# Logic-Based Argumentation with N-ary Graphs

## Présentation Journée GraphIK 2018

**Bruno YUN** [1] Srdjan VESIC [2] Madalina CROITORU [1]

1. Univ. Montpellier, LIRMM
2. CNRS, CRIL, Lens

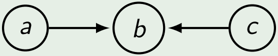
# What is argumentation ? (Part 1)

- Argumentation is a way of reasoning that is based on arguments and attacks between them.

# What is argumentation ? (Part 1)

- Argumentation is a way of reasoning that is based on arguments and attacks between them.
- It is usually modelled using the *Dung's framework* and represented as a directed graph.

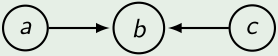
**Example**



# What is argumentation ? (Part 1)

- Argumentation is a way of reasoning that is based on arguments and attacks between them.
- It is usually modelled using the *Dung's framework* and represented as a directed graph.

**Example**




- Other generalisations exist and implement supports relation, weights on attacks, preferences on arguments, different kind of attacks and arguments, etc.

# What is argumentation ? (Part 2)

- We use *argumentation semantics* (preferred, stable semantics) to extract meaningful consistent subsets of the set of arguments.

**Example**


The set $\{a, c\}$ is a preferred extension (maximal conflict-free and defend itself).

# Inconsistent knowledge base in *Datalog*$^{\pm}$

We consider inconsistent *Datalog*$^{\pm}$knowledge bases.

**Example (Knowledge Base)**

- $\mathcal{F} = \{contains(m, saltC), contains(m, sugar), contains(m, yogurt), notSoup(m), \ edible(m)\}$
- $\mathcal{R} = \{\forall x(contains(x, saltC) \wedge contains(x, yogurt) \rightarrow tzaziki(x))\}$
- $\mathcal{N} = \{\forall x(contains(x, saltC) \wedge contains(x, sugar) \wedge contains(x, yogurt) \rightarrow \bot), \forall x(tzaziki(x) \wedge notSoup(x) \rightarrow \bot)\}$

# Why n-ary argumentation ?

Let $\mathcal{K}$ be a KB, $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ where $\mathcal{A}$ is a set of arguments and $\mathcal{C}$ a set of attacks defined as follows.

## Definition (Old arguments)

An *argument* is a tuple $(H, C)$ with $H$ a non-empty $\mathcal{R}$-consistent subset of $\mathcal{F}$ and $C$ a set of facts such that :

- $H \subseteq \mathcal{F}$ and $H$ is $\mathcal{R}$-consistent *(consistency)* ;
- $C \subseteq \mathcal{C}\ell_{\mathcal{R}}(H)$ *(entailment)* ;
- $\nexists H' \subset H$ s.t. $C \subseteq \mathcal{C}\ell_{\mathcal{R}}(H')$ *(minimality)*.

## Example

An argument is :
$a_1 = (\{contains(m, saltC), contains(m, yogurt)\}, \{tzaziki(m)\})$

# Why n-ary argumentation ?

Let $\mathcal{K}$ be a KB, $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ where $\mathcal{A}$ is a set of arguments and $\mathcal{C}$ a set of attacks defined as follows.

**Definition (Old attacks)**

We say that $a = (H, C)$ attacks $b = (H', C')$ denoted by $(a, b) \in \mathcal{C}$ iff there exists $\phi \in H'$ such that $C \cup \{\phi\}$ is $\mathcal{R}$-inconsistent.

**Example**

$a_1 = (\{contains(m, saltC), contains(m, yogurt)\}, \{tzaziki(m)\})$ attacks $a_2 = (\{notSoup(m)\}, \{notSoup(m)\})$.

## Why n-ary argumentation ?

- **Problem : we have too many arguments (and attacks). Here, we have 33 arguments and 360 attacks for a knowledge base with 5 facts, 1 rule and 2 negative constraints.**

# Why n-ary argumentation ?

- **Problem : we have too many arguments (and attacks). Here, we have 33 arguments and 360 attacks for a knowledge base with 5 facts, 1 rule and 2 negative constraints.**
- We need a way for arguments to **jointly** attack other arguments.

**Example**

a : “Martin is on the tandem bicycle”
b : “Madalina is on the tandem bicycle”
c : “Pierre is on the tandem bicycle”
**We need attacks of the form** $(\{a, b\}, c)$

# The new framework

Let us consider the KB $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$. $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ with $\mathcal{C} \subseteq 2^{\mathcal{A}} \times \mathcal{A}$ is such that :

## Definition (Argument)

An argument $a \in \mathcal{A}$ is :

- $f$, where $f \in \mathcal{F}$. $Conc(a) = f$ and $Prem(a) = \{f\}$
- $a_1, \ldots, a_n \rightarrow f'$ if $a_1, \ldots, a_n$ are arguments such that there exists a tuple $(r, \pi)$ where $r \in \mathcal{R}$, $\pi$ is a homomorphism from the body of $r$ to $\{Conc(a_1), \ldots, Conc(a_n)\}$ and $f'$ is the resulting atom from the rule application. $Conc(a) = f'$ and $Prem(a) = Prem(a_1) \cup \cdots \cup Prem(a_n)$

where $Prem(a)$ is $\mathcal{R}$-consistent.

# The new framework

Let us consider the KB $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$. $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ with $\mathcal{C} \subseteq 2^{\mathcal{A}} \times \mathcal{A}$ is such that :

## Definition (Attack)

An attack is a pair $(X, a)$ where $X \subseteq \mathcal{A}$ and $a \in \mathcal{A}$ such that $X$ is minimal for set inclusion such that $\bigcup\limits_{x \in X} Prem(x)$ is $\mathcal{R}$-consistent and there exists $\varphi \in Prem(a)$ such that $(\bigcup\limits_{x \in X} Conc(x)) \cup \{\varphi\}$ is $\mathcal{R}$-inconsistent.

# The new framework

## Example

Let $\mathcal{K}$ be the previous knowledge base about the choice of an appetiser, the resulting argumentation graph is composed of six arguments and 11 attacks :

- $a_1 = contains(m, sugar)$
- $a_2 = contains(m, saltC)$
- $a_3 = contains(m, yogurt)$
- $a_4 = notSoup(m)$
- $a_5 = edible(m)$
- $a_6 = a_2, a_3 \to tzaziki(m)$

An example attack of $\mathcal{C}$ is $(\{a_1, a_2\}, a_3)$.

# Properties

- We have the one-to-one correspondence between preferred/stable extensions and maximal consistent subset of facts.

## Example

$$\{contains(m, saltC), contains(m, yogurt), edible(m)\}$$
$$\updownarrow$$
$$\{a_2, a_3, a_5, a_6\}$$

# Properties

- We have an upper-bound on the number of attacks with respect to the number of arguments and we have an upper-bound on the number of arguments if there are no rules.

**Example (Attack upper-bound)**

Let $\mathcal{K}$ be a knowledge base and $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ be the corresponding argumentation framework. If $|\mathcal{A}| = n$ then $|\mathcal{C}| \leq \sum_{i=1}^{n-1} \binom{n}{i}(n-i)$.

# Properties

- We satisfy the basic rationality postulates (closure, indirect and direct consistency)

**Example (Indirect consistency)**

Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a knowledge base, $\mathcal{AS}_{\mathcal{K}}$ the corresponding argumentation framework and $x \in \{s, p, g\}$. Then :

- for every $E \in Ext_x(\mathcal{AS}_{\mathcal{K}})$, $Concs(E)$ is a $\mathcal{R}$-consistent.
- $Output_x(\mathcal{AS}_{\mathcal{K}})$ is $\mathcal{R}$-consistent.

# Properties

- Presence of structural properties (cycle, etc.)

**Example (Self-attacking Arguments)**

Let $\mathcal{K}$ be a knowledge base and $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ be the corresponding argumentation framework. There is no $(S, t) \in \mathcal{C}$ such that $t \in \mathrm{S}$.

**Example (Defense)**

Let $\mathcal{K}$ be a knowledge base and $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ be the corresponding argumentation framework. If there is $(S, t) \in \mathcal{C}$ then there exists $(S', s) \in \mathcal{C}$ such that $s \in S$.

# Experimentation & Results

- We generated this n-ary argumentation graph on a set of 134 existing knowledge bases.

# Experimentation & Results

- We generated this n-ary argumentation graph on a set of 134 existing knowledge bases.
  - A set of $A$ composed of 108 knowledge bases. This dataset is further split into three smaller set of knowledge bases :
    - A set of $A_1$ of 31 knowledge bases without rules, two to seven facts, and one to three negative constraints.
    - A set $A_2$ of 51 knowledge bases generated by fixing the size of the set of facts and successively adding negative constraints until saturation.
    - A set $A_3$ of 26 knowledge bases with only ternary negative constraints, three to four facts and one to three rules.
  - A set $B$ of 26 knowledge bases with eight facts, six rules and one or two negative constraints. This set contains more free-facts than the knowledge bases in set $A$.

# Experimentation & Results

- We generated this n-ary argumentation graph on a set of 134 existing knowledge bases.
- We compared the number of argument and attacks with the existing argumentation framework.

# Experimentation & Results

- We generated this n-ary argumentation graph on a set of 134 existing knowledge bases.
- We compared the number of argument and attacks with the existing argumentation framework.

| | Old Framework | | |
|---|---|---|---|
| $\mathcal{K}$ | # Arg. | # Att. | Gen. Time (ms) |
| $A_1$ | 22 | 128 | 160 |
| $A_2$ | 25 | 283 | 133 |
| $A_3$ | 85 | 1472 | 399,5 |
| B | 5967 | 11542272 | 533089 |

# Experimentation & Results

- We generated this n-ary argumentation graph on a set of 134 existing knowledge bases.
- We compared the number of argument and attacks with the existing argumentation framework.

| | New Framework | | | | | |
|---|---|---|---|---|---|---|
| $\mathcal{K}$ | # Arg. | % Arg. ↓ | # Att. | % Att. ↓ | G. Time | % Time ↕ |
| $A_1$ | 5 | 77,27 | 6 | 93,75 | 276,00 | -81,48 |
| $A_2$ | 7 | 72,00 | 8 | 92,93 | 342,00 | -183,57 |
| $A_3$ | 7 | 91,76 | 9 | 99,26 | 369,50 | 1,66 |
| B | 14 | 99.77 | 20.5 | 99.99 | 7814.5 | 98.08 |

# Naked : N-ary Argumentation graphs from Knowledge bases Expressed in $Datalog^{\pm}$

We developed the Naked tool for visualising and generating n-ary graphs from $Datalog^{\pm}$knowledge bases.