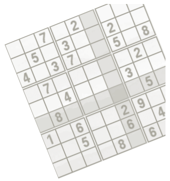


Constraint Programming

Practical Exercises

Roman Barták
Department of Theoretical Computer Science and Mathematical Logic

Constraint Modelling

Homework

- Petra needs to move her flat. She has three friends that can help her, but only 60 minutes to do the job. The following table shows how many people are necessary to move a given item and how much time it takes.

item	time (min.)	people
piano	30	3
chair	10	1
bed	15	3
table	15	2

- When should we move each item?

Modelling – house move

- We need to find the start time for moving each item in such a way that at any time, the capacity of four people is not exceeded.

```
:-use_module(library(clpfd)).

furniture(Furniture):-
  Furniture = [P, Ch, B, T],
  domain(Furniture,0,60),
  P+30 #=< 60,     % piano
  Ch+10 #=< 60,    % chair
  B+15 #=< 60,     % bed
  T+15 #=< 60,     % table
  P+30 #=< B #\/ B+15 #=< P,
  P+30 #=< T #\/ T+15 #=< P,
  B+15 #=< T #\/ B+15 #=< T,

  labeling([],Furniture).
```

item	time	people
piano	30	3
chair	10	1
bed	15	3
table	15	2

Piano and bed must be moved at different times due to capacity limit

Using global constraints

- Disjunctions (usually) do not propagate well. We can use a global constraint **cumulative** to describe allocation of activities to resources.

```
furnitureG(Furniture):-
  Furniture = [P, Ch, B, T],
  domain(Furniture,0,60),
  P+30 #= EP, EP #=< 60,      % piano
  Ch+10 #= ECh, ECh #=< 60,   % chair
  B+15 #= EB, EB #=< 60,      % bed
  T+15 #= ET, ET #=< 60,      % table
  cumulative([task(P,30,EP,3,1),
      task(Ch,10,ECh,1,2),
      task(B,15,EB,3,3),
      task(T,15,ET,2,4)], [limit(4)]),

  labeling([],Furniture).
```

The number of choice points

furniture	18
furnitureG	3

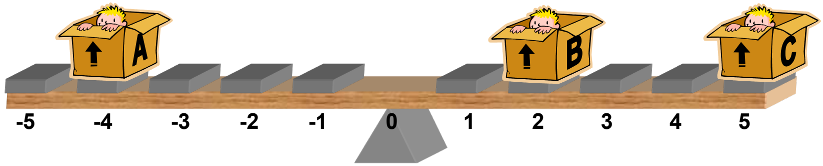
labeling([assumptions(K)],Vars).

task(start,duration,end,capacity,id)

Seesaw problem

The problem:

Adam (36 kg), Boris (32 kg) and Cecil (16 kg) want to sit on a seesaw with the length 10 foots such that the minimal distances between them are more than 2 foots and the seesaw is balanced.



A CSP model:

- A,B,C in -5..5 — position
- 36*A+32*B+16*C = 0 — equilibrium state
- |A-B|>2, |A-C|>2, |B-C|>2 — minimal distances

Seesaw problem - implementation

```
:-use_module(library(clpfd)).

seesaw(Sol):-
   Sol = [A,B,C],

   domain(Sol,-5,5),
   36*A+32*B+16*C #= 0,
   abs(A-B)#>2, abs(A-C)#>2, abs(B-C)#>2,

   labeling([ff],Sol).
```

```
?- seesaw(X).

X = [-4,2,5] ? ;
X = [-4,4,1] ? ;
X = [-4,5,-1] ? ;
X = [4,-5,1] ? ;
X = [4,-4,-1] ? ;
X = [4,-2,-5] ? ;

no
```

Symmetry breaking

– important to reduce search space

```
:-use_module(library(clpfd)).

seesaw(Sol):-
   Sol = [A,B,C],

   domain(Sol,-5,5),
   A #=< 0,
   36*A+32*B+16*C #= 0,
   abs(A-B)#>2, abs(A-C)#>2, abs(B-C)#>2,

   labeling([ff],Sol).
```

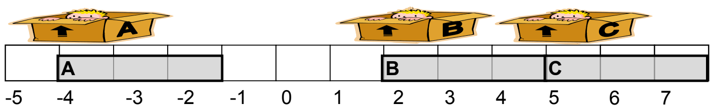
```
?- seesaw(X).

X = [-4,2,5] ? ;
X = [-4,4,1] ? ;
X = [-4,5,-1] ? ;

no
```

Seesaw problem - a different perspective

```
domain([A,B,C],-5,5),
A #=< 0,
36*A+32*B+16*C #= 0,
abs(A-B)#>2,
abs(A-C)#>2,
abs(B-C)#>2
```

```
A in -4..0
B in -1..5
C in -5..5
```

- A set of similar constraints typically indicates a structured sub-problem that can be represented using a **global constraint.**



- We can use a global constraint describing **allocation of activities to exclusive resource**.

```
domain([A,B,C],-5,5),
A #=< 0,
36*A+32*B+16*C #= 0,
cumulative([task(A,3,_,1,1),task(B,3,_,1,2),
           task(C,3,_,1,3)],[limit(1)]),
```

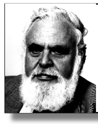
task(start,duration,end,capacity,id)

```
A in -4..0
B in -1..5
C in (-5..-3) \/ (-1..5)
```

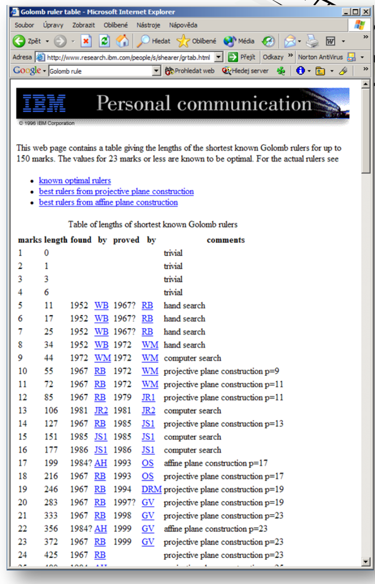
Golomb ruler

- A **ruler with M marks** such that **distances** between any two marks are **different**.
- The **shortest ruler** is the optimal ruler.

- **Hard** for M≥16, no exact algorithm for M ≥ 24!
- Applied in **radioastronomy**.



Solomon W. Golomb
Professor
University of Southern California
http://csi.usc.edu/faculty/golomb.html



This web page contains a table giving the lengths of the shortest known Golomb rulers for up to 150 marks. The values for 23 marks or less are known to be optimal. For the actual rulers see

- known optimal rulers
- best rulers from projective plane construction
- best rulers from affine plane construction

Table of lengths of shortest known Golomb rulers

marks	length	found	by	proved	by	comments
1	0					trivial
2	1					trivial
3	3					trivial
4	6					trivial
5	11	1952	WB	1967?	RB	hand search
6	17	1952	WB	1967?	RB	hand search
7	25	1952	WB	1967?	RB	hand search
8	34	1952	WB	1972	WM	hand search
9	44	1972	WM	1972	WM	computer search
10	55	1967	RB	1972	WM	projective plane construction p=9
11	72	1967	RB	1972	WM	projective plane construction p=11
12	85	1967	RB	1979	JR1	projective plane construction p=11
13	106	1981	JR2	1981	JR2	computer search
14	127	1967	RB	1985	JS1	projective plane construction p=13
15	151	1985	JS1	1985	JS1	computer search
16	177	1986	JS1	1986	JS1	computer search
17	199	1984?	AH	1993	OS	affine plane construction p=17
18	216	1967	RB	1993	OS	projective plane construction p=17
19	246	1967	RB	1994	DRM	projective plane construction p=19
20	283	1967	RB	1997?	GV	projective plane construction p=19
21	333	1967	RB	1998	GV	projective plane construction p=23
22	356	1984?	AH	1999	GV	affine plane construction p=23
23	372	1967	RB	1999	GV	projective plane construction p=23
24	425	1967	RB			projective plane construction p=23

Golomb ruler – a model

A base model:

Variables $X_1, \ldots, X_M$ with the domain 0..M*M

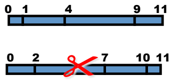
$X_1 = 0$ *ruler start*

$X_1 < X_2 < \ldots < X_M$ *no permutations of variables*

$\forall i<j\ D_{i,j} = X_j - X_i$ *difference variables*

all_different($\{D_{1,2}, D_{1,3}, \ldots D_{1,M}, D_{2,3}, \ldots D_{M,M-1}\}$)

Model extensions:



$D_{1,2} < D_{M-1,M}$ *symmetry breaking*

better bounds (**implied constraints**) for $D_{i,j}$

$D_{i,j} = D_{i,i+1} + D_{i+1,i+2} + \ldots + D_{j-1,j}$

so **$D_{i,j} \geq \Sigma_{j-i} = (j-i)*(j-i+1)/2$** *lower bound*

$X_M = X_M - X_1 = D_{1,M} = D_{1,2} + D_{2,3} + \ldots D_{i-1,i} + D_{i,j} + D_{j,j+1} + \ldots + D_{M-1,M}$

$D_{i,j} = X_M - (D_{1,2} + \ldots D_{i-1,i} + D_{j,j+1} + \ldots + D_{M-1,M})$

so **$D_{i,j} \leq X_M - (M-1-j+i)*(M-j+i)/2$** *upper bound*

Golomb ruler - some results

- **What is the effect of different constraint models?**

size	base model	base model + symmetry	base model + symmetry + implied constraints
7	220	80	30
8	1 462	611	190
9	13 690	5 438	1 001
10	120 363	49 971	7 011
11	2 480 216	985 237	170 495

time in milliseconds on Mobile Pentium 4-M 1.70 GHz, 768 MB RAM

- **What is the effect of different search strategies?**

size	fail first			leftmost first		
	enum	*step*	*bisect*	*enum*	*step*	*bisect*
7	40	60	40	30	30	30
8	390	370	350	220	190	200
9	2 664	2 384	2 113	1 182	1 001	921
10	20 870	17 545	14 982	8 782	7 011	6 430
11	1 004 515	906 323	779 851	209 251	170 495	159 559

time in milliseconds on Mobile Pentium 4-M 1.70 GHz, 768 MB RAM

Toaster

- Propose a constraint model describing the toaster problem – the task is to minimize the number of toaster usages. Two toasts can be toasted together, but only one side of each toast is baked. The input to the model is the number of toasts.

Homework

Department of Theoretical Computer Science and Mathematical Logic
bartak@ktiml.mff.cuni.cz