{i,j} \tag{12}$$

$$s_{0,0} \wedge \neg s_{0,1} \wedge s_{n,d} \wedge \neg s_{n,d+1} \tag{13}$$

A Trick for Lower Bounds ([4])

class	0	1	2	3	4	5	6	7	8	9	10	11
demand	9	4	22	2	1	62	31	4	24	4	3	36
0: 1/2	-	-	-	-	-	-	-	-	-	-	-	x
1: 2/3	-	-	-	-	-	x	x	x	x	x	x	-
2: 1/3	-	-	x	x	x	-	-	-	x	x	x	-
3: 2/5	-	x	-	-	x	-	x	x	-	-	x	-
4: 1/5	x	x	-	x	x	-	-	x	-	x	x	-

class	12	13	14	15	16	17	18	19	20	**21**	22	**23**
demand	3	25	3	8	5	2	6	21	5	**7**	11	**2**
0: 1/2	x	x	x	x	x	x	x	x	x	**x**	x	**x**
1: 2/3	-	-	-	-	-	-	x	x	x	**x**	x	**x**
2: 1/3	-	-	-	x	x	x	-	-	-	**x**	x	**x**
3: 2/5	-	x	x	-	-	x	-	x	x	**-**	x	**x**
4: 1/5	x	-	x	-	x	x	x	-	x	**x**	-	**x**

- Class 21 and 23 have option 0,1,2,4 with a total demand of 9
- All other classes share at least one option with 21 and 23
- Potential neighbours of 21 and 23?