Conflicts in Abstract Argumentation Systems

Diego C. Martínez[1]
dcm@cs.uns.edu.ar

Alejandro J. García
ajg@cs.uns.edu.ar

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial
Departamento de Ciencias de la Computación – Universidad Nacional del Sur
Avenida Alem 1253 – (0291) 4595135 – Bahía Blanca – Argentina

1. Introduction

In this work we explore the inclusion of the notion of multiple argument conflicts, those in which two or more arguments are involved.
In formal systems of defeasible argumentation, arguments for and against a proposition are produced and evaluated to verify the acceptability of that proposition.
The development of argumentation systems has grown in the last years [AG95, BV, Dung93, PRAK, Sim92, GS99] but no consensus has been reached yet on some issues, such as the representation of arguments, the way they interact, and the output of that interaction. Even then, the main idea in these systems is that any proposition will be accepted as true if there exists an argument that supports it, and this argument is acceptable according to an analysis between it and its counterarguments. Therefore, in the set of arguments of the system, some of them will be *acceptable* or *justified* arguments, while others not.
Almost every system of this kind is based on the notion of binary conflicts between arguments. We consider here the existence of a more complex form of conflict, how to solve it, and the corresponding acceptability semantic.

2. Abstract Argumentation Systems

Prakken says in [PVREE], that "*argumentation systems are built around an underlying logical language and an associated notion of logical consequence, defining the notion of an argument*". This is called the *logical level* of the system, which fits with the standard form of any logical system. The first element, which makes an argumentation system a framework of defeasible reasoning, is the notion of *conflict* between arguments. There are different types of conflict, but all of them are based in the notion of contradiction in the logical language.
Every language must include some mechanism to denote contradictory knowledge. There are basically two forms to achieve this. The most common way to do this is by means of a symbol "¬" to denote the contradiction of any proposition, as usual in propositional logic. This will be called *negation of propositions*. An alternative way to represent contradiction requires the inclusion in the language of a special element, usually denoted by "⊥" and called *falsum* Any set of propositions which leads to *falsum* is considered itself contradictory. This fact can be specified by means of rules like this one:

$$p,q,r \rightarrow \bot \quad (1)$$

here, the propositions p,q and r form a contradiction. This form to denote contradiction in the language will be called *contradiction by definition*.

The notion of conflict between arguments is closely related to the mechanism of contradiction used in the language. The negation of propositions leads to conflicts only between two arguments. The negation by definition leads to conflicts between two or more arguments. The rule (1) defines a

[1] Becario de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires.

conflict between an argument for *p*, an argument for *q* and another argument for *r*. The conflict appears only when the three arguments are considered together, as shown in the next example.

Example 2.1 *The arguments*

```
a = [canadian(Peter)].
b = [quebecois(Peter)].
c = [separatist(Peter)].
```

are in conflict under the rule

$$\texttt{canadian(X), quebecois(X), separatist(X)} \rightarrow \bot$$

because, for example, any person from Quebec who is separatist is not considered itself canadian. In the same way, any quebecois unionist is considered itself canadian, quebecois, but obviously not separatist.

An abstract argumentation system with this kind of conflicts between more than two arguments can be defined in the abstract level used in Dung [Dung93].

Definition 2.1 [Argumentation Framework] An abstract argumentation framework is a tuple <AR,At,On> where AR is a set of arguments, $At \in P(AR)$ denotes conflicts between arguments, and $On \subseteq AR \times AR$

The set On represents the partial order between arguments of the set AR. The conflicts of At may involve more than two arguments. The set $\{A_1,A_2,...,A_n\} \in At$ represents a conflict between the arguments $A_1,A_2,...,A_n$. A conflict is *binary* if the cardinality of the set is two, it is *multiple* if the cardinality is equal or greater than two.
The partial order in AR helps to solve the conflicts in the system. The pair (A,B) in On means "A is as strong as B". For simplicity and consistency with other works, it can be denoted $A \leq B$. If B is stronger than A this is noted by $A<B$.
As usual, binary conflicts are solved selecting the weakest argument of the conflictive pair, and establishing a relation of defeat between this argument and the other. For example, if {A,B} is a conflict and $A<B$, then A is defeated by B, because the latter is stronger. This defeat relation can be denoted by $A \xrightarrow{d} B$.
Multiple conflicts are solved in a similar way. The subset of weakest arguments must be identified in order to establish a defeat relation. This relation has the form

$$S_1 \xrightarrow{d} S_2$$

where S_1 and S_2 are sets of arguments, the first named *defeater set* and the second named *defeated set*.
For a set S of conflictive arguments, the defeated set is formed by all the weakest arguments in S. An argument A is a weakest argument of a set S if it is not better than any other argument in S.

A new defeat relation

The multiple defeat relation can be interpreted as "if all the arguments in the set S_1 are accepted, then the arguments of the set S_2 can not be accepted all together".
This can be viewed as the specification of conditional conflicts, because the phrase "...can not be accepted all together" is also the meaning of any conflictive set. Therefore, there is a conflict in the system (between the arguments in the *defeated set*) which appears only when a set of arguments are accepted (those in the *defeater set*).
The process of determining the defeat relation for a set S of conflictive arguments is basically the process of identifying the subset S_2 with the weakest arguments of S. Note that each argument in S_2 can not be compared with any argument in S_2.

An alternative notion of acceptability of arguments [Dung93] can be defined for this argumentation framework, which takes into account the possibility of conditional conflicts.

Definition 2.2 [Acceptability of arguments] A set of arguments *Arg* is acceptable with respect to a set of arguments *S*, if for all defeat relation

$$S_1 \xrightarrow{d} S_2$$

if $S_2 \subseteq S \cup Arg$, then exists at least an argument $d \in S_1$ such that exists a subset $B \subseteq S$ such that

$$B \xrightarrow{d} d$$

Note that the notion of acceptability is based only in defeat relations, because unsolved conflicts can not be determinant in the system.

Present and future work

Multiple arguments conflicts leads to more complex defeat relations, which can be interpreted as conditional conflicts. This kind of defeat and its expresive power is actually under research.
The framework is defined without committing to any logical language. This level of abstraction is intentional, because we are trying to define the general behaviour of systems with multiple conflicts. The behaviour of this framework and its semantics under binary conflicts is equivalent to Dung's argumentation frameworks [Dung93]. Multiple arguments conflicts can be solved in different ways. One of the next step is to establish differences and similarities of this new acceptability semantic with existing semantics for argumentation frameworks with this kind of contradiction in its logical languages.

Bibliography

[AG97] Alejandro García. *La programación en lógica rebatible. Su definición teórica y computacional.* Tesis de Magister en Cs. de la Computación. Universidad Nacional del Sur. 1997.

[BV] Bart Verheij. *Rules, reasons and arguments: formal studies of argumentation and defeat.* PhD Thesis. ISBN 90-9010071-7

[DCM2k] Diego C. Martínez, Alejandro J. García. *Una metodología para el tratamiento de la indecisión en los sistemas argumentativos abstractos.*. Congreso Argentino de Ciencias de la Computacion CACIC 2000. Ushuaia, Octubre de 2000.

[DUNG93] P. M Dung. *On the Acceptability of Arguments and its Fundamental Role in Nonmotonic reasoning and Logic Programming and N-persons games*, IJCAI 93 852-857, 1993.

[GS99] Alejandro García, Guillermo Simari. *Defeasible Argumentation for Extended Logic Programming.* Submitted to the 5th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'99), El Paso TX, Dic. 99.

[PRAK] Henry Prakken y Giovanni Sartor. *Argument-based extended logic programming with defeasible priorities* Journal of applied non classical logic 7, 25-75.

[PVREE] Henry Prakken y Gerard Vreeswijk. *Logics for Defeasible Argumentation*. Survey in defeasible argumentation.

[SIM92] G.R Simari y R.P. Loui. *Mathematical Treatment of Defeasible Reasoning and its Implementation.* Artificial Intelligence, 53:125--157,1992.

[VREE97] Gerard A.W. Vreeswijk. *Abstract argumentation systems.* Artificial Intelligence, 90:225--279, 1997.