1 Preliminaries

Definition 1 ($QFBAPA$)**.** Let T be a set of symbols

- set terms over T are:
  - empty set $\emptyset$ and universal set $\mathcal{U}$
  - every set symbol in T
  - if s, t are set terms then also $s \cap t$, $s \cup t$ and $s^\neg$
- set constraints over T are
  - $s \subseteq t$ and $s \not\subseteq t$
  - $s = t$ and $s \neq t$

  where s, t are set terms
- cardinality terms over T are:
  - every number $n \in \mathbb{N}$
  - $|s|$ if s is a set term
  - if k, l are cardinality terms then also $k + l$ and $n \cdot k$, $n \in \mathbb{N}$
- cardinality constraints over T are:
  - $k = l$ and $k \neq l$
  - $k < l$ and $k \not< l$
  - $k \leq l$ and $k \not\leq l$
  - $n \; dvd \; k$ and $n \; \neg dvd \; k$

  where k, l are cardinality terms and $n \in \mathbb{N}$

Definition 2 ($\mathcal{ALCSCC}$)**.** Concepts are:

- all concept names
- if C, D are concepts then:
  - $\neg C$
  - $C \sqcup D$
  - $C \sqcap D$
- $succ(c)$ if c is a set or cardinality constraint over $\mathcal{ALCSCC}$ concepts and role names

The set S is a set of constraints of the form $x : C$ and $(x, y) : s$, where C is a concept, s a set term and x, y variables. The constraint $(x, y) : s$ denotes that y is a successor of x, while y must hold s.
For the next definition we denote with $s \in k$ that the set term s occurs in the cardinality constraint k.

Definition 3 (Number of successors)**.** Let S be a set of constraints, x be a variable and s be a set term. Then $\#s(x)$ is the number of successors of x in S which satisfy s:

$$\#s(x) = |\{s|(x, y) : s \in S\}|$$

A constraint regarding a variable x is *violated* if

- $x : succ(k \leq n)$ and $n < \sum_{s \in k} \#s(x)$
- $x : succ(k \geq n)$ and $n > \sum_{s \in k} \#s(x)$
- $x : succ(k \leq l)$ and $\sum_{t \in l} \#t(x) < \sum_{s \in k} \#s(x)$
- $x : succ(k \geq l)$ and $\sum_{t \in l} \#t(x) > \sum_{s \in k} \#s(x)$

where $n \in \mathbb{N}$ a integer.

2 Tableau

Definition 4 (Merge)**.** *Merging* y_1 and y_2 results in one variable y: replace all occurrence of y_1 and y_2 with y.

Note that by merging two successors other constraints might become violated:

$$\begin{aligned} S = \{&x : succ(|r \cap A| = 1) \sqcap succ(|r \cap B| = 1) \sqcap succ(|r| > 1), \\ &y_1 : A, y_2 : B, x.r.y_1, x.r.y_2\} \end{aligned} \tag{1}$$

If we merge y_1 and y_2 then the constraint $x : succ(|r| > 1)$ which was satisfied becomes violated. But not only constraints regarding x might become violated after merging two successors of x:

$$\begin{aligned} S = \{&x : succ(|r| \leq 1), x.r.y_1, x.r.y_2, \\ &y_1 : succ(|s| \leq 1), y_2 : succ(|s| \leq 1), y_1.s.z_1, y_2.s.z_2\} \end{aligned} \tag{2}$$

We see that the first constraint is violated and therefore merging y_1 and y_2 would solve the problem but on the other hand the constraints regarding y_1 and y_2 become violated:

$$S = \{x : succ(|r| \leq 1), x.r.y, y : succ(|s| \leq 1), y.s.z_1, y.s.z_2\}$$

To solve this problem we merge z_1 and z_2.

Definition 5 (Clash)**.** A constraint set S contains a *clash* if

- $\{x : \perp\} \subseteq S$ or
- $\{x : A, \; x : \neg A\} \subseteq S$ or
- $\{x : succ(k \leq l)\}$ and $\#k(x) > \#l(x)$

- $\{x : succ(k \leq n)\}$ and $\#k(x) > n$
- $\{x : succ(k \geq l)\}$ and $\#k(x) < \#l(x)$
- $\{x : succ(k \geq n)\}$ and $\#k(x) < n$
- $\{x : succ(n\, dvd\, k)\}$ and $mod(\#k(x), n) \neq 0$

For the next definition we define first properties of the following notations:

- Conjunction binds stronger than disjunction: $s \cup t \cap u = s \cup (t \cap u)$
- $k \leq l$ and $k \geq l$ iff $k = l$
- $s \subseteq t$ and $s \supset t$ iff $s = t$

Definition 6 (Tableau)**.** Let S be a set of constraints.

1. $\sqcap$-rule: In S is $x : C_1 \sqcap C_2$ but not both $x : C_1$ and $x : C_2$
   $\rightarrow S := S \cup \{x : C_1, x : C_2\}$

2. $\sqcup$-rule: In S is $x : C_1 \sqcup C_2$ but neither $x : C_1$ or $x : C_2$
   $\rightarrow S := S \cup \{x : C_1\}$ or $S := S \cup \{x : C_2\}$

3. *choose*-rule: In S are $x : succ(k \leq l)$, $y : C$ or $x.r.y$ and
   C or r occur in k but $(x, y) : s \notin S$ for every set term s in k
   $\rightarrow$ choose one set term s in k and either $S := S \cup \{(x, y) : s\}$ or $S := S \cup \{(x, y) : \neg s\}$

4. *cardinality*-rule: In S is $x : succ(c)$, with $c \in \{k = l,\ k \leq l,\ k < l,\ n\, dvdl\}$, which is violated

   a) if there is a set term s in l
      $\rightarrow$ introduce new variable y and $S := S \cup \{(x, y) : x\}$

   b) if $l \in \mathbb{N}$ does not contain a set term then merge two successor $y_1 \neq y_2$ of x for which $(x, y_1) : k \in S$ and $(x, y_2) : k \in S$ if no other constraints regarding x become violated

5. *set*-rule: In S are $x : succ(c_1 \subseteq c_2)$ and $(x, y) : c_1$ but not $(x, y) : c_2$
   $\rightarrow S := S \cup \{(x, y) : c_2\}$

6. *set.term*-rule (Repeat until inapplicable): In S is $(x, y) : s$ and

   a) $s = s_1 \cap s_2$ but $\{(x, y) : s_1,\ (x, y) : s_2\} \not\subseteq S$ then
      $\rightarrow S := S \cup \{(x, y) : s_1,\ (x, y) : s_2\}$

   b) $s = s_1 \cup s_2$ and neither $\{(x, y) : s_1\} \subseteq S$ nor $S\{(x, y) : s_2\} \subset S$ then
      $\rightarrow$ either $S := S \cup \{(x, y) : s_1\}$ or $S := S \cup \{(x, y) : s_2\}$

   c) $s = r$ and $x.r.y \notin S$ then
      $\rightarrow S := S \cup \{x.r.y\}$

d) $s = C$ and $y : C \notin S$, where C is a $\mathcal{ALCSCC}$ concepts then
$\rightarrow S := S \cup \{y : C\}$

Note that:

- $4b$ is never applicable for $n\, dvd\, l$
- $n_1\, dvd\, n_2 \cdot l$ and $n_1 \neg dvd\, n_2$ then $n_1\, dvd\, l$ eventually

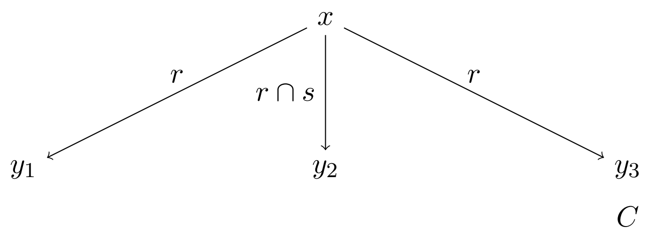
Example for $4b$:

$$S = \{x : succ(|r| = 1) \sqcap succ(|r \cap s| = 1) \sqcap succ = (|r \cap C| = 1|\} \tag{3}$$

After rule 1 (two times):

$$\begin{aligned} S = \{x : succ(|r| = 1) \sqcap succ(|r \cap s| = 1) \sqcap succ = (|r \cap C| = 1| \\ x : succ(|r| = 1), x : succ(|r \cap s| = 1), x : succ = (|r \cap C| = 1|\} \end{aligned}$$

If we try to satisfy at least two of the new constraints by the Tableau-algorithm above we end up with at least one constraint being violated. Let say we use the rules on the three new constraints sequentially. Then we have



After using rule 4b two times we have the variable x and its only $r \cap s$-successor y which is of the concept C. We could use this rule because we do not violate any other constraints. This condition helps to prevent an infinite chain of rule application:

$$S = \{x : succ(|r| < 2) \sqcap succ(|r| \geq 2)\} \tag{4}$$

First we apply the rules 4a and 6c two times to add two r-successors for x hence $x : succ(|r| < 2)$ is not satisfied any more. If we ignore the condition in rule 4b and apply it then we merge the two successors leading to $x : succ(|r| < 2)$ being satisfied but $x : succ(|r| \geq 2)$ being violated. Then we apply the rules 4a and 6c again leading to $x : succ(|r| < 2)$ being violated again and so on. By the condition in 4b we can not use rule 4b from the beginning which means the algorithm terminates with a clash stating that the constraint set is unsatisfiable.