Compact Representation of Sets of Binary Constraints

Jussi Rintanen

NATIONAL ICT AUSTRALIA LIMITED

August 30, 2006

Motivation: planning

- Practically all implementations of planning as satisfiability, have used a quadratic size translation from a problem instance to SAT.
- Recently Rintanen, Heljanko & Niemelä (AIJ 06 or 07) have given linear size translations which help scale up to much bigger problems than earlier.
- Invariants/mutexes, an important (but not logically necessary) part of efficient SAT planning, have quadratic size.
  This, as the only quadratic part of the formulae, is sometimes an obstacle to scalability: formulas have sizes of several gigabytes.

Motivation: planning

- Practically all implementations of planning as satisfiability, have used a quadratic size translation from a problem instance to SAT.
- Recently Rintanen, Heljanko & Niemelä (AIJ 06 or 07) have given linear size translations which help scale up to much bigger problems than earlier.
- Invariants/mutexes, an important (but not logically necessary) part of efficient SAT planning, have quadratic size.
  This, as the only quadratic part of the formulae, is sometimes an obstacle to scalability: formulas have sizes of several gigabytes.

Motivation: planning

- Practically all implementations of planning as satisfiability, have used a quadratic size translation from a problem instance to SAT.
- Recently Rintanen, Heljanko & Niemelä (AIJ 06 or 07) have given linear size translations which help scale up to much bigger problems than earlier.
- Invariants/mutexes, an important (but not logically necessary) part of efficient SAT planning, have quadratic size.
  This, as the only quadratic part of the formulae, is sometimes an obstacle to scalability: formulas have sizes of several gigabytes.

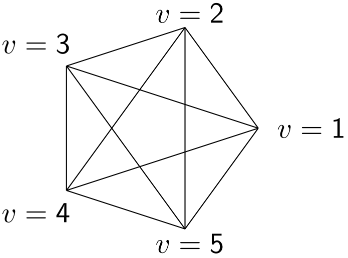
Motivation: general problem

- A binary relation (graph) on a set of n objects may have n^2 elements (edges).
- If the relation/graph is dense and n is high ($10^4 >$) the number of elements/edges can be very high ($10^8 >$).
- The representation of the elements/edges may become impractical.
- Goal: succinct representation of the relation/graph.

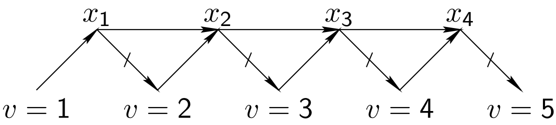
Cliques in constraint graphs

Definition

Let $\langle N, E \rangle$ be an undirected graph. Then *a clique* is $C \subseteq N$ such that $\{n, n'\} \in E$ for every $n, n' \in C$ such that $n \neq n'$.



Representation with $O(n)$ Size and $O(n)$ Auxiliary Variables



[Rintanen et al. 2005]

Representation with $\mathcal{O}(n \log n)$ size and $\mathcal{O}(\log n)$ auxiliary variables

Let $C = \{l_0, l_2, l_3, l_4, l_5, l_6, l_7\}$ be a clique consisting of 8 literals. Let x_0, x_1, x_2 be new Boolean variables.

$$
\begin{aligned}
l_0 &\rightarrow (\neg x_0 \wedge \neg x_1 \wedge \neg x_2) \\
l_1 &\rightarrow (\neg x_0 \wedge \neg x_1 \wedge x_2) \\
l_2 &\rightarrow (\neg x_0 \wedge x_1 \wedge \neg x_2) \\
l_3 &\rightarrow (\neg x_0 \wedge x_1 \wedge x_2) \\
l_4 &\rightarrow (x_0 \wedge \neg x_1 \wedge \neg x_2) \\
l_5 &\rightarrow (x_0 \wedge \neg x_1 \wedge x_2) \\
l_6 &\rightarrow (x_0 \wedge x_1 \wedge \neg x_2) \\
l_7 &\rightarrow (x_0 \wedge x_1 \wedge x_2)
\end{aligned}
$$

In general, for n literals there are $n\lceil \log_2 n \rceil$ 2-literal clauses.

Complexity of finding cliques

- Finding a maximum cardinality clique is NP-hard.
- Approximation to any constant factor is NP-hard.
- Of course, polynomial-time algorithms for finding cliques exist but they have no approximation guarantees.
- (Bicliques do have polynomial-time 2-approximation algorithms!)

General compression procedure

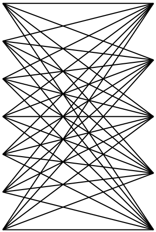
1. Find a big clique in the constraint graph.
2. If only small cliques were found, go to the last step.
3. Represent the clique compactly.
4. Remove the edges of the clique from the constraint graph.
5. Continue from step 1.
6. Represent the remaining edges explicitly as 2-literal clauses.

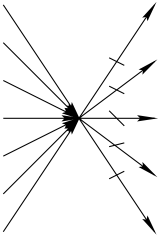
Bicliques

Definition

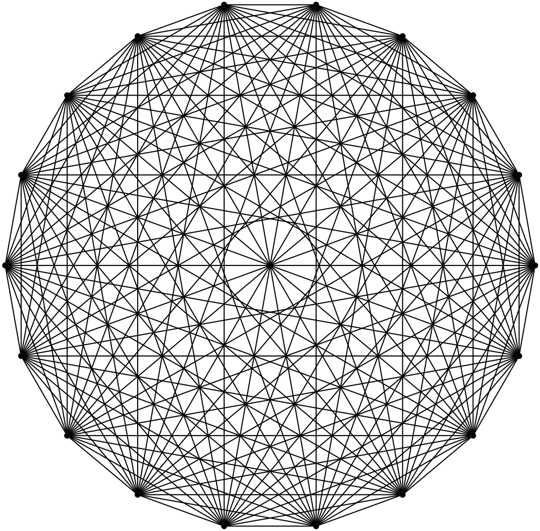
Let $\langle N, E\rangle$ be an undirected graph. A *biclique* is a pair of $C \subseteq N$ and $C' \subseteq N$ such that $C \cap C' = \emptyset$ and $\{\{n_1, n_2\} | n_1 \in C, n_2 \in C'\} \subseteq E$.

The nm edges of an n, m biclique can be represented with only one auxiliary variable and $n + m$ edges.

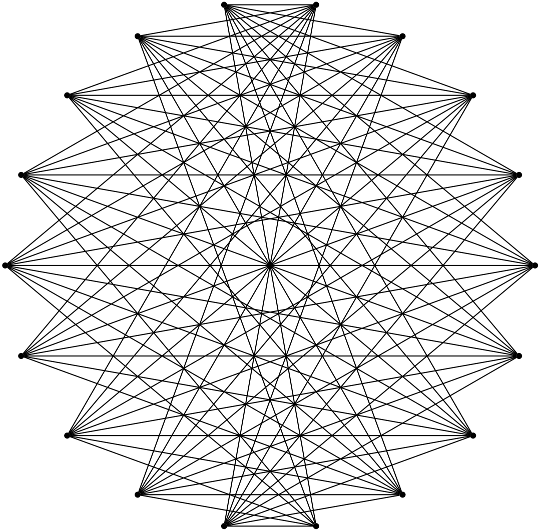




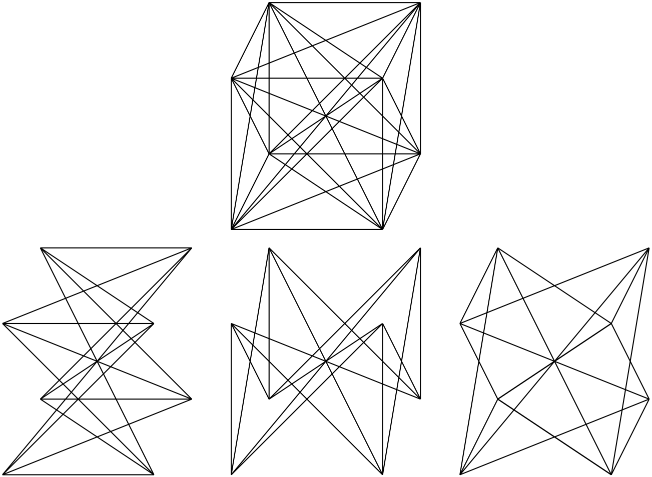
Every clique is also a biclique



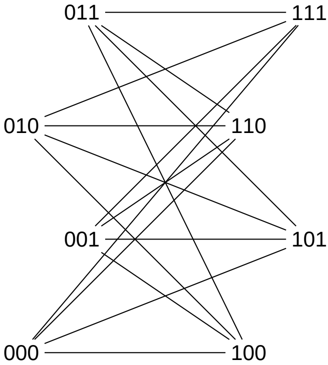
Every clique is also a biclique



Example: one 8-clique as three 4,4-bicliques



Example: one 8-clique as three 4,4-bicliques

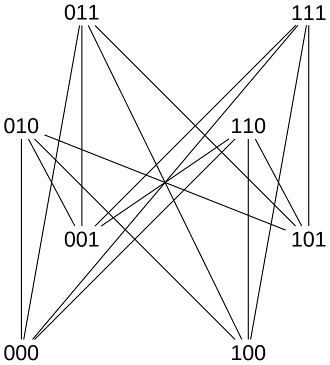


$$000 \rightarrow x_0, x_0 \rightarrow \neg 100$$
$$001 \rightarrow x_0, x_0 \rightarrow \neg 101$$
$$010 \rightarrow x_0, x_0 \rightarrow \neg 110$$
$$011 \rightarrow x_0, x_0 \rightarrow \neg 111$$

$$000 \rightarrow x_1, x_1 \rightarrow \neg 010$$
$$001 \rightarrow x_1, x_1 \rightarrow \neg 011$$
$$100 \rightarrow x_1, x_1 \rightarrow \neg 110$$
$$101 \rightarrow x_1, x_1 \rightarrow \neg 111$$

$$000 \rightarrow x_2, x_2 \rightarrow \neg 001$$
$$010 \rightarrow x_2, x_2 \rightarrow \neg 011$$
$$100 \rightarrow x_2, x_2 \rightarrow \neg 101$$
$$110 \rightarrow x_2, x_2 \rightarrow \neg 111$$

Example: one 8-clique as three 4,4-bicliques

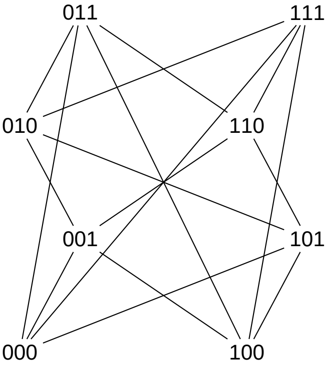


$000 \rightarrow x_0, x_0 \rightarrow \neg 100$
$001 \rightarrow x_0, x_0 \rightarrow \neg 101$
$010 \rightarrow x_0, x_0 \rightarrow \neg 110$
$011 \rightarrow x_0, x_0 \rightarrow \neg 111$

$000 \rightarrow x_1, x_1 \rightarrow \neg 010$
$001 \rightarrow x_1, x_1 \rightarrow \neg 011$
$100 \rightarrow x_1, x_1 \rightarrow \neg 110$
$101 \rightarrow x_1, x_1 \rightarrow \neg 111$

$000 \rightarrow x_2, x_2 \rightarrow \neg 001$
$010 \rightarrow x_2, x_2 \rightarrow \neg 011$
$100 \rightarrow x_2, x_2 \rightarrow \neg 101$
$110 \rightarrow x_2, x_2 \rightarrow \neg 111$

Example: one 8-clique as three 4,4-bicliques

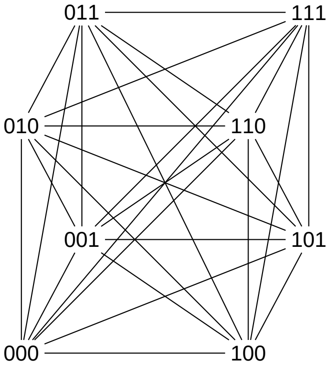


$000 \to x_0, x_0 \to \neg 100$
$001 \to x_0, x_0 \to \neg 101$
$010 \to x_0, x_0 \to \neg 110$
$011 \to x_0, x_0 \to \neg 111$

$000 \to x_1, x_1 \to \neg 010$
$001 \to x_1, x_1 \to \neg 011$
$100 \to x_1, x_1 \to \neg 110$
$101 \to x_1, x_1 \to \neg 111$

$000 \to x_2, x_2 \to \neg 001$
$010 \to x_2, x_2 \to \neg 011$
$100 \to x_2, x_2 \to \neg 101$
$110 \to x_2, x_2 \to \neg 111$

Example: one 8-clique as three 4,4-bicliques



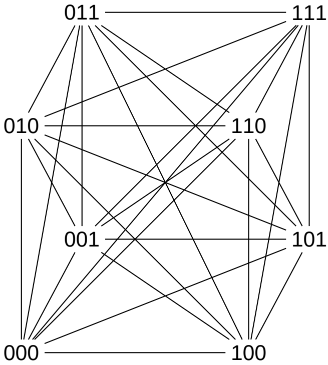
$$000 \rightarrow x_0, x_0 \rightarrow \neg 100$$
$$001 \rightarrow x_0, x_0 \rightarrow \neg 101$$
$$010 \rightarrow x_0, x_0 \rightarrow \neg 110$$
$$011 \rightarrow x_0, x_0 \rightarrow \neg 111$$

$$000 \rightarrow x_1, x_1 \rightarrow \neg 010$$
$$001 \rightarrow x_1, x_1 \rightarrow \neg 011$$
$$100 \rightarrow x_1, x_1 \rightarrow \neg 110$$
$$101 \rightarrow x_1, x_1 \rightarrow \neg 111$$

$$000 \rightarrow x_2, x_2 \rightarrow \neg 001$$
$$010 \rightarrow x_2, x_2 \rightarrow \neg 011$$
$$100 \rightarrow x_2, x_2 \rightarrow \neg 101$$
$$110 \rightarrow x_2, x_2 \rightarrow \neg 111$$

Example: one 8-clique as three 4,4-bicliques

It's equivalent to the $n \log_2 n$ encoding of cliques!



$000 \rightarrow x_0, 100 \rightarrow \neg x_0$
$001 \rightarrow x_0, 101 \rightarrow \neg x_0$
$010 \rightarrow x_0, 110 \rightarrow \neg x_0$
$011 \rightarrow x_0, 111 \rightarrow \neg x_0$

$000 \rightarrow x_1, 010 \rightarrow \neg x_1$
$001 \rightarrow x_1, 011 \rightarrow \neg x_1$
$100 \rightarrow x_1, 110 \rightarrow \neg x_1$
$101 \rightarrow x_1, 111 \rightarrow \neg x_1$

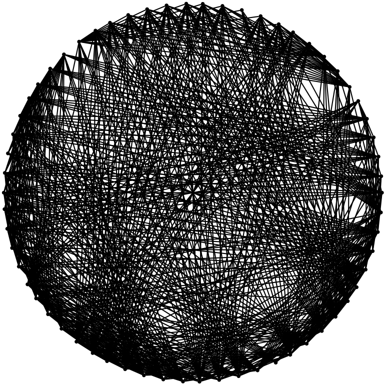
$000 \rightarrow x_2, 001 \rightarrow \neg x_2$
$010 \rightarrow x_2, 011 \rightarrow \neg x_2$
$100 \rightarrow x_2, 101 \rightarrow \neg x_2$
$110 \rightarrow x_2, 111 \rightarrow \neg x_2$

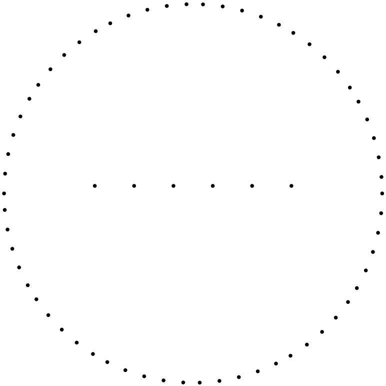
Example: IPC Airport Problem

- Problem represents the movement of airplanes at an airport.
- Constraints on the airplane movement
- Halfway the instance series the formula sizes exceed 1 GB. Culprit: binary invariants/mutexes
- All problems this far solvable in seconds: it's the physical size, not the actual difficulty.

Example: Compression of the Constraint Graph

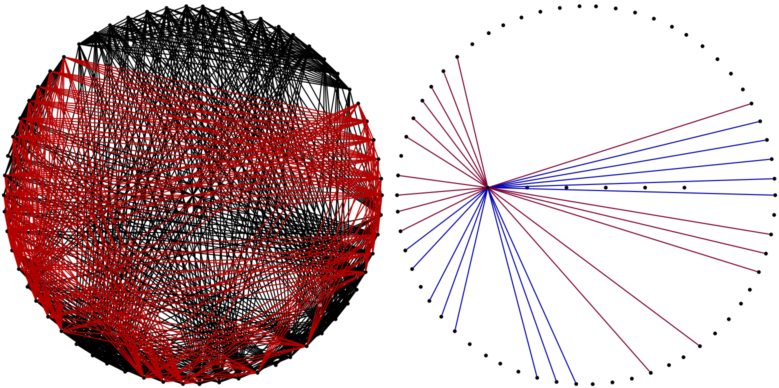
Constraint graph with 62 nodes and 653 edges





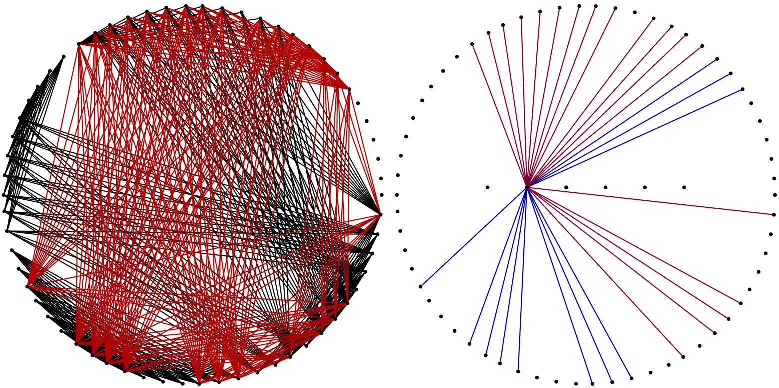
Example: Compression of the Constraint Graph

Replacing $13 \times 16 = 208$ by $13 + 16 = 29$ edges.



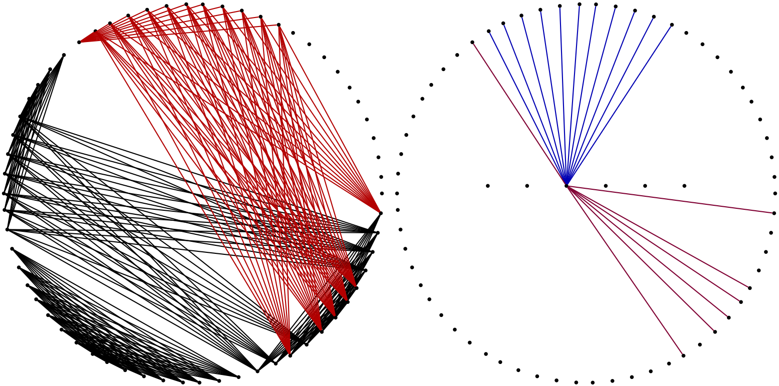
Example: Compression of the Constraint Graph

Replacing $11 \times 18 = 198$ by $11 + 18 = 29$ edges.



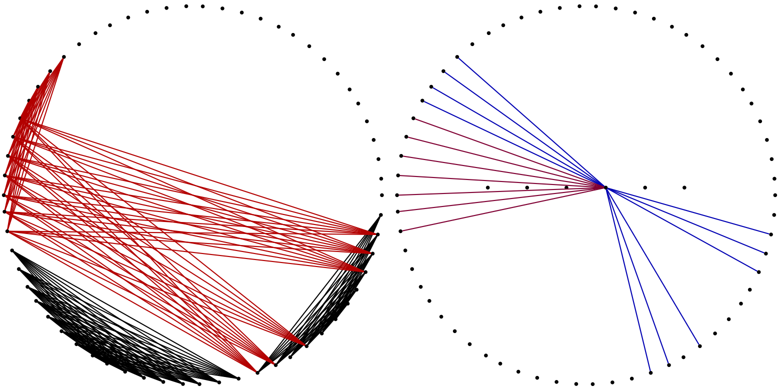
Example: Compression of the Constraint Graph

Replacing $11 \times 7 = 77$ by $11 + 7 = 18$ edges.



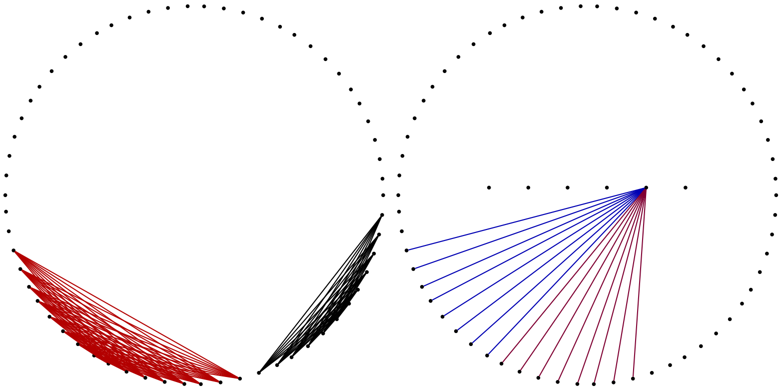
Example: Compression of the Constraint Graph

Replacing $10 \times 7 = 70$ by $10 + 7 = 17$ edges.



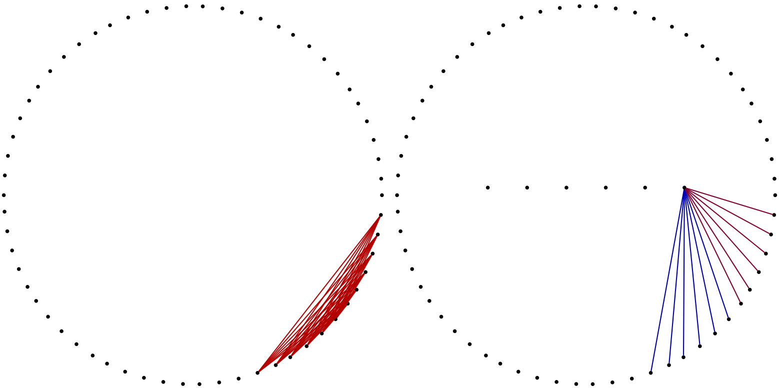
Example: Compression of the Constraint Graph

Replacing $8 \times 8 = 64$ by $8 + 8 = 16$ edges.



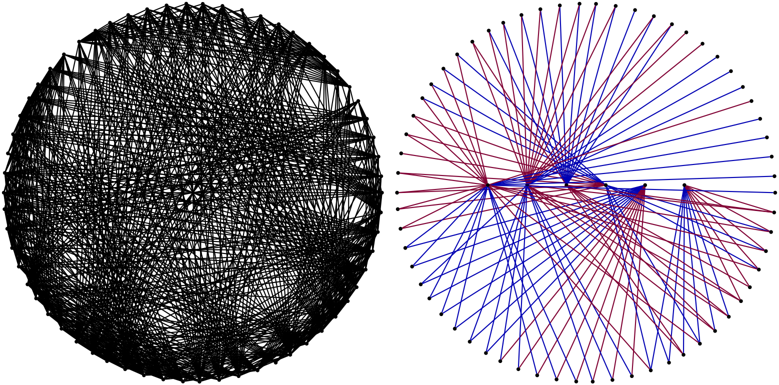
Example: Compression of the Constraint Graph

Replacing $6 \times 6 = 36$ by $6 + 6 = 12$ edges.



Example: Compression of the Constraint Graph

Total reduction is from 653 to 121 edges.



Example: IPC Airport Problem

	clauses for invariants		size in MB	
instance	before	after	before	after
21_4halfMUC_P2	182094	13191	2.59	0.37
22_4halfMUC_P3	275927	21388	4.06	0.58
23_4halfMUC_P4	381675	31776	5.60	0.84
24_4halfMUC_P4	383791	30407	5.72	0.90
25_4halfMUC_P5	478455	41719	7.24	1.18
26_4halfMUC_P6	587951	50247	8.85	1.43
27_4halfMUC_P6	572292	53721	9.01	1.57
28_4halfMUC_P7	670530	66060	10.62	1.89
36_5MUC_P2	325136	18872	4.68	0.52
37_5MUC_P3	490971	30681	7.40	0.93
38_5MUC_P3	487600	29464	7.30	0.86
39_5MUC_P4	655616	44647	10.08	1.34
40_5MUC_P4	657309	43872	10.04	1.27
41_5MUC_P4	653940	42314	9.93	1.20

Other domains and applications

- The size reduction for many other problems is far less dramatic: 10, 30, 50 per cent.
- Action mutexes / interference constraints:
  - Trivial $\mathcal{O}(n^2)$ representation (used in BLACKBOX, SatPlan, ...) catastrophic for big problems.
  - We have given (Rintanen et al. 2005, 2007) linear encodings: very good scalability in comparison to BLACKBOX/SatPlan.
  - Surprisingly, the biclique reduction is often better than the linear encoding, but in few cases far worse.

Other domains and applications

- The size reduction for many other problems is far less dramatic: 10, 30, 50 per cent.
- Action mutexes / interference constraints:
  - Trivial $\mathcal{O}(n^2)$ representation (used in BLACKBOX, SatPlan, ...) catastrophic for big problems.
  - We have given (Rintanen et al. 2005, 2007) linear encodings: very good scalability in comparison to BLACKBOX/SatPlan.
  - Surprisingly, the biclique reduction is often better than the linear encoding, but in few cases far worse.

Conclusions

- We presented a biclique based technique for representing sets of 2-literal clauses more compactly (sometimes much more).
- The basic idea is very simple and widely applicable.
- Quadratic worst-case cannot be eliminated (there is a simple argument showing this.)
- We have shown how compression with cliques is a special case of compression with bicliques.
- Challenges: more efficient algorithms for finding big cliques and bicliques