Knowledge Representation and Reasoning

Exercises on Advanced ASP

1 Cardinality Rules

Consider the following cardinality constraint in the head of a rule: $1\{a, b, c\}2$.

a) Compile the cardinality constraint into cardinality rules of the form

$$a_0 \leftarrow l\{a_1, \dots, a_m, \sim a_{m+1}, \dots, \sim a_n\}$$

along with normal and choice rules as well as integrity constraints.

b) Compile the logic program P resulting from the previous subtask into a program P' with normal and choice rules as well as integrity constraints only, using the $cc(i, j)$ construction from the lecture slides.

c) Determine the stable models of P and the corresponding stable models of P'.

Answer: a) P:

$$\begin{gathered} \{a, b, c\} \\ x \leftarrow 1\{a, b, c\} \\ y \leftarrow 3\{a, b, c\} \\ z \leftarrow x, \sim y \\ \leftarrow \sim z \end{gathered}$$

b) P':

$$\begin{array}{cccc}
\{a, b, c\} & cc(1,1) \leftarrow cc(2,0), a & cc(2,1) \leftarrow cc(3,0), b & cc(3,1) \leftarrow cc(4,0), c \\
x \leftarrow cc(1,1) & cc(1,0) \leftarrow cc(2,0) & cc(2,0) \leftarrow cc(3,0) & cc(3,0) \leftarrow cc(4,0) \\
y \leftarrow cc(1,3) & cc(1,2) \leftarrow cc(2,1), a & cc(2,2) \leftarrow cc(3,1), b & [cc(3,2) \leftarrow cc(4,1), c] \\
z \leftarrow x, \sim y & cc(1,1) \leftarrow cc(2,1) & cc(2,1) \leftarrow cc(3,1) & [cc(3,1) \leftarrow cc(4,1)] \\
\leftarrow \sim z & cc(1,3) \leftarrow cc(2,2), a & [cc(2,3) \leftarrow cc(3,2), b] & [cc(3,3) \leftarrow cc(4,2), c] \\
 & cc(1,2) \leftarrow cc(2,2) & [cc(2,2) \leftarrow cc(3,2)] & [cc(3,2) \leftarrow cc(4,2)] \\
 & [cc(1,4) \leftarrow cc(2,3), a] & [cc(2,4) \leftarrow cc(3,3), b] & [cc(3,4) \leftarrow cc(4,3), c] \\
cc(4,0) & [cc(1,3) \leftarrow cc(2,3)] & [cc(2,3) \leftarrow cc(3,3)] & [cc(3,3) \leftarrow cc(4,3)]
\end{array}$$

c)

P	P'
$\{a, x, z\}$	$\{a, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1)\}$
$\{b, x, z\}$	$\{b, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1), cc(2,1)\}$
$\{c, x, z\}$	$\{c, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1), cc(2,1), cc(3,1)\}$
$\{a, b, x, z\}$	$\{a, b, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1), cc(2,1), cc(1,2)\}$
$\{a, c, x, z\}$	$\{a, c, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1), cc(2,1), cc(3,1), cc(1,2)\}$
$\{b, c, x, z\}$	$\{b, c, x, z, cc(1,0), cc(2,0), cc(3,0), cc(4,0), cc(1,1), cc(2,1), cc(3,1), cc(1,2), cc(2,2)\}$

2 Tournament (Part 2)

Recall the problem of the tournament. The provided solution includes two cardinality constraints in the generator.

```
3{in(X,Y):team(X)}3 :- group(Y).
1{in(X,Y):group(Y)}1 :- team(X).
```

a) Compile each cardinality constraint (omitting the rule body) into cardinality rules. Note that you have to instantiate first. For simplicity, you may assume that there is only group 1, and teams a, b, and c, and, for the solution, it suffices to compile the cases of group 1 and team a.

b) Compile the logic program resulting from the previous task into a program P' with normal and choice rules as well as integrity constraints only, using the $cc(i,j)$ construction from the lecture slides (simplify when possible).

c) Create an alternative ASP encoding for the overall problem of the group assignment (as spelled out on exercise sheet P6) using only normal rules (and constraints) and compare the results.

Answer:

a) First, we need to ground w.r.t. group 1 and team a:

$3\{in(a,1), in(b,1), in(c,1)\}3 \leftarrow group(1)$

$1\{in(a,1)\}1 \leftarrow team(a)$

Then, we compile:

$$\begin{array}{rl} & \{in(a,1), in(b,1), in(c,1)\} \\ x_1 & \leftarrow 3\{in(a,1), in(b,1), in(c,1)\} \\ y_1 & \leftarrow 4\{in(a,1), in(b,1), in(c,1)\} \\ z_1 & \leftarrow x_1, \sim y_1 \\ & \leftarrow \sim z_1 \end{array} \qquad \begin{array}{rl} & \{in(a,1)\} \\ x_2 & \leftarrow 1\{in(a,1)\} \\ y_2 & \leftarrow 2\{in(a,1)\} \\ z_2 & \leftarrow x_2, \sim y_2 \\ & \leftarrow \sim z_2 \end{array}$$

b)

$$\begin{array}{rl} & \{in(a,1), in(b,1), in(c,1)\} \\ x_1 & \leftarrow cc(1,3) \\ y_1 & \leftarrow cc(1,4) \\ z_1 & \leftarrow x_1, \sim y_1 \\ & \leftarrow \sim z_1 \end{array} \quad \begin{array}{rl} & \{in(a,1)\} \\ x_2 & \leftarrow cc1(1,1) \\ y_2 & \leftarrow cc1(1,2) \\ z_2 & \leftarrow x_2, \sim y_2 \\ & \leftarrow \sim z_2 \end{array} \quad \begin{array}{rl} cc(3,1) & \leftarrow in(c,1) \\ cc(2,2) & \leftarrow cc(3,1), in(b,1) \\ cc(2,1) & \leftarrow cc(3,1) \\ cc(2,1) & \leftarrow in(b,1) \\ cc(1,3) & \leftarrow cc(2,2), in(a,1) \end{array} \quad \begin{array}{rl} cc1(1,2) & \leftarrow cc1(2,2) \\ cc1(1,2) & \leftarrow cc1(2,1), in(a,1) \\ cc1(1,1) & \leftarrow cc1(2,1) \\ cc1(1,1) & \leftarrow in(a,1) \end{array}$$

c)

```
group(1..2).
team(a;b;c;d;e;f).
in(X,Y):- not out(X,Y), team(X),group(Y).
out(X,Y):- not in(X,Y), team(X),group(Y).
g(X):- in(X,Y).
:- team(X),not g(X).
:- in(X,Y),in(X,Z),Y!=Z.
t(X):- in(X1,X),in(X2,X),in(X3,X),X1!=X2,X2!=X3,X1!=X3.
:- group(X), not t(X).
:- in(X1,X),in(X2,X),in(X3,X),in(X4,X),X1!=X2,X2!=X3,X1!=X3,X1!=X4,X2!=X4,X3!=X4.
:- in (a,1).
:- in(a,X), in(e,X).
:- in(e,X), in(f,X).
aux:- in(b,X),in(c,X).
:- not aux.
#show in/2.
```

3 Weight Rules

Consider the following weight constraint in the head of a rule: $4\{1 : b_1, 1 : b_2, 2 : c_1, 2 : c_2\}5$.

a) Compile the weight constraint into weight rules of the form

$$a_0 \leftarrow l\{w_1 : a_1, \ldots, w_m : a_m, w_{m+1} :\sim a_{m+1}, \ldots, w_n :\sim a_n\}$$

along with normal rules and integrity constraints.

b) Generalize (and simplify) the scheme used for cardinality constraints before, and compile the logic program P resulting from the previous subtask into a program P' with normal and choice rules as well as integrity constraints only.

Answer:
a) P:

$$\begin{aligned}
&\{b_1, b_2, c_1, c_2\}\\
x &\leftarrow 4\{1 : b_1, 1 : b_2, 2 : c_1, 2 : c_2\}\\
y &\leftarrow 6\{1 : b_1, 1 : b_2, 2 : c_1, 2 : c_2\}\\
z &\leftarrow x, \sim y\\
&\leftarrow \sim z
\end{aligned}$$

b)

$$\begin{array}{llll}
\{b_1, b_2, c_1, c_2\} & cc(4,2) \leftarrow c_2 & cc(2,5) \leftarrow cc(3,4), b_2 & cc(1,6) \leftarrow cc(2,5), b_1\\
x \leftarrow cc(1,4) & cc(3,2) \leftarrow c_1 & cc(2,4) \leftarrow cc(3,4) & cc(1,5) \leftarrow cc(2,5)\\
y \leftarrow cc(1,6) & cc(3,4) \leftarrow cc(4,2), c_1 & cc(2,3) \leftarrow cc(3,2), b_2 & cc(1,5) \leftarrow cc(2,4), b_1\\
z \leftarrow x, \sim y & cc(3,2) \leftarrow cc(4,2) & & cc(1,4) \leftarrow cc(2,4)\\
\leftarrow \sim z & & & cc(1,4) \leftarrow cc(2,3), b_1
\end{array}$$

Note that this includes simplifications omitting rules that cannot contribute to a counter above 3.

4 Extended Programs

Find the stable models of the following extended programs:

$$\begin{array}{llll}
a) P = \{ & 1\{p, q\} \leftarrow & 1\{r, s\}1 \leftarrow \{p, q\}1\} & \\
b) P = \{ & 1\{p, q, r\}2 \leftarrow & 2\{p, q, s\}2 \leftarrow 1\{q, r, s\}2\} & \\
c) P = \{ & 2\{p, q, r\} \leftarrow & \{p, q\}1 \leftarrow s & s \leftarrow q, r\}\\
d) P = \{ & p \leftarrow 2\{q, r, s\} & 1\{q, r, s\}2 \leftarrow \sim p & 2\{r, s\} \leftarrow \sim q\}\\
e) P = \{ & p \leftarrow 2\{q, r, s\} & 2\{p, q, r\} \leftarrow \sim s & 2\{r, s\} \leftarrow p\}
\end{array}$$

Answer:
a) $\{p, r\}$, $\{p, s\}$, $\{q, r\}$, $\{q, s\}$, and $\{p, q\}$
b) $\{p\}$, $\{p, q\}$, $\{p, s\}$, $\{q, s\}$, $\{p, r, s\}$, and $\{q, r, s\}$
c) $\{p, q\}$, $\{p, r\}$, and $\{q, r, s\}$
d) $\{q\}$ and $\{p, r, s\}$
e) none

5 Programs with Aggregates

Determine the stable models of the following logic programs P with aggregates, check whether the contained aggregates are monotone, anti-monotone, or non-monotone, and provide appropriate translations of the aggregates to propositional formulas.

a) $\mathrm{P} = \begin{Bmatrix} p \leftarrow sum\{1 : p, 1 : q\} \neq 1 \\ p \leftarrow q \\ q \leftarrow p \end{Bmatrix}$

b) $\mathrm{P} = \begin{Bmatrix} p \leftarrow sum\{1 : p, 1 : q\} < 1 \\ p \leftarrow sum\{1 : p, 1 : q\} > 1 \\ p \leftarrow q \\ q \leftarrow p \end{Bmatrix}$

c) $\mathrm{P} = \begin{Bmatrix} \{p\} \\ \{q\} \\ s \leftarrow sum\{1 : p, 1 : q, 2 : s\} \neq 3 \end{Bmatrix}$

d) $\mathrm{P} = \begin{Bmatrix} \{p\} \\ \{q\} \\ s \leftarrow sum\{1 : p, 1 : q, 2 : s\} < 3 \\ s \leftarrow sum\{1 : p, 1 : q, 2 : s\} > 3 \end{Bmatrix}$

Answer:
a) $\{p, q\}$; non-monotone; $(p \to q) \wedge (q \to p)$
b) none; antimonotone and monotone; $(\neg p \wedge \neg q)$ and $(p \wedge q)$
c) $\{s\}$ and $\{p, q, s\}$; non-monotone; $((p \wedge s) \to q) \wedge ((q \wedge s) \to p)$
d) $\{s\}$; antimonotone and monotone; $\neg s \vee (\neg p \wedge \neg q)$ and $(s \wedge p \wedge q)$