Having FUN with CATs and Arguments

Bas van Gijzel

University of Nottingham

May 24, 2012
Argumentation theory

Interdisciplinary area with various applications:
Argumentation theory

Interdisciplinary area with various applications:

- **Law:**
  System **modelling** legal problems/cases,
Argumentation theory

Interdisciplinary area with various applications:

- **Law:** Systems *modelling* legal problems/cases,
- **Decision making:** *Organising* information and source of *efficiency* in decision theory,
Argumentation theory

Interdisciplinary area with various applications:

- **Law:**
  Systems *modelling* legal problems/cases,

- **Decision making:**
  *Organising* information and source of *efficiency* in decision theory,

- **Communication theory:**
  *Making* argumentation in existing texts *precise*. 
Argumentation

What is argumentation?
Argumentation

In its basic setting:

1. Construct arguments **in favour** and **against** a certain statement,
Argumentation

In its basic setting:

1. Construct arguments in favour and against a certain statement,

2. Select the acceptable arguments,
Argumentation

In its basic setting:

1. Construct arguments in favour and against a certain statement,
2. Select the acceptable arguments,
3. Determine whether the statement holds.
1. Abstract argumentation

2. Structured argumentation
   - ASPIC$^+$
   - Carneades

3. Putting in CATs and FUN: a glimpse
   - FUN with Carneades
   - CATs: Logic of Argumentation and ASPIC$^+$
1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$
Dung’s argumentation frameworks (AFs)

In 1995, Dung gave an abstract account of argumentation.

• Was able to model several contemporary approaches to non-monotonic logic,
Dung’s argumentation frameworks (AFs)

In 1995, Dung gave an abstract account of argumentation.

• Was able to model several contemporary approaches to non-monotonic logic,

• Some scholars believe it to be too abstract,
Dung’s argumentation frameworks (AFs)

In 1995, Dung gave an abstract account of argumentation.

- Was able to model several contemporary approaches to non-monotonic logic,
- Some scholars believe it to be too abstract,
- However the model can be instantiated with more structure
  - For instance: ASPIC\(^+\).
Definition (1)

Two parts: a set of abstract arguments and an (abstract) notion of defeat between arguments.

\[ C \quad B \quad A \]
Definition (1)

Two parts: a set of abstract arguments and an (abstract) notion of defeat between arguments.

\[ C \rightarrow B \rightarrow A \]
Definition (2)

Formally:
An abstract argumentation framework is a tuple $AF = \langle \text{Args}, \text{defeats} \rangle$ such that:

• $\text{Args}$ is a set of (abstract) arguments,
• $\text{defeats} \subseteq \text{Args} \times \text{Args}$.

In other words a directed graph.
Formally:
An abstract argumentation framework is a tuple $AF = \langle \text{Args, defeats} \rangle$ such that:

- $\text{Args}$ is a set of (abstract) arguments,
- $\text{defeats} \subseteq \text{Args} \times \text{Args}$. 
Definition (2)

Formally:
An abstract argumentation framework is a tuple \( AF = \langle \text{Args}, \text{defeats} \rangle \) such that:
- \( \text{Args} \) is a set of (abstract) arguments,
- \( \text{defeats} \subseteq \text{Args} \times \text{Args} \).

In other words a directed graph.
Evaluation of an acyclic Dung framework can be done by assigning arguments a status from \{in, out\}.
Evaluation (1)

\[ C \rightarrow B \rightarrow A \]
Evaluation (1)

$C \rightarrow B \rightarrow A$
Evaluation (1)

\[ C \rightarrow B \rightarrow A \]
Evaluation (1)

C → B → A
What about?

$B \leftrightarrow A$
Evaluation (2)

Two possible labellings using status $= \{\text{in}, \text{out}\}$:

$$B \leftrightarrow A \text{ and } B \leftrightarrow A$$
Two possible labellings using status $= \{\textit{in}, \textit{out}\}$:

\[ B \leftrightarrow A \text{ and } B \leftrightarrow A \]

or three possible labellings using status $= \{\textit{in}, \textit{out}, \textit{undecided}\}$:

\[ B \leftrightarrow A \]
Dung defined different semantics for dealing with cycles using extensions.

- Complete extension
- Grounded extension
- Preferred extension
- Stable extension
Dung defined different semantics for dealing with cycles using extensions.

- **Complete** extension
- **Grounded** extension
- **Preferred** extension
- **Stable** extension

Where extension =

“set of arguments that are acceptable when taken together”
Outline

1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$
Outline

1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$
Structured argumentation: \textit{ASPIC}^+
ASPIC+ by Prakken (2010) starts with:

- **Knowledge:**
  indisputable facts, assumptions, normal premises, issue premises.

- **Inference rules:**
  strict or defeasible.

Arguments are given **structure**.
Arguments in ASPIC$^+$ (1)

Argument structure:

• **Trees** where:
Arguments in ASPIC\(^+\) (1)

Argument structure:

- **Trees** where:
  
  - **Nodes** are wff of a logical language \( \mathcal{L} \),
  
  - **Links** are applications of inference rules:
    
    - \( \mathcal{R}_s \), strict rules of the form: \( \phi_1, \ldots, \phi_n \rightarrow \phi \),
    
    - \( \mathcal{R}_d \), defeasible rules of the form: \( \phi_1, \ldots, \phi_n \Rightarrow \phi \),
    
  - Smallest argument is a fact from knowledge base \( \mathcal{K} \subseteq \mathcal{L} \).
Arguments in ASPIIC$^+$ (1)

Argument structure:

- **Trees** where:
  - **Nodes** are *wff* of a logical language $\mathcal{L}$,
  - **Links** are applications of inference rules:
    - $\mathcal{R}_s$, strict rules of the form: $\phi_1, \ldots, \phi_n \rightarrow \phi$,
    - $\mathcal{R}_d$, defeasible rules of the form: $\phi_1, \ldots, \phi_n \Rightarrow \phi$,
  - Smallest argument is a fact from knowledge base $\mathcal{K} \subseteq \mathcal{L}$.
- **Acceptability of arguments** determined by corresponding Dung framework.
The defeat relation is given structure as following:

- Determine possible attacks, based on:
  - Contrariness relation: $\neg$, (asymmetric “negation”)
  - Structure of argument: attacks possible on premises, rules and conclusions.
The defeat relation is given structure as following:

- Determine possible attacks, based on:
  - Contrariness relation: $\neg$, (asymmetric “negation”)
  - Structure of argument: attacks possible on premises, rules and conclusions.

Inclusion of attacks into the defeat relation depends on:

- Preferences on rules,
- Preferences on knowledge.
Structured evaluation

B: \( \frac{CATs \quad FUN}{\neg normal\_talk} \)

A: \( \frac{Bas\_talks}{argumentation\_talk} \)
\( \frac{boring?}{normal\_talk} \)
Structured evaluation

B: \[ \frac{CATs \quad FUN}{\neg normal\_talk} \]

A: \[ \frac{Bas\_talks}{\frac{argumentation\_talk}{boring?}} \quad normal\_talk \]

Evaluated through Dung:

\[ B \rightarrow A \]
Outline

1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$

- **Structured argumentation** (like ASPIC$^+$):
  - Arguments are *simple trees*, with exactly one inference step,
  - Language is *propositional*,
  - Notion of conflict is standard *propositional negation*. 

**Carneades**
Pro and con arguments

Two types of arguments regarding a conclusion $c$:
- An argument with conclusion $c$ is called pro $c$,
- An argument for an opposite conclusion, $\bar{c}$, is called con $c$. 
Pro and con arguments

Two types of arguments regarding a conclusion $c$:

- An argument with conclusion $c$ is called \textit{pro} $c$,
- An argument for an opposite conclusion, $\overline{c}$, is called \textit{con} $c$.

“Aggregation” of \textit{pro} and \textit{con} is done through \textit{proof standards}. 

Arguments in Carneades visualised

- $a_1$ is a factor with a high impact on murder.
- $a_2$ is a factor with a moderate impact on intent.
- $a_3$ is a factor with a moderate impact on ~intent.

Factors:
- intent
- kill
- witness
- unreliable
- witness2
- unreliable2

Probabilities:
- 0.8 for murder from $a_1$
- 0.3 for intent from $a_2$
- 0.3 for ~intent from $a_3$
Proof standards

Five proof standards:
Proof standards

Five proof standards:

• Scintilla of evidence,
• Preponderance of the evidence,
• Clear and convincing evidence,
• Beyond reasonable doubt,
• Dialectical validity.
Beyond reasonable doubt (informal)

Given arguments pro $p$ and con $p$, $p$ holds under beyond reasonable doubt iff:

- There exists an argument pro $p$, $a$, which is applicable and $weight(a) > \alpha$,
Beyond reasonable doubt (informal)

Given arguments pro $p$ and con $p$, $p$ holds under beyond reasonable doubt iff:

• There exists an argument pro $p$, $a$, which is applicable and $\text{weight}(a) > \alpha$,

• $\text{weight}(a)$ exceeds the weight of the applicable con arguments by $\beta$, 
Beyond reasonable doubt (informal)

Given arguments pro $p$ and con $p$, $p$ holds under beyond reasonable doubt iff:

- There exists an argument pro $p$, $a$, which is applicable and $\text{weight}(a) > \alpha$,
- $\text{weight}(a)$ exceeds the weight of the applicable con arguments by $\beta$,
- The weight of all applicable con arguments is less than $\gamma$. 
Proof standards as defeat

- Proof standards are assigned locally to propositions,
- Similar to preferences on attack but not quite?
Proof standards as defeat

- Proof standards are assigned locally to propositions,
- Similar to preferences on attack but not quite?

Structured \( \rightarrow \) \( \rightarrow \) Abstract

\( \text{Carneades} \) \( \rightarrow \) \( \text{ASPIC}^+ \) \( \rightarrow \) \( \text{Dung} \)

This is the work I’ve done for my MSc. and improved upon in an article.
Proof standards as defeat

- Proof standards are assigned locally to propositions,
- Similar to preferences on attack but not quite?

Structured \(\rightarrow\) Structured \(\rightarrow\) Abstract

\[\text{Carneades} \quad \text{translates to} \quad \text{ASPIC}^+ \quad \text{translates to} \quad \text{Dung}\]

This is the work I’ve done for my MSc. and improved upon in an article. Might be nice, but just proved correct, not implemented!
Outline

1. Abstract argumentation

2. Structured argumentation
   - ASPIC
   - Carneades

3. Putting in CATs and FUN: a glimpse
   - FUN with Carneades
   - CATs: Logic of Argumentation and ASPIC
Outline

1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$
State of existing implementation

- Well-developed implementation of Carneades in Clojure\(^1\).
- A large part is focused on user interaction, efficiency.
- Code relatively easy to read, but hard to see the formal relation.

\(^1\)http://carneades.github.com
State of existing implementation

- Well-developed implementation of Carneades in Clojure\(^1\).
- A large part is focused on user interaction, efficiency.
- Code relatively easy to read, but hard to see the formal relation.
  - Implementation is more general than the original model.
  - Function definitions behave differently from the original definitions.

\(^1\)http://carneades.github.com
Representing constructions such as proof standards directly:

```haskell
  type ProofStandard = Proposition \rightarrow CAES \rightarrow Bool
  type Weight       = Argument \rightarrow Double
```

where a CAES is a Carneades argument evaluation structure, grouping arguments, etc. together.
Carneades in Haskell (2)

Giving a functional implementation of proof standards:

\[\text{beyond\_reasonable\_doubt} :: \text{ProofStandard}\]
\[\text{beyond\_reasonable\_doubt } p \text{ caes@}(\text{CAES } (g, (_, \text{weight})), _))\]
\[= \text{maxWeight} p > \alpha \land\]
\[\text{maxWeight} p > \text{maxWeight} \neg p + \beta \land\]
\[\text{maxWeight} \neg p < \gamma\]

where

\[\text{proArgs} = \text{getArgs } p \text{ g}\]
\[\text{conArgs} = \text{getArgs } (\text{negation } p) \text{ g}\]
\[\text{applicableArgs} = \text{filter } (\text{‘applicability’caes})\]
\[\text{maxWeight} = \text{maxWeightBy } \text{weight}\]
\[\text{maxWeight} p = \text{maxWeight }\]
\[\text{applicableArgs } \text{proArgs}\]
\[\text{maxWeight} \neg p = \text{maxWeight }\]
\[\text{applicableArgs } \text{conArgs}\]
Results

Carneades in Haskell gives us:

• **High-level code close to the mathematical definitions:**
  • Allowing greater understanding of the implementation,
  • Easier realisation of existing/future translations,
  • Giving a DSL already familiar to argumentation theorists.

• **No separate** parser, compiler, etc.
Carneades in Haskell gives us:

- **High-level code close to the mathematical definitions:**
  - Allowing greater understanding of the implementation,
  - Easier realisation of existing/future translations,
  - Giving a DSL already familiar to argumentation theorists.
- **No separate** parser, compiler, etc.

Written up with Henrik and to be presented at TFP.
Future directions

Transfer the functional definitions to an interactive theorem prover, such as Agda.

• Possible to prove properties of the implementation,
• Possible to prove a translation keeps desired properties.
1 Abstract argumentation

2 Structured argumentation
   ASPIC$^+$
   Carneades

3 Putting in CATs and FUN: a glimpse
   FUN with Carneades
   CATs: Logic of Argumentation and ASPIC$^+$
The **Logic of Argumentation** is an argumentation model by Krause et al. (1995), based on the Curry-Howard-Lambek correspondence.

- types $\leftrightarrow$ propositions
The **Logic of Argumentation** is an argumentation model by Krause et al. (1995), based on the Curry-Howard-Lambek correspondence.

- types ↔ propositions ↔ objects of a Cartesian Closed Category.
Logic of Argumentation (2)

Building on the Curry-Howard-Lambek correspondence:

- **Propositions** are objects,
- **Arguments** or uncertain proofs are arrows:
  - Thus arguments are lambda terms,
  - Open variables in arguments are assumptions.
- Ordering information on arguments/arrows (semi-lattice), giving an enriched category,
- Functions (homomorphisms) from this ordering to a dictionary of weights.
Logic of Argumentation (2)

Building on the Curry-Howard-Lambek correspondence:

- **Propositions** are objects,
- **Arguments** or uncertain proofs are arrows:
  - Thus **arguments** are lambda terms,
  - **Open variables** in arguments are assumptions.
- **Ordering information** on arguments/arrows (semi-lattice), giving an enriched category,
- **Functions** (homomorphisms) from this ordering to a dictionary of weights.

So what?
The model is from before Dung (1995), however:

- This is the model with the most general and elegant description of argument aggregation,
- Argument aggregation has never been combined with defeat properly (not even in LA).
Combine aggregation and defeat:

- **Implemented** in Haskell a version of ASPIC$^+$ that contains proof standards as defeat (first implementation),
- Properly **instantiated** LA’s argument aggregation to this version of ASPIC$^+$, giving a system that combines both,
- Then **implemented** this as well.
Results

Combine aggregation and defeat:

• **Implemented** in Haskell a version of ASPIC$^+$ that contains proof standards as defeat (first implementation),
• Properly **instantiated** LA’s **argument aggregation** to this version of ASPIC$^+$, giving a system that combines both,
• Then **implemented** this as well.

Draft paper and a typesetted implementation lying around.
Future directions

Lift back to the nice world of category theory:

1. Give proof standards a categorical equivalent,
2. Give a proper account of defeat by generalising LA or changing its notion/propagation of contradiction.
Future directions

Lift back to the nice world of category theory:

1. Give proof standards a categorical equivalent,
2. Give a proper account of defeat by generalising LA or changing its notion/propagation of contradiction.

Possible cooperation with Nicolai and Peter McBurney.