$T_P^0(\emptyset) = \emptyset$

$T_P^1(\emptyset) = \underbrace{\{border(a,l), \ldots\ldots\ border(bol,br)\}}$

$T_P^2(\emptyset) = T(T_P^1(\emptyset)) = T_P(\ \ )$

anwendbare Regeln suchen und anwenden:

$= \{ T_P^1(\emptyset) \cup \{border(d,a), \ldots\ border(br,bol)\}$ Regel 1

$\cup \{reachable(a,d)^1 \ldots reachable(bol,br)^1\}$

2. Regel anw.

$T_P^3(\emptyset) = T_P(T_P^2(\emptyset)) = \{\ldots\} \cup \{unreachable(d,a)^1, \ldots, reachable(bol,br)^1\}$ Regel 2

$\cup \{$ Regel 3 aus $r(a,d), b(l,f) \rightsquigarrow r(a,f)^2$

$unreachable(X,Y) \leftarrow \neg reachable(X,Y)$

unendlich $unreachable(1,2)$
$unreachable(alice,bob)$
→ Variablen pos. binden

↑ Regel 1 ist „nicht sicher“ (safe)

$unreachable(X,Y) \leftarrow country(X), country(Y), \neg reachable(X,Y).$

z.B. $unreachable(D,BR)$

$p \leftarrow \neg q \qquad q \leftarrow \neg p$

zyklische neg. Abhängigkeiten

mögl. Modell: $\{p, \neg q\} \qquad \{q, \neg p\}$

Resolutionsverfahren

Regel $h \leftarrow b_1, b_2, b_3$

$\rightsquigarrow$ Klauselmenge: $\{h, \neg b_1, \neg b_2, \neg b_3\}$

weitere Klausel $\{\neg h\}$

$\{\neg b_1, \neg b_2, \neg b_3\}$

$\{b_1\}$

$\{\neg b_2, \neg b_3\}$

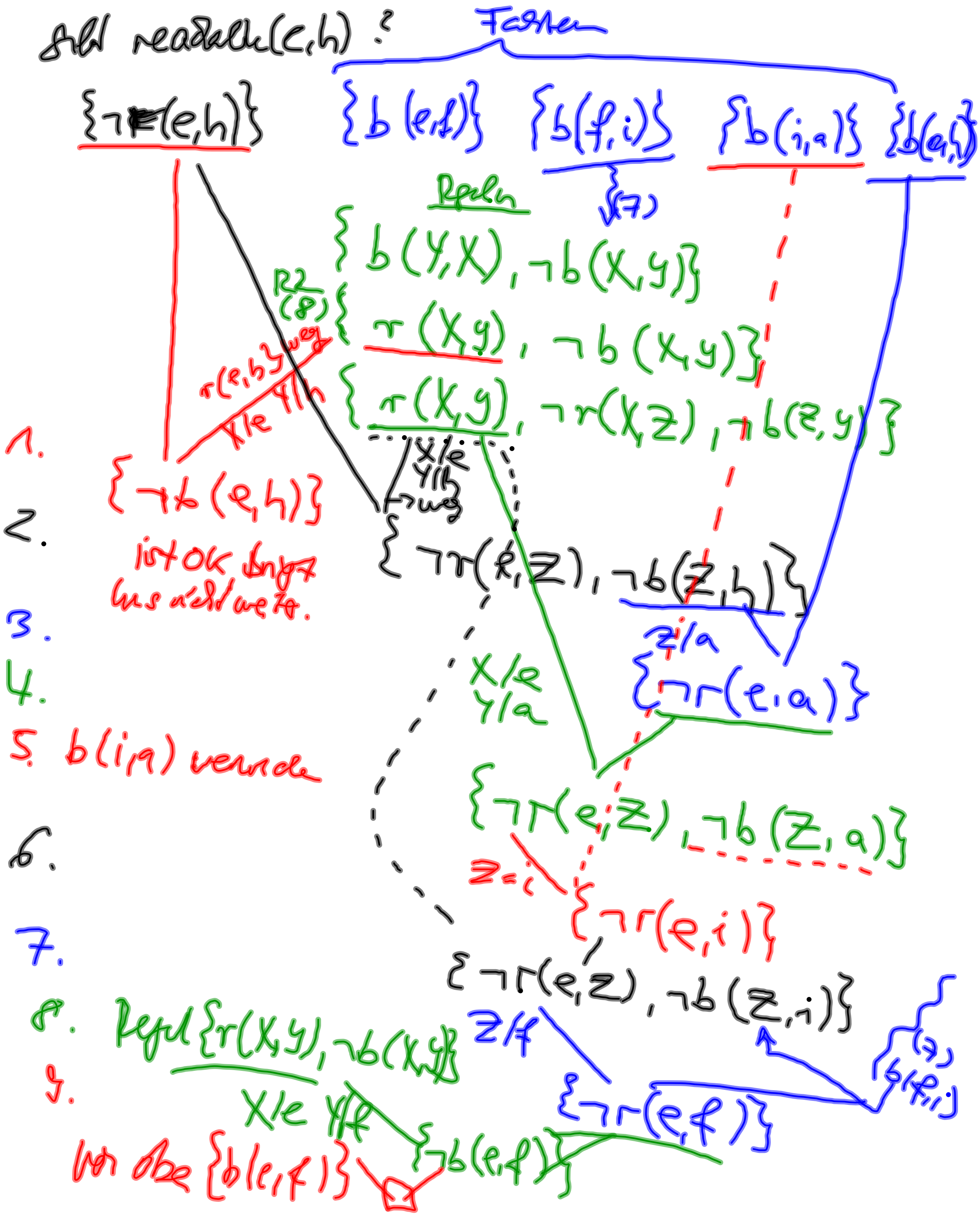
$\{b_2\}$

$\{\neg b_3\}$

$\{b_3\}$

$\Box$

Annahme „$\neg h$"
Kann [illegible]
$\Longrightarrow$ h abgeleitet


Gilt readable(e,h)?
Fakten
{¬r(e,h)}
{b(e,f)}
{b(f,i)}
{b(i,a)}
{b(a,h)}
(7)
Regeln
{b(Y,X), ¬b(X,Y)}
R2 (8)
{r(X,Y), ¬b(X,Y)}
{r(X,Y), ¬r(X,Z), ¬b(Z,Y)}
r(e,h) weg
X/e Y/h
X/e Y/h
¬ weg
1.
{¬b(e,h)}
ist OK, bringt uns nicht weiter.
2.
{¬r(e,Z), ¬b(Z,h)}
3.
Z/a
4.
{¬r(e,a)}
X/e Y/a
5. b(i,a) verwenden
{¬r(e,Z), ¬b(Z,a)}
6.
Z=i
{¬r(e,i)}
7.
{¬r(e,Z), ¬b(Z,i)}
Z/f
8. Regel {r(X,Y), ¬b(X,Y)}
{¬r(e,f)}
(7) {b(f,i)}
9.
X/e Y/f
hier aber {b(e,f)}
{¬b(e,f)}

eben: Unitresolution

$$\{\neg p\} \qquad \{p, \neg b_1, \ldots, \neg b_n\}$$

$$\{\neg b_1, \ldots, \neg b_n\}$$

geht auch anders:

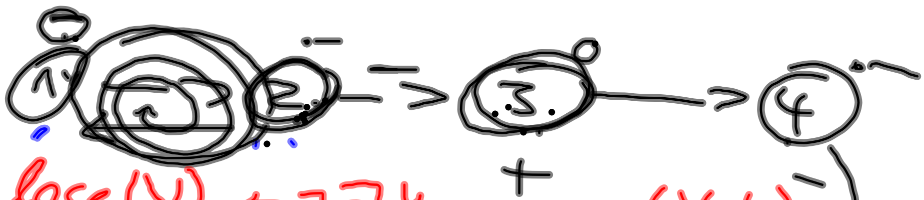
Regel 2

Regel 3

$y \to z$ $\{\underline{r(X,y)}, \neg b(X,y)\}$

$\{r(X,y), \underline{\neg r(X,Z)}, \neg r(Z,y)\}$ $\vdash 2$ $\vdash 3$

$\{r(X,y), \neg r(Z,y), \neg b(X,Z)\}$ 3



(a) $\forall x: lose(Y) \leftarrow \neg \exists y: move(X,y)$

b) $\forall x: win(X) \leftarrow \exists y: move(X,y) \wedge lose(y)$

b) win(X) :- move(X,y), lose(Y).

lose(X) :- ¬win(X)

c) ~~einen~~

$\forall x: lose(X) \leftarrow \forall y: (move(X,y) \rightarrow win(y))$

insbesondere: wenn es kein y gibt! ist diese Regel auch ok.

$\forall x: draw(X) \leftarrow not\ win(X) \wedge \neg lose(X)$.

$\forall draw(X) \leftarrow \neg \exists y: move(X,y) \wedge lose(Y)$

$\wedge \neg \forall y\, (move(X,y) \rightarrow win(y))$

in Regeln:

win(X) :- move(X,Y), lose(Y)

$\forall x\ lose(x) \leftarrow \forall y: (move(X,Y) \rightarrow win(Y))$

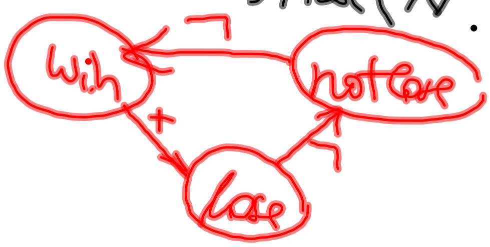
keine Regel

$notlose(X) \leftarrow move(X,Y), \neg win(Y).$

$lose(X) \leftarrow \neg notlose(X).$

Abhängigkeitsgraph:

-> neg. Zyklus



WF

lose(f). lose(k). lose(i)

w(i) w(a) w(b)

lose(e)

win(d)

lose(l)

win(c)

fehlen: m, h, g
⇒ drawn