Foundations of Constraint Programming
Tutorial 6 (on January 15th)

Sarah Gaggl

WS 2014/15

Abstract argumentation frameworks allow to represent and solve conflicting knowledge. They consist of a set of abstract arguments and a binary relation between them, denoting attacks. The inherent conflicts are solved on a semantical level by selecting sets of arguments which are *acceptable* together.

More formally, an *argumentation framework (AF)* is a pair $F = (A, R)$ where A is a set of arguments and $R \subseteq A \times A$ is the attack relation. The pair $(a, b) \in R$ means that a attacks b. We say that an argument $a \in A$ is *defended* (in F) by a set $S \subseteq A$ if, for each $b \in A$ such that $(b, a) \in R$, there exists a $c \in S$ such that $(c, b) \in R$.

An argumentation framework can be represented as a directed graph. Let $F = (A, R)$ be an AF with $A = \{a, b, c, d, e\}$ and $R = \{(a, b), (b, c), (c, b), (d, c), (d, e), (e, e)\}$. The corresponding graph representation is depicted in Fig. 1.

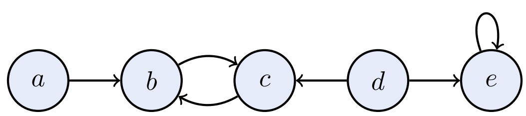

Figure 1: Example argumentation framework

Let $F = (A, R)$ be an AF. A set $S \subseteq A$ is *conflict-free (in F)*, if there are no $a, b \in S$, such that $(a, b) \in R$. $cf(F)$ denotes the collection of conflict-free sets of F. For a conflict-free set $S \in cf(F)$, it holds that

- S is a *stable extension*, i.e. $S \in stable(F)$, if each $a \notin S$ is attacked by S;
- S is an *admissible set*, i.e. $S \in adm(F)$, if each $a \in S$ is defended by S;
- S is a *complete extension* (of F), i.e. $S \in comp(F)$, if $S \in adm(F)$ and for each $a \in A$ defended by S (in F), $a \in S$ holds.

Exercise 6.1:
We want to compute all extensions of a given semantics (stable, admissible or complete) with a CSP. Let $F = (A, R)$ be an AF, formulate for each semantics the associated CSP, such that the solutions of the CSP correspond to the extensions of the AF F.