

The attack as strong negation, part I

D. GABBAY*, *King's College London, Department of Informatics, Strand, London, WC2R 2LS, UK; Bar Ilan University, Ramat Gan, Israel and University of Luxembourg, Luxembourg.*

M. GABBAY**, *Cambridge University, UK.*

Abstract

We add strong negation N to classical logic and interpret the attack relation of 'x attacks y' in argumentation as $(x \rightarrow Ny)$. We write a corresponding object level (using N only) classical theory for each argumentation network and show that the classical models of this theory correspond exactly to the complete extensions of the argumentation network. We show by example how this approach simplifies the study of abstract argumentation networks. We compare with other translations of abstract argumentation networks into logic, such as classical predicate logic or modal logics, or logic programming, and we also compare with Abstract Dialectical Frameworks.

Keywords: Intuitionistic logic, strong negation, argumentation network, disjunctive attack, higher level attack.

1 Background from classical logic: non-logical axiomatic theories

This article introduces a particularly intuitive and simple representation/translation of abstract argumentation networks into classical propositional logic. All we need is a simple version of strong negation. So our starting point must be to introduce this negation.

Classical propositional logic can be properly axiomatized in many ways. For simplicity, let us take the set \mathbf{T} of all tautologies as axioms and the rule of modus ponens as the deduction rule. Let us assume that the connectives used are the usual ones $\{\neg, \wedge, \vee, \rightarrow, \perp, \top\}$ and the atomic sentences are the set $P = \{p_1, p_2, p_3, \dots\}$. Classical logic is strongly complete for the classical semantics. Models are assignments h giving values in $\{0, 1\}$ to the atoms of P .

We have satisfaction defined as follows for wffs A and theories Δ .

- $h \models p$ iff $h(p) = 1$, for $p \in P$
- $h \models \top$
- $h \not\models \perp$
- $h \models \neg A$ iff $h \not\models A$
- $h \models A \wedge B$ iff $h \models A$ and $h \models B$
- $h \models A \vee B$ iff $h \models A$ or $h \models B$
- $h \models A \rightarrow B$ iff $h \models A$ implies $h \models B$.
- $h \models \Delta$ iff $h \models A$ for all $A \in \Delta$.

The notion of proof $A_1, \dots, A_n \vdash B$ can be defined in classical logic in many ways and completeness holds for any set of wffs Δ and any A :

- $\Delta \vdash A$ iff for all models h we have $h \models \Delta$ implies $h \models A$.

We now introduce the notion of a set of sentences being a specific non-logical axiomatic theory Θ_n .

*E-mail: dov.gabbay@kcl.ac.uk

**E-mail: mg639@cam.ac.uk

2 The attack as strong negation

Consider again the set of atomic wff

$$P = \{p_1, p_2, p_3, \dots\}.$$

Suppose we insist, for our own reasons, that we want to consider only those models h satisfying the restriction **(n)** below:

(n) For all even index atoms $p_2, p_4, p_6, \dots, p_{2i}, \dots$ we have $h(p_{2i}) = 1$ implies $h(p_{2i-1}) = 0, i = 1, 2, 3, \dots$

There is a theory Θ_n of classical propositional logic whose models are exactly all the models satisfying **(n)**. This theory is

$$\Theta_n = \{p_2 \rightarrow \neg p_1, p_4 \rightarrow \neg p_3, p_6 \rightarrow \neg p_5, \dots\}$$

Let us now for the sake of clarity, rename the atoms of P with the help of a new symbol N . We rename as follows:

$$\begin{aligned} q_1 &= \text{def. } p_1 \\ Nq_1 &= \text{def. } p_2 \\ q_2 &= \text{def. } p_3 \\ Nq_3 &= \text{def. } p_4 \\ &\vdots \\ q_i &= \text{def. } p_{2i-1} \\ Nq_i &= \text{def. } p_{2i} \\ &\vdots \end{aligned}$$

We can thus write the set of atoms as P as the set Q

$$Q = \{q_1, Nq_1, q_2, Nq_2, \dots\}.$$

The theory Θ_n becomes the theory

$$\Theta_n = \{Nq \rightarrow \neg q \mid q \in \{q_1, q_2, \dots\}\}$$

Θ_n is considered a non-logical set of axioms on the symbol N .

Note that N cannot be iterated and can be applied only to atoms taken from $\{q_1, q_2, \dots\}$.

We said that we regard Θ_n as a non-logical axiomatic theory on the symbol N . This is common practice in logic and model theory. Consider, e.g., the classical theory of Abelian groups formalized in classical logic for the multiplication symbol $*$ and the constant $\mathbf{1}$. We add to the logical axioms of predicate logic the non-logical group axioms below

- $\forall xyz(x*(y*z) = (x*y)*z)$
- $\forall xy(x*y = y*x)$
- $\forall x(x*\mathbf{1} = x)$
- $\forall x\exists y(x*y = \mathbf{1})$

In our case our non-logical axioms are $\Theta_n = \{Nq \rightarrow \neg q\}$.

Thus our logic for N is a theory (of N) within classical propositional logic, much in the same way as the theory of Abelian groups is a theory within the classical predicate calculus. The next section defines the logic **CN** formally.

2 The logic CN

We add to classical propositional logic the strong negation symbol N . We can thus form atomic sentences like $\{q_1, \dots, q_n, \dots\}$ as well as atoms of the form $\{Nq_1, Nq_2, \dots\}$. We do this as explained in Section 1.

At this point we do not allow iterations of N . We have the usual connectives \neg (negation), $\wedge, \vee, \rightarrow, \top, \perp$. We shall discuss iterated use of N in a later section. We require the only additional non-logical axioms $\Theta_n = \{Nq \rightarrow \neg q\}$. We can thus view Nq as strong negation. For example q might be true or q might be false (i.e. $\neg q$ is true) or q might be strongly false (Nq true) or q might be false but not strongly false ($\neg q$ and $\neg Nq$ true).

Let us define our logic directly.

DEFINITION 2.1 (The CN classical propositional logic with atomic strong negation)

- (1) Let \mathbb{L}_N be a language with a set of atoms $\mathcal{Q}_0 = \{q_1, q_2, \dots, q_n, \dots\}$ and the connectives $\{\neg, \wedge, \vee, \rightarrow, N\}$. $\{\neg, \wedge, \vee, \rightarrow\}$ are the usual classical connectives and N is a unary connective applied once to atoms only.
- (2) An atomic formula has the form q or Nq where $q \in \mathcal{Q}_0$.
- (3) A wff has the form $A =$ atomic formula or $A \wedge B, \neg A, A \vee B, A \rightarrow B$ where A and B are wffs.
- (4) We regard as axioms for our logic CN the nonlogical set of axioms Θ_n as discussed in Section 1. The proof theory CN is relying on the proof theory of classical logic C, as follows:
 - $\Delta \vdash_{\text{CN}} A$ iff (def) $\Delta \cup \Theta_n \vdash_{\text{C}} A$.
- (5) A model h for the logic CN is an assignment h giving each atomic wff of the form q or Nq a value in $\{0, 1\}$ such that if $h(Nq) = 1$ then $h(q) = 0$.
- (6) Satisfaction is defined in the usual way.

THEOREM 2.2

CN is complete for the proposed semantics.

PROOF. Our discussion in Section 1 presented CN as a non-logical theory Θ_n of the classical propositional calculus of item 4 of Definition 2.1 defined the consequence for CN, via the classical consequence. Since we have strong completeness for C we also have it for CN. ■

REMARK 2.3

Note that what we are calling our ‘logic’ CN is actually a theory in a two sorted classical propositional logic with two sorts of atoms of the form q and Nq . In Appendix 10 we will turn our logic into a proper modal logic, which we will call CNN.

We now have enough machinery to faithfully represent abstract argumentation networks in the logic CN. This is the job of the next section.

3 Expressing argumentation networks in CN

This section will show a simple way of translating formal argumentation networks into CN. We assume we are dealing with finite argumentation networks.¹

We shall use the Caminada labelling characterization of complete extensions. See [17] for a survey. Given an argumentation network (S, R) , with $S \neq \emptyset$ and $R \subseteq S \times S$, a legitimate Caminada

¹Actually, we do not need the requirement that the network is finite. All we need is that it is finitary, namely that each point is attacked by a finite number of attackers. This will allow us to write classical wffs describing the attacks.

4 The attack as strong negation

labelling λ on S is a function $\lambda : S \mapsto \{\text{in}, \text{out}, \text{und}\}$ giving values ‘in’, ‘out’, ‘undecided’ to each $x \in S$, satisfying the following properties.

- (C1) $\lambda(x) = \text{in}$ iff either $\neg \exists y(yRx)$ or $\forall y(yRx \rightarrow \lambda(y) = \text{out})$.
- (C2) $\lambda(x) = \text{out}$ iff $\exists y(yRx \wedge \lambda(y) = \text{in})$
- (C3) $\lambda(x) = \text{und}$ iff $\forall y(yRx \rightarrow \lambda(y) \neq \text{in})$ and $\exists y(yRx \wedge \lambda(y) = \text{und})$.

Each legitimate Caminada labelling gives rise to a unique complete extension and all complete extensions can be obtained in this way. See [17].

DEFINITION 3.1

Let $\mathcal{A} = (S, R)$ be a formal argumentation network which is finitary, i.e. each point has a finite number of attackers. This means that $S \neq \emptyset$ and $R \subseteq S \times S$ is the attack relation. Define a theory $\Delta_{\mathcal{A}}$ of the logic CN as follows

- (1) We can assume that $S \subseteq Q$ (i.e. the arguments of S are identified as atoms of the logic).
- (2) $\Delta_{\mathcal{A}} = \{x \mid \neg \exists y(yRx)\} \cup \{y \leftrightarrow \bigwedge_{z \in \text{Attack}(y)} Nz \mid y \in S\} \cup \{z \rightarrow Ny \mid zRy\} \cup \{(\bigwedge_{z \in \text{Attack}(y)} \neg z) \wedge (\bigvee_{z \in \text{Attack}(y)} \neg Nz) \rightarrow \neg y \wedge \neg Ny \mid y \in S\}$, where $\text{Attack}(y) = \{z \mid zRy\}$.

THEOREM 3.2 (First Correspondence Theorem)

Let $\mathcal{A} = (S, R)$. Then the models of $\Delta_{\mathcal{A}}$ correspond exactly to the complete extensions of \mathcal{A} .

PROOF.

- (1) Let h be a model of $\Delta_{\mathcal{A}}$. We show it defines a complete extension on \mathcal{A} (note that two different models can yield the same complete extension): We use the Caminada labelling function. Let $x \in S$. Define

$$\lambda_h(x) = \begin{cases} \text{in}, & \text{if } h(x) = 1 \\ \text{out}, & \text{if } h(Nx) = 1 \\ \text{und}, & \text{if } h(x) = h(Nx) = 0 \end{cases}$$

we prove the following

- (a) λ_h is well defined. For each x we can have exactly one of the three cases mentioned in the definition of λ_h . The reason for that is that we have the axiom $Nq \rightarrow \neg q$ and so the case $h(x) = 1$ and $h(Nx) = 1$ does not arise.
- (b) The crucial points to show are
 - (i) If x is not attacked then $\lambda_h(x) = \text{in}$. This holds because $x \in \Delta_{\mathcal{A}}$.
 - (ii) If zRy and $\lambda_h(z) = \text{in}$ then $h(z) = 1$. Therefore $h(Ny) = 1$ (because $z \rightarrow Ny$ is in the theory) and so $\lambda_h(y) = \text{out}$.
 - (iii) Suppose for all z such that zRy we have $\lambda_h(z) = \text{out}$. We want to show the $\lambda_h(y) = \text{in}$. We have for all z such that zRy that $h(Nz) = 1$ hence $h(\bigwedge_{zRy} Nz) = 1$ and therefore $h(y) = 1$ (because of the theory), and so $\lambda_h(y) = \text{in}$.
 - (iv) Suppose $\lambda_h(y) = \text{und}$. Then $h(y) = h(Ny) = 0$. Let zRy . We cannot have $\lambda_h(z) = \text{in}$ because then $h(z) = 1$ and this implies $h(Ny) = 1$. Thus none of the attackers z of y are in. Thus for all z such that zRy we have $h(z) = 0$. Can we have that for all of such z , $h(Nz) = 1$? If this were the case, since $y \leftrightarrow \bigwedge_{zRy} Nz$ is in $\Delta_{\mathcal{A}}$ we get $h(y) = 1$ i.e. $\lambda_h(y) = \text{in}$, contradicting our assumption that $\lambda_h(y) = \text{und}$. Therefore for some z such that zRy we have $h(Nz) = 0$. But this means that $\lambda_h(z) = \text{und}$. We thus got that if $\lambda_h(y) = \text{und}$ then none of the attackers z of y are in and at least one of them is undecided.

- (v) If $\lambda_h(y)=\text{out}$, (i.e. $h(Ny)=1$), show that for some z such that zRy , we have $\lambda_h(z)=\text{in}$, (i.e. $h(z)=1$). Otherwise for all z such that zRy , we have $h(z)=0$. We ask about any such z , is $h(Nz)=1$? If the answer is yes for all of them then we must have $h(y)=1$, by item (iii) above. So for some z we have $h(Nz)=0$. Then by the axiom $\bigwedge_{z \in \text{Attack}(y)} \neg z \wedge (\bigvee_{z \in \text{Attack}(y)} \neg Nz) \rightarrow \neg y \wedge \neg Ny$ we get that $h(Ny)=0$.

Either way, the assumption that for all z such that zRy we have $h(z)=0$, leads to a contradiction.

Thus λ_h is a legitimate labelling, giving rise to a complete extension.

- (2) Let λ be a legitimate labelling giving rise to a complete extension. We show that it gives a model for $\Delta_{\mathcal{A}}$.

Define h_λ as follows:

$$\begin{aligned} h_\lambda(x) &= 1 \text{ if } \lambda(x) = \text{in} \\ h_\lambda(Nx) &= 1 \text{ if } \lambda(x) = \text{out}. \end{aligned}$$

We show that all axioms of $\Delta_{\mathcal{A}}$ hold

- (a) Show that $h_\lambda \models Nx \rightarrow \neg x$ otherwise $h_\lambda(Nx)=h_\lambda(x)=1$. This means that $\lambda(x)=\text{out}$ and $\lambda(x)=\text{in}$ which is not possible.
- (b) Second we show that if zRy then $h_\lambda \models z \rightarrow Ny$. Otherwise we have $h_\lambda(z)=1$ and $h_\lambda(Ny)=0$. The former implies $\lambda(z)=\text{in}$. Therefore $\lambda(y)=\text{out}$ and so by definition $h_\lambda(Ny)=1$, a contradiction.
- (c) We now show that

$$h_\lambda \models y \leftrightarrow \bigwedge_{zRy} Nz.$$

Assume $h_\lambda(y)=1$. Then $\lambda(y)=\text{in}$. So for all z such that zRy , $\lambda(z)=\text{out}$, and so by definition of h_λ , for all such z , $h_\lambda(Nz)=1$ and hence $h_\lambda(\bigwedge_{zRy} Nz)=1$.

Assume $h_\lambda(y)=0$. Therefore $\lambda(y) \neq \text{in}$. Then either $\lambda(y)=\text{out}$ or $\lambda(y)=\text{und}$. If $\lambda(y)=\text{out}$ then for some z such that zRy we have $\lambda(z)=\text{in}$. Therefore $\lambda(z) \neq \text{out}$ and so $h_\lambda(Nz)=0$. If $\lambda(y)=\text{und}$, then for some z , zRy an $\lambda(z)=\text{und}$ again $\lambda(z) \neq \text{out}$ and so $h_\lambda(Nz)=0$.

Thus for sure if $h_\lambda(y)=0$ then for some z , zRy and $h_\lambda(Nz)=0$ and so $h_\lambda(\bigwedge_{zRy} Nz)=0$.

- (d) We show that $h_\lambda \models \bigwedge_{z \in \text{Attack}(y)} \neg z \wedge (\bigvee_{z \in \text{Attack}(y)} \neg Nz) \rightarrow \neg y \wedge \neg Ny$. If the antecedent holds then all attackers of y are not in and one of them is undecided. Therefore y is undecided and so the consequent holds. ■

COROLLARY 3.3

Let \mathcal{A} be an argumentation network. Consider $\Delta_{\mathcal{A}}$. Then $\Delta_{\mathcal{A}}$ is CN consistent.

PROOF. Since \mathcal{A} has the complete ground extension, by the previous Theorem 3.2 this yields a model for $\Delta_{\mathcal{A}}$. ■

EXAMPLE 3.4

- (1) Consider the argumentation network of Figure 1.

Its theory in CN is $\Delta = \{x \rightarrow Nx, Nx \leftrightarrow x, \neg x \wedge \neg Nx \rightarrow \neg x \wedge \neg Nx\}$. Since we have the N axiom $Nx \rightarrow \neg x$ we get that $x \rightarrow \neg x$ is provable and so $\neg x$ is provable and so $\neg Nx$ is also provable.



FIG. 1.

This means we have only one model h with $h(x) = h(Nx) = 0$ and therefore the only extension is $\lambda(x) = \text{und}$.

- (2) Consider the additional axiom added to the theory Θ_n namely

Stable: $\{x \vee Nx \mid \text{for all } x\}$.

The theory $\Theta_n + \text{stable}$ does not have models h in which $h(x) = h(Nx) = 0$ for any x . Thus this axiom characterises all stable extensions.

THEOREM 3.5

Let $\mathcal{A} = (S, R)$ be an argumentation network. Consider $\Delta_{\mathcal{A}}$ and let $E = \{x \mid \Delta_{\mathcal{A}} \vdash_{\text{CN}} x\}$. Then E is the ground extension of \mathcal{A} .

PROOF. We know that $\Delta_{\mathcal{A}}$ is consistent. Hence E is consistent. We show that E is a complete extension. Since other complete extension E' corresponds to a model of $\Delta_{\mathcal{A}}$, E' contains E . Thus E would be the smallest complete extension — namely E is the ground extension. We now show that E is a complete extension.

- (1) E is conflict free. Let $x, y \in E$. If xRy holds, then $x \rightarrow Ny \in \Delta_{\mathcal{A}}$ and hence $\Delta_{\mathcal{A}} \vdash_{\text{CN}} x \rightarrow \neg y$, which contradicts the consistency of $\Delta_{\mathcal{A}}$ since $\Delta_{\mathcal{A}} \vdash_{\text{CN}} y$.
- (2) Assume E protects x . We show $x \in E$. Let zRx . then for some $y \in E$, yRz holds. Hence $y \rightarrow Nz$ is in $\Delta_{\mathcal{A}}$ and so $\Delta_{\mathcal{A}} \vdash_{\text{CN}} Nz$. Thus we have

$$\Delta_{\mathcal{A}} \vdash_{\text{CN}} \bigwedge_{zRx} Nz$$

and hence $\Delta_{\mathcal{A}} \vdash_{\text{CN}} x$ and so $x \in E$. ■

REMARK 3.6

This is a clarifying remark about the correspondence between extensions of an argumentation network $\mathcal{A} = (S, R)$ and **CN** models of the theory $\Delta_{\mathcal{A}}$.

Consider the three argumentation networks $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ in Figure 2. They all have the same extension, $x = \text{in}$, $y = \text{out}$, $z = \text{in}$.

Their theories $\Delta_{\mathcal{A}_i}$ are different, but they have the same models.

$$\begin{aligned} \Delta_{\mathcal{A}_1} &= \{x, x \rightarrow Ny, y \leftrightarrow Nx, z, \neg x \wedge \neg Nx \rightarrow \neg y \wedge \neg Ny\} \\ \Delta_{\mathcal{A}_2} &= \{x, x \rightarrow Ny, y \leftrightarrow Nx, y \rightarrow Nz, z \leftrightarrow Ny, \\ &\quad \neg x \wedge \neg Nx \rightarrow \neg y \wedge \neg Ny, \neg y \wedge \neg Ny \rightarrow \neg z \wedge \neg Nz\} \\ \Delta_{\mathcal{A}_3} &= \{x, x \rightarrow Ny, y \leftrightarrow Nx \wedge Nz, x \vee z \rightarrow Ny, \neg x \wedge \neg z \wedge \\ &\quad (\neg Nx \vee \neg Nz) \rightarrow \neg y \wedge \neg Ny, y \rightarrow Nz, z \rightarrow Ny, \\ &\quad \neg y \wedge \neg Ny \rightarrow \neg z \wedge \neg Nz, \neg z \wedge \neg Nz \rightarrow \neg y \wedge \neg Ny, \\ &\quad y \leftrightarrow Nz, z \leftrightarrow Ny\} \end{aligned}$$

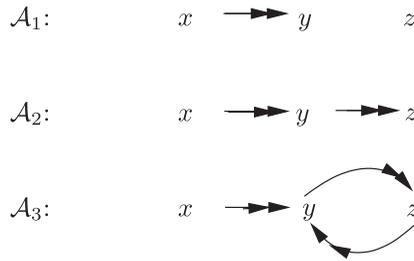


FIG. 2.

All three theories have only one model h

$$x = 1, Nx = 0, y = 0, Ny = 1, z = 1, Nz = 0.$$

The moral of the example is that the theories describe the extensions and not the geometry of the networks. In fact, $\Delta_{\mathcal{A}_i}$ are all equivalent to $\Delta_{\mathcal{A}} = \{x, \neg Nx, \neg y, Ny, z, \neg Nz\}$.

Compare the CN approach with the truly meta-level approach of [17, Section 5], discussed below in the beginning of Section 5.

We describe (S, R) in predicate logic with unary predicates $Q_1(x) = x$ is in, $Q_0(x) = x$ is out, and $Q_7(x) = x$ is und, and a relation xRy for attack. Thus using R the networks of Figure 2 can be distinguished in the semantics. Note, however, that we can read the geometry of the networks from the syntax of the theories $\Delta_{\mathcal{A}_i}$, $i = 1, 2, 3$, but not from their models! This highlights the importance of proof theory.

REMARK 3.7

The previous correspondence theorem reduces the idea of attack in argumentation networks to the idea of strong negation in classical propositional logic. This reduction simplifies every move we make in the argumentation area and gives us a tremendous advantage in making available to us all the machinery of classical logic. This includes

- (1) extensions for formal argumentations such as joint attacks, support, higher level attacks, probabilistic argumentation, predicate/modal logic argumentation and more, all can be done more simply and easily using strong negation, see the following sections;
- (2) applications of argumentation become application of classical logic;
- (3) new ideas can be more readily imported from classical logic into argumentation;
- (4) the Caminada labels, x is in, x is out and x is undecided are available in the object language as x is true, Nx is true and $\neg x \wedge \neg Nx$ is true, respectively.
- (5) we now need to ask ourselves: what is the added value of argumentation over classical logic? We need a clear and detailed answer to this question.

4 Intermediate critical evaluation

This section pauses our formal development to evaluate what we have so far and to explain to the reader where we are going. We shall do this by a list of critical comments.

CC1. Basing argumentation on the unary notion of ‘being attacked’

We read Nx as ‘ x is being attacked’, we are not saying how and from where this attack comes. This makes Nx a kind of strong negation (with axiom $Nx \rightarrow \neg x$), see [40]. This single simple idea allows us to have Theorem 3.2. It also allows us to go in the direction of turning the system CN into a paraconsistent logic of negation (see Wikipedia) by adding axioms on iterating N (e.g. the axiom $NNx \leftrightarrow x$). We can do this safely for as long as Theorem 3.2 is retained. We shall address this direction in later sections. We believe we can achieve similar results for logic programming by reading Nx as ‘ x fails’. This direction, and the connection with answer set programming, is left for a subsequent paper.

There is another direction we can go in. We can use another meaningful logic such as intuitionistic logic, linear logic or relevance logic or fuzzy logic to replace classical logic and thus get argumentation theory in those contexts. Again we shall look at this in a subsequent paper.

CC2. Simple way of defining joint attacks

The theory $\Delta_{\mathcal{A}}$ written for an argumentation network $\mathcal{A} = (S, R)$ is composed of several components.

- (1) The logic CN (the use of N).
- (2) Formulas defining when an argument x is ‘in’.

This is the part

$$(\mathbf{F}_{\text{in}}) \quad x \leftrightarrow \bigwedge_{zRx} Nz$$

\mathbf{F}_{in} also includes the part relating to ‘ x is not attacked’, since the empty conjunction is \top .

- (3) Formulas defining when an argument x is ‘out’. This is the part $z \rightarrow Nx$, for all zRx , or if we write it as a single formula, it is \mathbf{F}_{out} .

$$(\mathbf{F}_{\text{out}}) \quad \bigvee_{zRx} z \rightarrow Nx.$$

- (4) Formulas defining when x is undecided

$$(\mathbf{F}_{\text{und}}) \quad \left(\bigwedge_{z \in \text{Attack}(x)} \neg z \right) \wedge \left(\bigvee_{z \in \text{Attack}(x)} \neg Nz \right) \rightarrow \neg x \wedge \neg Nx.$$

Once we put the above in $\Delta_{\mathcal{A}}$ we get the correspondence theorem, Theorem 3.2, generating complete extensions by models of $\Delta_{\mathcal{A}}$.

Now we can see how easy it is to generalize argumentation to joint attacks. Joint attacks, introduced in [28, 44] can be described by the configuration in Figure 3 (using Gabbay’s notation from [28]).

The meaning is that we have x is out if all of z_1, \dots, z_n are in. This we can write in our logic CN simply as $\bigwedge_{(z_1, \dots, z_n)R_0x} z_i \rightarrow Nx$ where R_0 is the joint attack relation of the form $R_0 \subseteq (2^S - \emptyset) \times S$. So we can write the formula $\mathbf{F}_{\text{out}}^{\text{Joint}}$ for joint attacks as

$$(\mathbf{F}_{\text{out}}^{\text{Joint}}) \quad \left(\bigvee_{GR_0x} \bigwedge_{z \in G} z \right) \rightarrow Nx,$$

where $G \subseteq S, x \in S$.

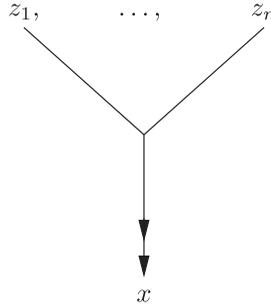


FIG. 3.

The formula for $\mathbf{F}_{\text{out}}^{\text{Joint}}$ for a point x will be

$$(\mathbf{F}_{\text{in}}^{\text{Joint}}) \quad x \leftrightarrow \left(\bigwedge_{GR_0x} \bigvee_{z \in G} Nz \right)$$

$$(\mathbf{f}_{\text{und}}^{\text{Joint}}) \quad \bigwedge_{GR_0x} \left(\bigvee_{z \in G} \neg z \right) \wedge \left(\bigvee_{GR_0xz \in G} \bigwedge (z \vee \neg Nz) \right) \rightarrow \neg x \wedge \neg Nx,$$

If we use these three wffs $\mathbf{F}_{\text{in}}^{\text{Joint}}$ and $\mathbf{F}_{\text{out}}^{\text{Joint}}$ and $\mathbf{F}_{\text{und}}^{\text{Joint}}$ to define $\Delta_{\mathcal{A}}$, we can study the semantics for joint attacks, in the object level, in classical logic as models of $\Delta_{\mathcal{A}}$, provided we prove a correspondence theorem similar to Theorem 3.2.

The reader can compare the simplicity of this approach to what is done in the papers [28, 44].

CC3. Simple way of defining higher level attacks

Higher level attacks on attacks. These were studied in many papers [5, 6, 29, 39, 43]. They were first introduced in general in Gabbay's paper [8]. Figure 4 illustrates the basic configuration for higher level attacks.

z attacks x and y attacks the attack $z \rightarrow x$. We write it as $y \rightarrow (z \rightarrow x)$.

We need to write the $\mathbf{F}_{\text{in}}^{\text{higher}}$ and $\mathbf{F}_{\text{out}}^{\text{higher}}$ and $\mathbf{F}_{\text{und}}^{\text{higher}}$ of this type of attack. In our set up with N it is easy to write this! The $\mathbf{F}_{\text{out}}^{\text{higher}}$ is $z \wedge Ny \rightarrow Nx$ and the $\mathbf{F}_{\text{in}}^{\text{higher}}$ involves $Nz \vee y$. Let us write the wff's in detail for a network with one level of higher attacks. So our networks have the form (S, R, R_1) , where $R \subseteq S \times S$ are the attacks and $R_1 \subseteq S \times R$ are the attacks on the attacks. We write:

$$(\mathbf{F}_{\text{out}}^{\text{higher}}) \quad \left(\bigvee_{zRx} \bigwedge_{yR_1(z,x)} z \wedge Ny \right) \rightarrow Nx$$

$$(\mathbf{F}_{\text{in}}^{\text{higher}}) \quad \bigwedge_{zRx} \bigvee_{yR_1(z,x)} (Nz \vee y) \leftrightarrow x$$

$$(\mathbf{F}_{\text{und}}^{\text{higher}}) \quad \left(\neg \bigwedge_{zRx} \bigvee_{yR_1(z,x)} (Nz \vee y) \right) \wedge \neg \bigvee_{zRx} \bigwedge_{yR_1(z,x)} Ny \rightarrow \neg x \wedge \neg Nx.$$

Again we use $\mathbf{F}_{\text{in}}^{\text{high}}$, $\mathbf{F}_{\text{out}}^{\text{high}}$ and $\mathbf{F}_{\text{und}}^{\text{high}}$ to define $\Delta_{\mathcal{A}}$, and let the semantics be all CN models of $\Delta_{\mathcal{A}}$. We need to prove a correspondence theorem similar to Theorem 3.2.

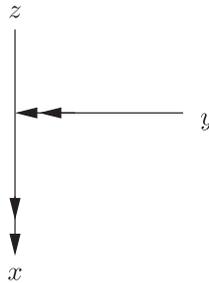


FIG. 4.

CC4. A word of caution

Although we are showing how other types of networks can be translated into CN, we are not just saying ‘look, our paper is introducing you to another master generalization of argumentation’. We reserve judgement about N until the end of this paper and we might say at the end to the reader ‘in view of our article, do not think any more in the old conceptual framework of argumentation of (S, R) but think in terms of strong negation N of just being attacked, and do your argumentation from now on in classical logic with N ’. The reason we reserve judgement is because we want to figure out first how the use of N affects related systems such as ABA (Assumption Based Argumentation, see [23]) and ASPIC (see [46]). In ASPIC and ABA, the game is to start with a logic theory Δ , define proofs from Δ as the argument set S , define a respective suitable attack R between proofs, then define (S, R) as an instantiated network, take extensions $E \subseteq S$ and then make sure that E is consistent in the logic of Δ . This we perceive as a potentially unnecessary external detour. From the point of view of our method, we might say to a proponent of ABA and ASPIC, “why do you need all these roundabout ways involving a multitude of problems? Why not add strong negation N directly to the logic of Δ and model your argumentation there and you are done?” If CN can swing this and we succeed in working out the details, then ABA and ASPIC would immensely benefit from our conceptual view. See CC7 below.

CC5. Comparing with abstract dialectical framework (ADF)

ADF were introduced in [16] by Professor G. Brewka as a generalization of argumentation frameworks and immediately caused both excitement and criticism. In this article, we shall use an example from Brewka’s slides [14] to do our comparison.

An ADF has the form (S, R, C) where S is a set of arguments and $R \subseteq S \times S$ is the link relation (Brewka calls them ‘links’ because he does not view them as attacks). C is a family of acceptance relations. For each $x \in S$, there is a formula of classical propositional logic φ_x , specifying the acceptance (‘in’) condition for x , based on the acceptance values of $\{y \mid yRx\}$.

In our terms, as well as in classical logic terms, what ADF is saying is condition \mathbf{F}_{in}^C :

$$(\mathbf{F}_{\text{in}}^C) \quad x \leftrightarrow \varphi_x, \text{ for all } x \in S.$$

If we regard \mathbf{F}_{in}^C as a condition in our logic CN, then we can define all complete extensions in our sense as all extensions obtained from models \mathbf{m} of $\Delta_C = (\Theta_n + \mathbf{F}_{\text{in}}^C)$, where the values {in, out, und} are as in the proof of part 1 of Theorem 3.2, namely:

- $x = \text{in}$ if $\mathbf{m}(x) = 1$
- $x = \text{out}$ if $\mathbf{m}(Nx) = 1$
- $x = \text{und}$ if $\mathbf{m}(x) = \mathbf{m}(Nx) = 0$.

Brewka, however, works only in classical proposition logic with the three valued semantics according to Kleene. So his models give 3 values $\{1, 0, u\}$ with

$$\begin{aligned} 1 \wedge 1 &= 1 \\ 0 \wedge 0 &= 0 \\ 1 \wedge u &= 0 \wedge u = u \wedge u = u. \end{aligned}$$

Brewka derives his extensions for the 3 valued models using some process. See [14, slides 13–18]. We now reproduce in Figure 5 the original slide 18 of Brewka [14] in order to compare ADF with our CN approach.

The Brewka extensions are given in the figure.

Figure 6 lists the models in CN obtained for the ADF theory.

We note that our model \mathbf{m}_1 is the same as Brewka's v_1 and our model \mathbf{m}_4 is the same as Brewka's v_2 . However, model \mathbf{m}_3 gives $a = \text{in}, b = \text{in}, c = \text{in}$ and $d = \text{und}$, and model \mathbf{m}_2 gives $b = c = \text{und}$ and $a = d = \text{in}$. We do not get the Brewka's grounded model $G = \{a = \text{in}, b = c = d = \text{und}\}$.

This model is not one of the model of the theory

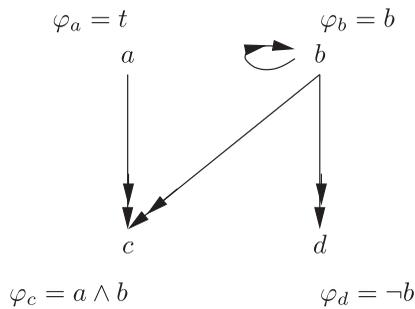
$$\Delta_C = \{x \leftrightarrow \varphi_x \mid x \text{ a node in Figure 5}\} \cup \Theta_n.$$

So how does Brewka get his grounded extension? Look at v_1 and v_2 , a gets 1 in both, but b, c and d get value 1 in one of them and 0 in the other. So if we look at G with the undecided value u given to b, c and d for each of these arguments there are extensions which make its value 1 and there are extensions which make its value 0. Therefore their value, according to ADF is undecided.

From our point of view, this way of looking at undecided is just external combinatorics, devoid of conceptual content. According to our view only $\neg x \wedge \neg \mathcal{N}x$ makes x undecided, namely x is false but not strongly false.

G is not an extension. It is not a model of Δ_C because of the axiom $d = \neg b$.

Obviously we could try and add restrictions on the models to get the same results as Brewka (i.e. implement ADF in CN) but why should we do that at all? Our methodology is sound and stronger.



- models:
 - $v_1 = \{a \mapsto \mathbf{t}, b \mapsto \mathbf{t}, c \mapsto \mathbf{t}, d \mapsto \mathbf{f}\}$ corresponds to $\{a, b, c, \neg d\}$
 - $v_2 = \{a \mapsto \mathbf{t}, b \mapsto \mathbf{f}, c \mapsto \mathbf{f}, d \mapsto \mathbf{t}\}$ corresponds to $\{a, \neg b, \neg c, d\}$
- grounded model: $v_1 = \{a \mapsto \mathbf{t}, b \mapsto \mathbf{u}, c \mapsto \mathbf{u}, d \mapsto \mathbf{u}\}$ corresponds to $\{a\}$

FIG. 5.

12 *The attack as strong negation*

model	$a = \top$	Na	$b = b$	Nb	$c = a \wedge b$	Nc	$d = \neg b$	Nd
m_1	1	0	1	0	1	0	0	1
m_2	1	0	0	0	0	0	1	0
m_3	1	0	1	0	1	0	0	0
m_4	1	0	0	1	0	1	1	0

FIG. 6.

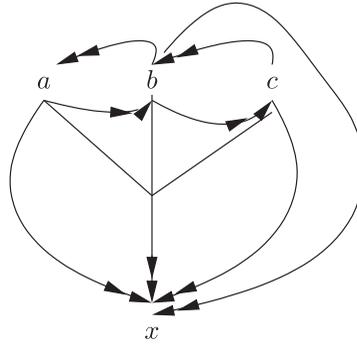


FIG. 7.

Given that we had N in the object language, we can do more than ADF. ADF writes the acceptance conditions in 2-valued classical logic and brings in the undecided value only through the semantic interpretation. We have the undecided value $(\neg x \wedge \neg Nx)$ in the language itself and therefore we can put the undecided property into the acceptance conditions. Consider the joint attack described in Figure 7.

Suppose all three attackers a, b, c are undecided. In this case, we traditionally say that x is also undecided because we do not know, maybe all three $a=b=c = \text{in}$. This could happen. If, however, we adopt the new view that the chance that all three attackers are ‘in’ can be disregarded, then we want to say it in our acceptance conditions.

So we want to say

$$(\mathbf{F}_{\text{in}}(x)) \quad x \leftrightarrow (Na \wedge Nb \wedge Nc) \bigvee (\neg a \wedge \neg Na \wedge \neg b \wedge \neg Nb \wedge \neg c \wedge \neg Nc)$$

$$(\mathbf{F}_{\text{out}}(x)) \quad a \vee b \vee c \rightarrow Nx$$

$$(\mathbf{F}_{\text{in}}(a, b, c)) \quad \begin{aligned} a &\leftrightarrow Nb \\ b &\leftrightarrow Na \wedge Nc \\ c &\leftrightarrow Nb \end{aligned}$$

$$(\mathbf{F}_{\text{out}}(a, b, c)). \quad \begin{aligned} a \vee c &\rightarrow Nb \\ b &\rightarrow Na \wedge Nc. \end{aligned}$$

Can ADF express exactly the same extensions as what CN gets for the above, especially the one $a=b=c = \text{und}$, $x = \text{in}$?

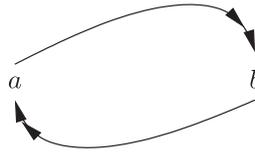


FIG. 8.

CC6. Using CN for the probabilistic approach

There are many papers on probabilistic argumentation. Our paper [37] offers a comprehensive approach based on probability models of classical logic. We shall therefore compare the probabilistic use of CN with the approach in [37]. We shall use the network of Figure 8 as an example.

The approach of [37] regards $\{a, b\}$ as atoms of classical propositional logic. The logic with these atoms has the following models

$$\begin{aligned} \mathbf{m}_1 &: a \wedge b; x \\ \mathbf{m}_2 &: a \wedge \neg b; y \\ \mathbf{m}_3 &: \neg a \wedge b; z \\ \mathbf{m}_4 &: \neg a \wedge \neg b; 1 - x - y - z. \end{aligned}$$

We can give a probability distribution P on the models

$$\begin{aligned} P(\mathbf{m}_1) &= x \\ P(\mathbf{m}_2) &= y \\ P(\mathbf{m}_3) &= z \\ P(\mathbf{m}_4) &= 1 - x - y - z. \end{aligned}$$

The fact that these nodes $\{a, b\}$ are part of the network (S, R) of Figure 8 is reflected in the probability having to satisfy the equational approach equation called E3 in [37] (see [27] for the Equational Approach to argumentation).

$$(E3) \quad P(x) = P\left(\bigwedge_{yRx} \neg y\right).$$

In our case this means

- $P(a) = P(\neg b)$
- $P(b) = P(\neg a)$

where

- $P(x) = \sum_{\mathbf{m} \models x} P(\mathbf{m})$

Therefore the equations we get are

- $x + y = 1 - x - z$
- $x + z = 1 - x - y$

14 *The attack as strong negation*

These two are the same equation, yielding

$$1 - x - y - z = P(\mathbf{m}_4) = x.$$

Therefore any probability distribution P of the form

$$\begin{aligned} P(a \wedge b) &= x \\ P(a \wedge \neg b) &= y \\ P(\neg a \wedge b) &= z \\ P(\neg a \wedge \neg b) &= x \end{aligned}$$

with $2x + y + z = 1$ is a good one, respecting the attack relation.

We have $P(a) = x + y$, $P(b) = x + z$, $P(\neg a \wedge \neg b) = x$.

Let us now check how the models of **CN** relate to probability. For the language with atoms $\{a, Na, b, Nb\}$ satisfying the **CN** axiom

$$Nx \rightarrow \neg x$$

we can have the following models $h_{i,j} = \alpha_i \wedge \beta_j$, $i = 1, 2, 3$, $j = 1, 2, 3$ where

$$\begin{aligned} \alpha_1 &= a \wedge \neg Na \\ \alpha_2 &= \neg a \wedge \neg Na \\ \alpha_3 &= \neg a \wedge Na \\ \beta_1 &= b \wedge \neg Nb \\ \beta_2 &= \neg b \wedge \neg Nb \\ \beta_3 &= \neg b \wedge Nb. \end{aligned}$$

Here we have 9 models as opposed to 6 models of the previous approach. Let the probabilities on these models be

$$\pi(\alpha_i \wedge \beta_j) = P_{i,j}$$

with

$$\sum_{i,j} P_{i,j} = 1.$$

The models must satisfy the theory $\Delta_{\mathcal{A}}$ for the network of Figure 8. These are

$$\begin{aligned} Na &\rightarrow \neg a \\ Nb &\rightarrow \neg b \\ a &\leftrightarrow Nb \\ b &\leftrightarrow Na \\ \neg a \wedge \neg Na &\rightarrow \neg b \wedge \neg Nb \\ \neg b \wedge \neg Nb &\rightarrow \neg a \wedge \neg Na. \end{aligned}$$

So only 3 models are models of $\Delta_{\mathcal{A}}$. These are

$$\begin{aligned} h_1 &= a \wedge \neg Na \wedge Nb \wedge \neg b \\ h_2 &= \neg a \wedge Na \wedge b \wedge \neg Nb \\ h_3 &= \neg a \wedge \neg Na \wedge \neg b \wedge \neg Nb. \end{aligned}$$

Let the relative probabilities be p_1, p_2, p_3 respectively with $\sum p_i = 1$.

Note that h_1 is the extension $a = \text{in}, b = \text{out}$ h_2 is the extension $b = \text{in}, a = \text{out}$ and h_3 is the extension $a = b = \text{und}$.

The first difference between the approaches is that while the approach in [37] gives probability to arguments (called in [37] ‘the internal approach’), the CN approach ends up giving probabilities to extensions (called in [37] ‘the external approach’).

Giving probability to extensions is not new. It is supported by many authors (see references/discussion in [37]). Let us calculate the probabilities $\pi(a)$ and $\pi(b)$. We get

$$\pi(a) = p_1, \pi(b) = p_2, \pi(\neg a \wedge \neg b) = p_3.$$

Comparing the two approaches, it makes sense to equate

$$p_3 = x, p_2 = x + z, p_1 = x + y.$$

The main difference is that we are giving probabilities to 2-valued models in [37] and using CN we are using 3-valued models.

In both cases we give probabilities to models. However, in case of CN, the models are extensions and so we are giving probabilities to extensions also.

CC7. CN and bipolar networks

Note that in CN we get support and contrary arguments for free. Since we have implication ‘ \rightarrow ’ in the logic, we can write ‘ $x \rightarrow y$ ’ to mean ‘ x supports y ’ and since we have negation ‘ \neg ’ in the logic, we can view $\neg x$ as the contrary of x . We need not introduce additional atoms into the argumentation network for contraries, nor do we have to introduce an additional arrow for support into the network. Furthermore, since we have negation, we have the additional option to represent ‘ x supports y ’ as ‘ $x \twoheadrightarrow \neg y$ ’ namely as $x \rightarrow N\neg y$.

Let us do this in a systematic manner.

DEFINITION 4.1

- (1) A bipolar network \mathcal{B} has the form $\mathcal{B} = (S, R_a, R_s)$, where $R_a \subseteq S \times S$ is the attack relation (also denoted by \twoheadrightarrow) and $R_s \subseteq S \times S$ is the support relation (also denoted by \Rightarrow).
- (2) Given a bipolar network \mathcal{B} , we offer two possible translations into CN.
 - $\tau_1(x \twoheadrightarrow y) = x \rightarrow Ny$
 - $\tau_1(x \Rightarrow y) = x \rightarrow y$
 - $\tau_2(x \twoheadrightarrow y) = x \rightarrow Ny$
 - $\tau_2(x \Rightarrow y) = x \rightarrow N\neg y$
- (3) Note that the complete extensions of the bipolar networks will be obtained from all the models of CN.

REMARK 4.2

- (1) We note that $x \rightarrow y$ implies $x \rightarrow N\neg y$. This holds because, as we shall see in Section 5, $\neg Ny \rightarrow N\neg y$ and because $y \rightarrow N\neg y$ both hold. Thus τ_1 is stronger than τ_2 .
- (2) When we have a translation τ from one system (e.g. \mathcal{B}) into another (e.g. CN), we need to examine the properties of *soundness* and *completeness*. Soundness in our case means that whatever we consider as a bipolar complete extension for \mathcal{B} will turn out to be a complete extension according to CN. Completeness means that CN does not give any additional complete extensions.

The problem in this case is that there is disagreement in the community about how to define the complete extensions for \mathcal{B} . The main papers of C. Cayrol and M. C. Lagasque-Schiex are [19, 20]. These have been criticised by G. Brewka and S. Woltran [16] and a solution was proposed in our paper [47]. Thus a detailed analysis of soundness and completeness for our translations must be postponed to a continuation paper [35]. However, we can point out in this article some key properties involved in [47].

The possible properties are as follows:

- (P1) $\frac{x \Rightarrow y, z \rightarrow y}{z \rightarrow x}$
 (P2) $\frac{x \Rightarrow y, y \rightarrow u}{x \rightarrow u}$
 (P3) $\frac{w \rightarrow x, x \Rightarrow y}{w \rightarrow y}$

We need to check these properties for both

$$\tau(x \rightarrow y) = x \rightarrow y$$

and

$$\tau(x \Rightarrow y) = x \rightarrow N\neg y.$$

What we need is consistency. Can we add the translation of these rules to CN and remain consistent?

The answer is positive. So we can hope to have something like the following theorem:

THEOREM 4.3

Let \mathbb{P} be a set of properties for a bipolar network \mathcal{B} (e.g. (P1)–(P3) of Remark 4.2). Let τ be a translation of \mathcal{B} into CN, and let $\tau(\mathbb{P})$ be the translation of \mathbb{P} into CN. Then the translation τ of \mathcal{B} into $\text{CN} + \tau(\mathbb{P})$ is sound and complete.

As we said, we shall address this theorem in Part 2 of this article.

CC8. Limitations of the CN approach

The CN approach transforms the geometrical representation of an argumentation network $\mathcal{A} = (S, R)$ into a theory $\Delta_{\mathcal{A}}$ of the logic CN. Theorem 3.2 ensures that the correspondence between complete extensions of \mathcal{A} and models of $\Delta_{\mathcal{A}}$ is sound and complete. We are trading off here, however, geometry for model theory.

The previous CC1–CC4 discussed the advantages of the CN approach. The limitations come from the fact that in the CN approach we obscure/lose the geometrical aspects of \mathcal{A} . Therefore, any moves in argumentation theory which use the geometry (e.g. the strongly connected components, SCC of Baroni *et al.* [7]) will become less transparent. We can mathematically do them in CN, but we would have to extract the geometry of $\mathcal{A} = (S, R)$ back out of $\Delta_{\mathcal{A}}$!

The following Figure 9 can be used to illustrate our point using the CF2 semantics [7, 32].

The CN semantics gives the complete extension of all undecided to the network \mathcal{A} of Figure 9 in agreement, of course, with the traditional Dung approach. The CF2 semantics takes maximal conflict free subsets of the top SCC and therefore yields the extensions $\{a, d\}, \{b, d\}, \{c, e\}$. CF2 relies on identifying the SCCs. It relies on the geometry of (S, R) .

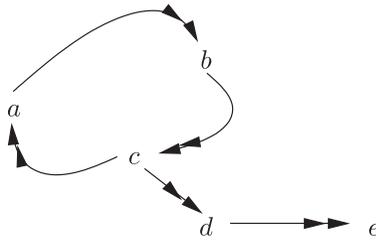


FIG. 9.

The CN theory for this network \mathcal{A} is

$$\Delta_{\mathcal{A}} = \Theta_{\mathbf{n}} \cup \{a \rightarrow Nb, b \rightarrow Nc, c \rightarrow Na, c \rightarrow Nd, d \rightarrow Ne, e \leftrightarrow Nd, d \leftrightarrow Nc, c \leftrightarrow Nb, b \leftrightarrow Na, a \leftrightarrow Nc\}$$

Looking at $\Delta_{\mathcal{A}}$, we have to define/extract the cycles in order to define the CF2 semantics for \mathcal{A} . Gabbay’s approach using annihilators [7] also requires the use of geometry but to a lesser extent. We can identify syntactically a cycle, say

$$x_1 \rightarrow Nx_2, x_2 \rightarrow Nx_3, \dots, x_n \rightarrow Nx_1$$

and break the cycle by applying an annihilator, say to annihilate the point x_i , i.e. add a new point $z(x_i)$ to $\Delta_{\mathcal{A}}$ with $z \rightarrow Nx_i$, and look instead at

$$\Delta_{\mathcal{A}}^{z(x_i)} = \Delta_{\mathcal{A}} \cup \{z(x_i), z(x_i) \rightarrow Nx_i\}.$$

We still need some geometric intuition in doing this.

5 Introducing the logic CNN

5.1 A meta-level object level short discussion

The perceptive reader may be aware of Section 5 of my 2009 paper with Caminada [17]. In this article, we describe several options of looking at an abstract argumentation network (S, R) . Since 2009, there were many other papers, which put forward different representations of Abstract Argumentation Networks in terms of well-known logics, see e.g. [21]–[38]. We shall compare and discuss these papers in our Comparison with the Literature Section 9. Meanwhile in this section, we want to make a critical point about the difference between Object Level Vs Meta-Level representation of Abstract Argumentation Networks, and so we consider now one of the possible options of Section 5 of [17]. This option is to describe (S, R) completely in classical predicate logic. We consider S as the domain of the logic and we consider the attack relation R as a binary relation on S . In addition to that, we need 3 unary predicates, Q_1, Q_0 an Q_2 , representing the 3 Caminada labels for the elements of S , namely x is ‘in’, x is ‘out’ and x is ‘undecided’, respectively.

We write axioms in predicate logic basically expressing the properties of the labelling relative to R making it a legitimate labelling. These are the following:

Consider the following classical theory $\Delta(R, Q_0, Q_1, Q_2)$.

$$(1) \quad \forall x(Q_0(x) \vee Q_1(x) \vee Q_2(x))$$

- (2) $\neg\exists x(Q_i(x) \wedge Q_j(x))$ for $i \neq j, i, j \in \{0, 1, ?\}$
- (3) $\forall y(\forall x(xRy \rightarrow Q_0(x)) \rightarrow Q_1(y))$
- (4) $\forall y(\exists x(xRy \wedge Q_1(x)) \rightarrow Q_0(y))$
- (5) $\forall y(\forall x(xRy \rightarrow (Q_0(x) \vee Q_?(x))) \wedge \exists x(xRy \wedge Q_?(x)) \rightarrow Q_?(y))$

Any model of Δ with domain D defines an argumentation framework with the set of argument $S = D$, and the attack relation is R and the labelling λ_Δ is what we obtain from the elements satisfying the respective predicates Q_0, Q_1 and $Q_?$. Notice that we are not using '='.

If we want to characterize any specific argumentation framework $\mathcal{A} = (S, R)$ we need equality and we need constant names for every element of S . We write the following additional axioms \mathcal{A}

- (6) $\forall x(\bigvee_{a \in S} x = a)$
- (7) $\bigwedge_{a, b \in S, a \neq b} a \neq b$
- (8) $\bigwedge_{a, b \in R} aRb$

The use of predicate logic to talk about (S, R) is meta-level. The predicates Q_1, Q_0 and $Q_?$ are not logical connectives. We cannot write, e.g. the expression $Q_0(Q_0(x))$. In contrast, the logic CN is object level. It can express the predicates Q_1, Q_0 and $Q_?$ in the object level, as well as the relation R , as follows (see, however, Remark 3.6 of Section 3):

$$\begin{aligned} Q_1(x) &= \text{def } x \\ Q_0(x) &= \text{def } Nx \\ Q_?(x) &= \text{def } \neg x \wedge \neg Nx \\ xRy &= \text{def } x \rightarrow Ny. \end{aligned}$$

Note the difference between object level and meta-level. Suppose we want to instantiate (S, R) with arguments which are formulas of predicate logic, say we have xRy and we instantiate $x = \alpha$ and $y = \beta$. In the meta-level language we cannot write $\alpha R \beta$. Even if we allow for the use of names ' α ' and ' β ' and add appropriate axioms for the correct handling of names, we still do not know what ' α ' R ' β ' means in terms of semantics. The meta-level translation does not give any meaning to R it is just a translation. On the other hand, we can write in our object level system the formula $\alpha \rightarrow N\beta$, and if we use predicate logic with N , we can let the models of this predicate logic with N provide a proposed semantics for the instantiation by using Theorem 3.2 as a definition. Furthermore we can write additional axioms for N . For example we can also write NNx and add an axiom (which is valid)

$$\text{NN:} \quad x \leftrightarrow NNx$$

On the other hand, the meta-level predicate approach can deal with infinite non-finitary networks, while our object level approach requires the network to be finitary. We shall discuss the predicate approach further in the context of comparing with paper [25] in Section 9.

5.2 The logics CN and CNN

Having explained the object level nature of our translation of argumentation into CN, let us now focus on this logic.

So what kind of a logic is CN? Can we say more about it, in addition to what we said in Section 1? The answer is yes. We can add a modal point of view to that of Section 1.

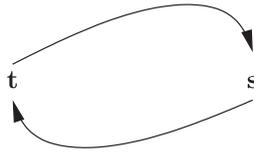


FIG. 10.

Consider a modal logic with two possible worlds \mathbf{t} and \mathbf{s} . Assume that \mathbf{s} is accessible to \mathbf{t} and \mathbf{t} is accessible to \mathbf{s} . So we have a symmetric non-reflexive relation. See Figure 10.

We further require on assignments to atoms that if the atom q is assigned \top at \mathbf{t} , then it is assigned \top at \mathbf{s} (but not necessarily vice versa).

Let \mathbf{t} be the actual world and let us have the modal connective N with the following truth table

- $\mathbf{t} \models NA$ iff $\mathbf{s} \models \neg A$
- $\mathbf{s} \models NA$ iff $\mathbf{t} \models \neg A$
- $\models A$ iff $\mathbf{t} \models A$ (i.e. \mathbf{t} is the actual world)

Let us call this modal logic **CNN**.

We therefore have the following true in the actual world \mathbf{t}

- **CNN** $\models Nq \rightarrow \neg q, q$ atomic
- **CNN** $\models NNA \leftrightarrow A$
- **CNN** $\models N(A \wedge B) \leftrightarrow NA \vee NB$
- **CNN** $\models N(A \vee B) \leftrightarrow NA \wedge NB$
- **CNN** $\models N(N\neg q \rightarrow q)$
- **CNN** $\models (A \rightarrow B) \rightarrow N(NB \rightarrow NA)$

In this context for atomic arguments q we understand the following in \mathbf{t} :

- (*1) q attacks p , reads $q \rightarrow Np$
- (*2) q is out, reads Nq
- (*3) q is in, reads q
- (*4) q is undecided, reads $\neg q \wedge \neg Nq$.

DEFINITION 5.1

Let **CNN** be the logic extending **CN** with the following axioms

- NN**: $NNx \leftrightarrow x$
- N \wedge** : $N(x \wedge y) \leftrightarrow Nx \vee Ny$
- N \vee** : $N(x \vee y) \leftrightarrow Nx \wedge Ny$
- N \leftrightarrow** : $x \leftrightarrow y$ implies $Nx \leftrightarrow Ny$, where IN does not occur in x or y ²
- N \neg** : $N\neg x \leftrightarrow \neg Nx$

EXAMPLE 5.2

Consider the network of Figure 11

The Figure describes a network containing the points x, y, z where x is the only attacker of y and y is the only attacker of z . Given that, we know that no matter what complete extension we are dealing

²Note that we have $\top \leftrightarrow (Nx \rightarrow \neg x)$ but not $N\top \leftrightarrow N(Nx \rightarrow \neg x)$.

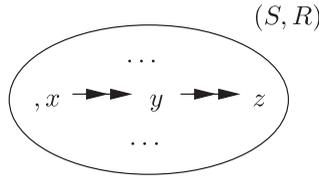


FIG. 11.

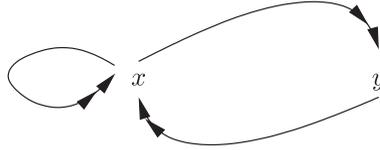


FIG. 12.

with, the value of x must be equal to the value of z .

- $x = \text{in} \Rightarrow y = \text{out} \Rightarrow z = \text{in}$
- $x = \text{out} \Rightarrow y = \text{in} \Rightarrow z = \text{out}$
- $x = \text{und} \Rightarrow y = \text{und} \Rightarrow z = \text{und}$

If we look at \mathbf{F}_{in} , \mathbf{F}_{out} and \mathbf{F}_{und} applied to these points, we see that the formulas

- (1) $x \rightarrow Ny$
- (2) $y \rightarrow Nz$
- (3) $y \leftrightarrow Nx$
- (4) $z \leftrightarrow Ny$
- (5) $\neg x \wedge \neg Nx \rightarrow \neg y \wedge \neg Ny$
- (6) $\neg y \wedge \neg Ny \rightarrow \neg z \wedge \neg Nz$

must hold in the logic. We also have, of course, the axiom schema $Nu \rightarrow \neg u$.

These formulas must prove $x \leftrightarrow z$ and $Nx \leftrightarrow Nz$. We now proceed to prove them:

- (a) Assume x , then from 1. and 4. we get z .
- (b) Assume z , then from 4. we get Ny and from the axiom we get $\neg y$ and so from 3. we get $\neg Nx$. If we had $\neg x$ then from 5. and 6. we would have got $\neg z$. Therefore we must have x .
- (c) Assume Nx , then from 2. and 3. we get Nz .
- (d) Assume Nz , then from axiom we get $\neg z$, and therefore from 4. and 1. we get $\neg x$. If we had $\neg Nx$, we would have got from 5. and 6. $\neg Nz$, and so we must have Nx .

We thus proved $x \leftrightarrow z$ from a. and b. and we proved $Nx \leftrightarrow Nz$ from c. and d..

EXAMPLE 5.3

Consider the network $\mathcal{A}=(S, R)$ of Figure 12

The axioms of $\Delta_{\mathcal{A}}$ are the following:

- (1) $x \leftrightarrow Nx \wedge Ny$
- (2) $y \leftrightarrow Nx$
- (3) $x \rightarrow Nx$
- (4) $x \rightarrow Ny$

- (5) (a) $(\neg x \wedge \neg y) \wedge (\neg Nx \vee \neg Ny) \rightarrow \neg x \wedge \neg Nx$
 (b) $\neg x \wedge \neg Nx \rightarrow \neg y \wedge \neg Ny$

The network \mathcal{A} has two possible extensions $y = \text{in}$, $x = \text{out}$, and $x = y = \text{und}$. So we want to be able to prove (using $\Delta_{\mathcal{A}}$ and the extra axiom $NMA \leftrightarrow A$) that either $y = \text{in}$ and $x = \text{out}$ or $y = x = \text{und}$. This means we want to prove (6)

$$(6) (y \wedge Nx) \vee (\neg y \wedge \neg Ny \wedge \neg x \wedge \neg Nx)$$

Let us show we can do it!

From (1) we get $x \rightarrow Nx$ and since we have the axiom $Nx \rightarrow \neg x$ we get

$$(7) \neg x$$

We now prove

$$(8) Ny \rightarrow \neg Ny$$

Assume Ny , therefore by axiom we get $\neg y$ and so from (2) we get $\neg Nx$, and so from (7) and (5b) we get $\neg Ny$. Thus we have proved (9) below:

$$(9) \neg Ny.$$

Therefore, in view of (9) and (7), (6) becomes (6*)

$$6^*. (y \wedge Nx) \vee (\neg y \wedge \neg Nx)$$

or, equivalently (6**)

$$6^{**}. (y \vee Nx) \rightarrow y \wedge Nx.$$

In view of (2), (6**) becomes

$$6^{***}. (y \vee y) \rightarrow (y \wedge y).$$

which holds.

6 Conjunctive and disjunctive attacks in CN

In 2009, Gabbay [28] introduced the option of conjunctive (joint) attacks and the notion of disjunctive attacks. The basic configuration can be seen in Figures 13, 14, 15 and 16. In Figure 15, the nodes $\{y_1, \dots, y_m\}$ jointly mount a disjunctive attack on the nodes $\{z_1, \dots, z_n\}$. The meaning in Figure 15 is that if all of y_i are in then at least one of z_j is out. See Figure 16. Translated into CN this means:

$$\bigwedge_{i=1}^m y_i \rightarrow \bigvee_{j=1}^n Nz_j.$$

DEFINITION 6.1 (CD network)

- (1) Let S be a non-empty set of arguments. Consider the set Ω of all non-empty subsets of S . Use the notation $G, H \subseteq S$ for such subsets. Let \mathbb{R} be a binary relation on such subsets, i.e.

$$\mathbb{R} \subseteq (2^S - \emptyset)^2.$$

A conjunctive–disjunctive argumentation network (CD-network) has the form (S, Ω, \mathbb{R}) .



FIG. 13.



FIG. 14.

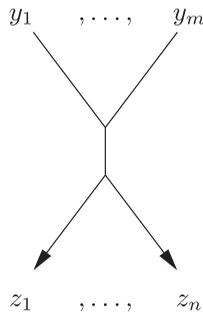


FIG. 15.

(2) When all attacked sets are singletons we get only joint attacks of the form

$$G \mathbb{R} \{x\}, x \in S.$$

We can use the notation

$$R_0 = \{(G, x) \mid G \mathbb{R} \{x\}\}.$$

(3) When all attacking sets are singletons we get only disjunctive attacks of the form $\{x\} \mathbb{R} H$.

$$\bigwedge_{i=1}^m y_i \rightarrow \bigwedge_{j=1}^n z_j$$

FIG. 16.

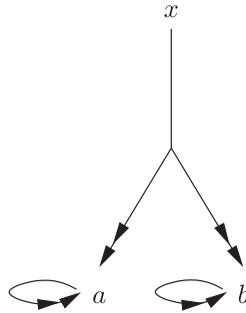


FIG. 17.

REMARK 6.2

We need to discuss the nature of the joint–disjunctive attack. The disjunctive part of the definition is different in nature from the conjunctive part. This may cause difficulties. Consider the disjunctive attack of Figure 17. x is not attacked and so $x = \text{in}$. Thus we must have either $a = \text{out}$ or $b = \text{out}$. So the extensions are

$$\begin{aligned} x = \text{in}, b = \text{out}, a = \text{und} \\ x = \text{in}, a = \text{out}, b = \text{und}. \end{aligned}$$

In CN we can write this as

$$x \rightarrow Na \vee Nb.$$

The reader should be strongly aware that the translation is from $\mathcal{A} = (S, R)$ into CN and *not* the other direction.

So if we have a joint attack of $\{x, a\} \rightarrow b$, then the translation would be $x \wedge a \rightarrow Nb$.

To make this point clearer note that the condition $x \rightarrow Na \vee Nb$ is equivalent to the following two conditions

- (i) $x = \text{in}$ and $a \neq \text{out} \rightarrow b = \text{out}$
 $x \wedge \neg Na \rightarrow Nb$
- (ii) $x = \text{in}$ and $b \neq \text{out} \rightarrow a = \text{out}$
 $x \wedge \neg Nb \rightarrow Na.$

Consider now (i) as a new kind of joint attack as in Figure 18.

In this new kind of joint attack if one or more of the attackers is undecided, then the attack still goes through. This also means that we get a new type of single attack which we denote by

$$x \text{---}\diamond\text{---} y,$$

where y is out even when $x = \text{undecided}$.

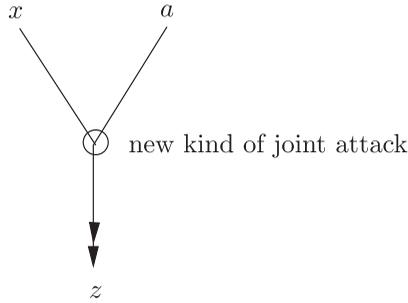


FIG. 18.

In fact, we can define $x \dashv\rightarrow y$ as the truth \top disjunctively attacking $x \wedge y$. This view also implies that $x \dashv\rightarrow y$ iff $y \dashv\rightarrow x$.

This is not a good understanding of disjunctive attacks. Our intuitive understanding of being undecided is that x is undecided because x can go either way. Either to $x = \text{in}$ or to $x = \text{out}$. So we might want to say that in Figure 14, if all z_i are either in or undecided, with one z_i undecided, then $\bigwedge_i z_i$ is undecided and hence z is undecided.

It is not our purpose to study in detail disjunctive attacks. We just want to show that the logics **CN** and **CNN** are very good in expressing attacks. The reader can see for a fact that in the **CN** language we can say whatever we want. The attack $x \dashv\rightarrow y$ can be written as $\neg Nx \rightarrow Ny$ or equivalently $\neg Ny \rightarrow Nx$.

There is no agreement in the literature on the meaning of the argument z attacking the set of arguments $H = \{z_1, \dots, z_n\}$. In [28, 33] the definition is $z \rightarrow \bigvee_j Nz_j$, while in [44] the definition is $\bigvee_j (z \rightarrow Nz_j)$. See Section 8 below for further discussion and also see [26].

These 2 options are the same in 2-valued logic but not if we have the third undecided value.

Given the uncertainty of how to define complete extensions for networks with disjunctive attacks, we can simply write the **CN**-formulas $\mathbf{F}_{\text{in}}, \mathbf{F}_{\text{out}}, \mathbf{F}_{\text{und}}$ that we want and *stipulate* that the complete extensions are to be all **CN** models of these formulas. For further discussion see Section 8 below.

Having concluded our discussion in the previous Remark 6.2, we still do need to prove a correspondence theorem for the case of joint attacks only, since there is agreement about them in the literature.

DEFINITION 6.3

- (1) An argumentation network with joint attacks has the form (S, R_0) , where S is a non-empty set of arguments and $R_0 \subseteq (2^S - \emptyset) \times S$ is the joint attack relation.
- (2) A legitimate Caminada–Gabbay labelling from S into $\{\text{in}, \text{out}, \text{und}\}$ is a function λ satisfying the following conditions:
 - (CG1) $\lambda(x) = \text{in}$ iff either x is not attacked by any $G \subseteq S$ or for every $G \subseteq S$ such that GR_0x , there exists a $y \in G$ such that $\lambda(y) = \text{out}$.
 - (CG2) $\lambda(x) = \text{out}$ iff for some $G \subseteq S$, GR_0x and for all $y \in G$, $\lambda(y) = \text{in}$.
 - (CG3) $\lambda(x) = \text{und}$ iff for all $G \subseteq S$, such that GR_0x there exists a $y \in G$ such that $\lambda(y) \neq \text{in}$, and for some $G' \subseteq S$, GR_0x we have that for all $y \in G'$, $\lambda(y) = \text{either in or und}$.
- (3) We identify the complete extensions of (S, R_0) with the legitimate Caminada–Gabbay labellings of S .

THEOREM 6.4

Let $\mathcal{A} = (S, R_0)$ be an argumentation network with joint attacks. Let $\Delta_{\mathcal{A}}$ be its associated CN theory with $\mathbf{F}_{\text{in}}^{\text{Joint}}$, $\mathbf{F}_{\text{out}}^{\text{Joint}}$ and $\mathbf{F}_{\text{und}}^{\text{Joint}}$ as defined in CC2 of Section 4 and listed below.

Then the models of $\Delta_{\mathcal{A}}$ correspond exactly to the Caminada–Gabbay labellings of \mathcal{A} according to Definition 6.3.

$$\mathbf{F}_{\text{out}}^{\text{Joint}} : \quad \left(\bigvee_{GR_0x} \bigwedge_{z \in G} z \right) \rightarrow Nx \text{ for all } x \in S$$

$$\mathbf{F}_{\text{in}}^{\text{Joint}} : \quad x \leftrightarrow \left(\bigwedge_{GR_0x} \bigvee_{z \in G} Nz \right) \text{ for all } x \in S$$

$$\mathbf{F}_{\text{und}}^{\text{Joint}} : \quad \bigwedge_{GR_0x} \left(\bigvee_{z \in G} \neg z \right) \wedge \left(\bigvee_{GR_0xz \in G} \bigwedge (z \vee \neg Nz) \right) \rightarrow \neg x \wedge \neg Nx$$

PROOF.

Part A. Assume we have a legitimate λ . We define a model $h = h_\lambda$ of $\Delta_{\mathcal{A}}$.

Let

- $h_\lambda(x) = 1$ if $\lambda(x) = \text{in}$
- $h_\lambda(Nx) = 1$ if $\lambda(x) = \text{out}$

We show that all axioms of CN hold.

- (1) Show $Nx \rightarrow \neg x$. Otherwise $h_\lambda(Nx) = h_\lambda(x) = 1$, but this means $\lambda(x) = \text{in} = \text{out}$ which is impossible.
- (2) We check $\mathbf{F}_{\text{out}}^{\text{Joint}}$. Assume $h_\lambda(\bigvee_{GR_0x} \bigwedge_{z \in G} z) = 1$. So for some GR_0x we have $h_\lambda(z) = 1$ for all $z \in G$. This means for this G that $\lambda(z) = \text{in}$ for all $z \in G$. Hence $\lambda(x) = \text{out}$. Hence $h_\lambda(Nx) = 1$.
- (3) We check $\mathbf{F}_{\text{in}}^{\text{Joint}}$.
 - (a) Assume $h_\lambda(x) = 1$. This means $\lambda(x) = \text{in}$. Hence for any GR_0x we have that for some $z \in G$, $\lambda(z) = \text{out}$. Thus for all GR_0x there is a $z \in G$ s.t. $h(Nz) = 1$, i.e. $h(\bigwedge_{GR_0x} \bigvee_{z \in G} Nz) = 1$.
 - (b) Suppose $h_\lambda(\bigwedge_{GR_0x} \bigvee_{z \in G} Nz) = 1$. This means for every GR_0x there is a z such that $h_\lambda(Nz) = 1$. But this means for every GR_0x there is a $z \in G$ s.t. $\lambda(z) = \text{out}$. Thus $\lambda(x) = \text{in}$ so $h_\lambda(x) = 1$.
- (4) We check $\mathbf{F}_{\text{und}}^{\text{Joint}}$: Assume that
 - (a) $h_\lambda(\bigwedge_{GR_0x} \bigvee_{z \in G} \neg z) = 1$
 - (b) $h(\bigvee_{GR_0x} \bigwedge_{z \in G} (z \vee \neg Nz)) = 1$.

We show that $h(\neg x \wedge \neg Nx) = 1$.

From (a) we get that every GR_0x has a z such that $h_\lambda(z) = 0$, i.e. $\lambda(z) \neq \text{in}$.

From (b) we get that there is a GR_0x such that for all $z \in G$, we have that $\neg z \rightarrow \neg Nz$ holds. Thus for one GR_0x there is a z such that $h(\neg z \wedge \neg Nz) = 1$. This means $\lambda(z) \neq \text{in}$ and $\lambda(z) \neq \text{out}$, i.e. $\lambda(z) = \text{undecided}$.

So we got that for every GR_0x , not all $\lambda(z) = \text{in}$, for $z \in G$ and for one such G' all z are either in or undecided with one $\lambda(z) = \text{und}$.

This implies $\lambda(x) = \text{und}$. So $h_\lambda(\neg x) = h_\lambda(\neg Nx) = 1$.

Part B. Assume h is a model of $\Delta_{\mathcal{A}}$. Define λ_h

- $\lambda_h(x) = \text{in}$ if $h(x) = 1$

26 *The attack as strong negation*

- $\lambda_h(x) = \text{out}$ if $h(Nx) = 1$
- $h_\lambda(x) = \text{und}$ if $h(\neg x \wedge \neg Nx) = 1$.

We show λ_h is a legitimate labelling.

- (1) λ_h is well defined because h satisfies

$$h \models Nx \rightarrow \neg x.$$

- (2) We show (CG1)

(a) If x is not attacked then $x \in \Delta_{\mathcal{A}}$ and so $\lambda_h(x) = \text{in}$.

(b) Assume that for every GR_0x there is $y \in G$ with $\lambda_h(y) = \text{out}$.

This means for every GR_0x there is $y \in G$ with $h(Ny) = 1$. So $h(\bigwedge_{GR_0x} \bigvee_{y \in G} Ny) = 1$.

So by $\mathbf{F}_{\text{in}}^{\text{Joint}}$ we get that $h(x) = 1$ i.e. $\lambda(x) = \text{in}$.

- (3) We show (CG2). Assume for some GR_0x and all $y \in G$ that $\lambda_h(y) = \text{in}$. This means

$$h(\bigvee_{GR_0x} \bigwedge_{y \in G} y) = 1.$$

Hence by $\mathbf{F}_{\text{out}}^{\text{Joint}}$, $h(Nx) = 1$, i.e. $\lambda(x) = \text{out}$.

- (4) We show (CG3). Suppose for every GR_0x there is a y s.t. $\lambda(y) \neq \text{in}$ and in some $G'R_0x$ for $y \in G'$, $\lambda(y) = \text{in}$ or undecided and that there is a $y_0 \in G'$ such that $\lambda(y_0) = \text{und}$.

Then $h(\bigwedge_{GR_0x} \bigvee_{y \in G} \neg y) = 1$ and $h(\bigvee_{G'R_0x} \bigwedge_{y \in G'} (y \vee \neg Ny)) = 1$.

This means that the antecedent of $\mathbf{F}_{\text{und}}^{\text{Joint}}$ holds and so $h(\neg x \wedge \neg Nx) = 1$ and so $\lambda(x) = \text{und}$. ■

THEOREM 6.5

Let $\mathcal{A} = (S, R_0)$ be an argumentation network with joint attacks. Then there exists a Caminada–Gabbay legitimate labelling λ for it, (i.e. it does have complete extensions).

PROOF. We reproduce a construction from Gabbay’s 2009 paper [28] which faithfully reduces joint attacks to single attacks. This means that (S, R_0) can be faithfully embedded into a traditional network (S', R') with $S \subseteq S'$. Since (S', R') has complete extensions, we will get that (S, R_0) has complete extensions.

We now proceed to generate the new points to add to S to obtain S' , and we define R' .

Let $x \in S$. Let G_1, \dots, G_n be all the attacking sets of x . Let $G_i = \{z_{i,1}, \dots, z_{i,r(i)}\}$, $i = 1, \dots, n$. Introduce new points as follows:

$$S_x = \{x(G_i), \mathbf{e}(x, G_i, z_{i,1}), \dots, \mathbf{e}(x, G_i, z_{i,r(i)}) \mid i = 1, \dots, n\}.$$

Let $S' = S \cup \bigcup_{x \in S} S_x$.

Note that for any configuration (G_1, \dots, G_n, x) , where G_i are all the sets attacking x , the sets S_x are all sets of distinct disjoint points. Note also the correspondence between each $z_{i,j} \in G_i$ and the point $\mathbf{e}(x, G_i, z_{i,j})$. So, for example, if $\{a, b\}R_0x$ and $\{a, b\}R_0y$ hold and $\{a, b\}$ is the only attacker of x and the only attacker of y , the new points will be

$$\begin{aligned} S_x &= \{x(\{a, b\}), \mathbf{e}(x, \{a, b\}, a), \mathbf{e}(x, \{a, b\}, b)\} \\ S_y &= \{y(\{a, b\}), \mathbf{e}(y, \{a, b\}, a), \mathbf{e}(y, \{a, b\}, b)\}. \end{aligned}$$

Define R' on S' as follows.

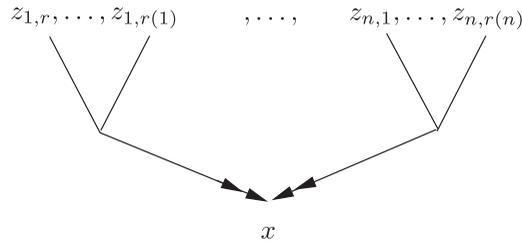


FIG. 19. Joint attacks of G_1, \dots, G_n .

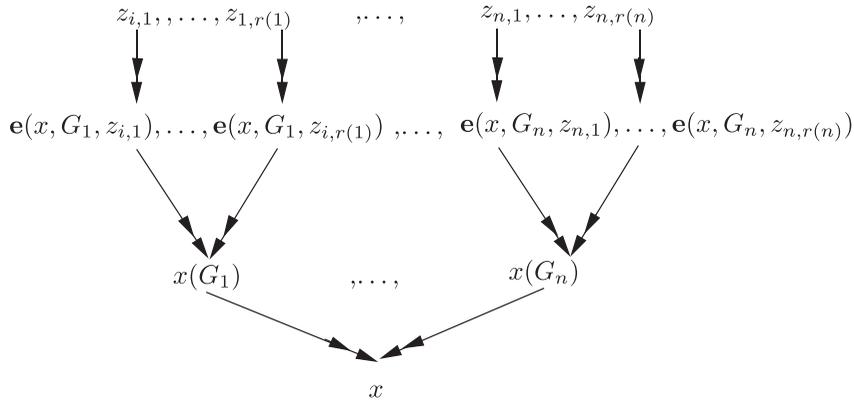


FIG. 20. Reduction of the joint attacks.

For any G_1, \dots, G_n attacking any x , let

$$\begin{aligned} & z_{i,j}R'e(x, G_i.z_{i,j}) \\ & \mathbf{e}(x, G_i, z_{i,j})R'x(G_i) \\ & x(G_i)R'x. \end{aligned}$$

Figures 19 and 20 show what we have down for the case of $x \in S$ and G_i are the only joint attacker of x with $G_i = \{z_{i,1}, \dots, z_{i,r(i)}\}, i = 1, \dots, n$.

We want to prove the following claims.

Claim 1. Let λ be a legitimate labelling of (S', R') . Then $\mu = \lambda \upharpoonright S$ is a legitimate labelling of (S, R_0) .

Claim 2. Let μ be a legitimate labelling of (S, R_0) . Then μ can be uniquely extended to a labelling λ of (S', R') and this λ is legitimate for (S', R') .

To prove the above we need some shorthand definitions.

Let G be a set of nodes. We say that G is in if every $x \in G$ is in. We say that G is out if some $x \in G$ is out and we say G is undecided if every x in G is not in and some $y \in G$ is undecided.

Proof of Claim 1 and Claim 2. This is proved in [28]. We can see the idea of the proof by comparing Figures 19 and 20.

- (1) Suppose x is in, in Figure 20 then we must have that all of $x(G_i)$ are out. For $x(G_i)$ to be out, one of $\{\mathbf{e}(x, G_i, z_{i,j}) | j = 1, 2, \dots\}$ must be in. So one of $\{z_{i,j} | j = 1, \dots\}$ must be out and so the set G_i must be out.
This means that $x = \text{in}$ implies that all G_i are out. In fact we have that x is in if and only if all G_i are out. But this is exactly the condition for $x = \text{in}$, in (S, R_0) .
- (2) Suppose x is out in (S', R') . Then some $x(G_i)$ is in. Therefore all of $\{\mathbf{e}(x, G_i, z_{i,j}) | j = 1, \dots\}$ for this G_i are out. Hence all of $\{z_{i,j}\}$ for this G_i are in, and hence G_i is in. Again this goes in both (iff) direction and is the out condition in (S, R_0) .
- (3) Suppose x is undecided in (S', R') . Then all of $x(G_i)$ are either out or undecided, with one $x(G_i)$ at least being undecided. If $x(G_i)$ is out then as we have seen before, G_i must be in. If $x(G_i)$ is undecided then all elements of G_i must be either in or if not, in then undecided, with at least one element of G_i being undecided. Again this argument is an if and only if argument and is exactly the condition for $x = \text{und}$ in (S, R_0) .
- (4) The considerations in (1), (2), (3) also show that Claim 2 is true because the values of G_i and x determine the values of $\mathbf{e}(x, G_i, z_{i,j})$ and $x(G_i)$ uniquely.

The theorem follows from the claims, since (S', R') does have legitimate labellings.

REMARK 6.6

The considerations in the proof of Theorem 6.5 show that basically the configuration in Figure 20 ‘implies uniquely’ the configuration of Figure 19. This means that if we write in the logic CNN the conditions $\mathbf{F}'_{\text{in}}, \mathbf{F}'_{\text{out}}, \mathbf{F}'_{\text{und}}$ for x and for all the additional new points in (S', R') , it should logically imply in CNN, the conditions $\mathbf{F}^{\text{Joint}}_{\text{in}}, \mathbf{F}^{\text{Joint}}_{\text{out}}$ and $\mathbf{F}^{\text{Joint}}_{\text{und}}$ for x in (S, R_0) . This means the new points play no real logical role.

Let us prove this, also as an exercise in the deduction capabilities of CNN.

Let us write all the \mathbf{F}' of x for (S', R') and prove all the $\mathbf{F}^{\text{Joint}}$ of x for (S, R_0) .

We are given the following data.

- (1) $x \leftrightarrow \bigwedge_i Nx(G_i)$
- (2) $\bigvee_i x(G_i) \rightarrow Nx$
- (3) $x(G_i) \leftrightarrow \bigwedge_j N\mathbf{e}(x, G_i, z_{i,j})$
- (4) $\bigvee_j \mathbf{e}(x, G_i, z_{i,j}) \rightarrow Nx(G_i)$
- (5)
 - (a) $z_{i,j} \rightarrow N\mathbf{e}(x, G_i, z_{i,j})$
 - (b) $\neg z_{i,j} \wedge \neg Nz_{i,j} \rightarrow \neg\mathbf{e}(x, G_i, z_{i,j}) \wedge \neg N\mathbf{e}(x, G_i, z_{i,j})$
- (6) $\mathbf{e}(x, G_i, z_{i,j}) \leftrightarrow Nz_{i,j}$
- (7)
 - (a) $\bigwedge_i \neg x(G_i) \wedge \bigvee_i \neg Nx(G_i) \rightarrow \neg x \wedge \neg Nx$
 - (b) $\bigwedge_i \neg\mathbf{e}(x, G_i, z_{i,j}) \wedge \bigvee_j \neg N\mathbf{e}(x, G_i, z_{ij}) \rightarrow \neg x(G_i) \wedge \neg Nx(G_i)$

Let us see what we can prove in CNN from (1)–(7). We start by proving $\mathbf{F}^{\text{Joint}}_{\text{in}}$:

$$x \leftrightarrow \bigwedge_{GR_0xz \in G} \bigvee Nz$$

- (8)
- (a) Assume x
 - (b) from (1) we get for all i , $Nx(G_i)$.
 - (c) From (8b) we get $\neg x(G_i)$.
 - (d) Fix an arbitrary i . We show that for at least one i we have $\mathbf{e}(x, G_i, z_{i,j})$.
 - (e) we assume in order to reach a contradiction that for all j we have $\neg \mathbf{e}(x, G_i, z_{i,j})$. If by any chance we have that for at least one j $\neg N\mathbf{e}(x, G_i, z_{i,j})$, we would get by (7b) $\neg Nx(G_i)$ also, contradicting (8b). Thus we must have $N\mathbf{e}(x, G_i, z_{i,j})$ for all j . But then by (3) we get $x(G_i)$, contradicting (8c). Therefore (10d) must hold.
 - (f) So from (6) and (10d) we get that for at least one j we have $Nz_{i,j}$.
 - (g) summarizing, having assumed x in (8a) we get from (8d) and (8e) that for all i , there exists a j such that $Nz_{i,j}$ holds. This is the direction $x \rightarrow \bigwedge_{GR_0x} \bigvee_{z \in G} Nz$.
- (9)
- (a) Assume for each i there exists a j such that $Nz_{i,j}$ holds.
 - (b) Therefore from (6) we get that for all i there exists a j such that $\mathbf{e}(x, G_i, z_{i,j})$ holds.
 - (c) Therefore from (4) we have for all i , $Nx(G_i)$ holds.
 - (d) Therefore from (1) we have that x holds.
 - (e) Summarizing, we have proved that $\bigwedge_{GR_0x} \bigvee_{z \in G} Nz \rightarrow x$.

We continue to derive $\mathbf{F}_{\text{out}}^{\text{Joint}}$.

- (10) From (5) we get $\bigwedge_j z_{i,j} \rightarrow \bigwedge_j \mathbf{e}(x, G_i, z_{i,j})$.
- (11) Using (6) and (3) we get $\bigwedge_j z_{i,j} \rightarrow x(G_i)$.
- (12) From (2) and (11) we get $\bigvee_i \bigwedge_j z_{i,j} \rightarrow Nx$. This is $\mathbf{F}_{\text{out}}^{\text{Joint}}$.
We now show $\mathbf{F}_{\text{und}}^{\text{Joint}}$.
We need first to prove (14) below.
From (3) we get
- (13) $\bigwedge_i \neg x(G_i) \leftrightarrow \bigwedge_i \bigvee_j \neg N\mathbf{e}(x, G_i, z_{i,j})$.
From (6) substituted in the right-hand side of (13) we get
- (14) $\bigwedge_i \neg x(G_i) \leftrightarrow \bigwedge_i \bigvee_j \neg NNz_{i,j}$
hence $\bigwedge_i \neg x(G_i) \leftrightarrow \bigwedge_i \bigvee_j \neg z_{i,j}$.

We now carry on to prove $\mathbf{F}_{\text{und}}^{\text{Joint}}$.

- (15) Assume the antecedents of $\mathbf{F}_{\text{und}}^{\text{Joint}}$, namely
 - (a) $\bigvee_i \bigwedge_j (z_{i,j} \vee \neg Nz_{ij})$.
 - (b) $\bigwedge_i \bigvee_j \neg z_{i,j}$.
We want to show the consequent
 - (c) $\neg x \wedge \neg Nx$.
From (a) there is an i^* satisfying
- (16) $\bigwedge_j (z_{i^*,j} \vee \neg z_{i^*,j})$.
For this i^* we use (b) and get $\bigvee_j \neg z_{i^*,j}$. Let j^* be a choice for this disjunction. We thus get that
- (17) $\neg z_{i^*,j^*} \wedge \neg Nz_{i^*,j^*}$ holds.
Therefore by axiom (5b) we get that
- (18) $\neg \mathbf{e}(x, G_{i^*}, z_{i^*,j^*}) \wedge \neg N\mathbf{e}(x, G_{i^*}, z_{i^*,j^*})$ holds.

Look again at $G_{i^*} = \{z_{i^*,1}, \dots, z_{i^*,r(i^*)}\}$. We have for each (i^*, j) that either $z_{i^*,j}$ holds in which case by (5a) $Ne(x, G_{i^*}, z_{i^*,j})$ holds or $\neg z_{i^*,j}$ holds in which case by (15) we get that $\neg z_{i^*,j} \wedge \neg Nz_{i^*,j}$ holds and therefore by (5b)

$$(19) \neg e(x, G_{i^*}, z_{i^*,j}) \wedge \neg Ne(x, G_{i^*}, z_{i^*,j}) \text{ holds.}$$

Therefore (7b) holds for G_{i^*} and hence

$$(20) \neg x(G_{i^*}) \wedge \neg Ne(G_{i^*}) \text{ holds.}$$

Let us see what we got so far. From (14) we have

$$\bigwedge_i \neg x(G_i)$$

From (19) we get

$$\bigvee_i \neg Nx(G_i)$$

holds.

Therefore the antecedent of (7a) holds and we get

$$(c) \neg x \wedge \neg Nx$$

■

7 Higher level attacks

Higher level attacks are attacks on attacks. This concept was first introduced in 2005 in [8] (see also [9] for an expanded version). The idea is very simple. Suppose we are given an abstract system (S, R) where S is a non-empty set of objects and R is a binary relation on S , (i.e. R is a set of pairs (x, y) where x and y are from S). In argumentation theory S is a set of arguments and R is the attack relation (denoted by $x \rightarrow y$, for (x, y) in R), but in general there are other interpretations for S and R . In the abstract, we can regard the elements of R also as objects and therefore we can expand the relation R into a wider relation R_1 also containing elements from R itself. This means that in argumentation we allow also to have attacks on attacks. So we can also write possibly

- (i) $z \rightarrow (x \rightarrow y)$
- (ii) $(u \rightarrow v) \rightarrow (x \rightarrow y)$
- (iii) $(u \rightarrow v) \rightarrow z$

The papers [5, 6, 29, 39, 43] deal only with possibility (i) above, but there is no mathematical reason not to generalize to all three possibilities and even iterate the process.

The problem is to extend whatever semantics we have for the case of (S, R) to the general iterated case where we turned the elements of R into objects.

In the case of higher level attacks in argumentation, we need to give semantics for the generalized higher level networks. This is not difficult, thanks to the nature of the Caminada labelling. Since we regard higher level attacks as legitimate objects to be attacked (following the pattern in (i)), we can give such objects labels from the set {in, out, undecided} and require the Caminada conditions (conditions C1, C2, C3 of the beginning of Section 3) to hold for the higher level attacks.

Our task in this article is to express higher level attacks in the system **CN**.

To show how to express higher level attacks in **CN** we shall reduce higher level attacks to joint attacks by adding points to the network. First, let us define the networks we are dealing with.

DEFINITION 7.1

- (1) Let S be a non-empty set of arguments. A higher level network based on S has the form $\mathcal{A}=(S, \rho_1, \dots, \rho_n)$ where

$$\begin{aligned} \rho_1 &\subseteq S \times S, && \text{first level attacks} \\ \vdots & && \\ \rho_{i+1} &\subseteq S \times \rho_i, && i+1 \text{ level attacks} \\ \vdots & && \\ \rho_n &\subseteq S \times \rho_{n-1}, && \text{nth level attacks} \end{aligned}$$

Note that S, ρ_1, \dots, ρ_n are all pairwise disjoint

- (2) Let $\mathcal{A}=(S, \rho_1, \dots, \rho_n)$ be a higher level network. We define the associated joint attacks network $\mathcal{A}_\tau=(S_\tau, R_\tau)$ as follows:
- (a) $S_\tau = S \cup \bigcup_i \rho_i$
 - (b) Let $x \in S_\tau$. Then there is a unique i such that $x \in \rho_i$. Let $(z_1, x), \dots, (z_k, x)$ be all the attackers of x in ρ_{i+1} . Then let the following joint attacks be in R_τ

$$\begin{aligned} \{z_1, (z_1, x)\} &R_\tau x \\ \vdots & \\ \{z_k, (z_k, x)\} &R_\tau x. \end{aligned}$$

- (3) We stipulate/define that the complete extensions of \mathcal{A} to be the restrictions of the complete extensions λ of \mathcal{A}_τ to S

EXAMPLE 7.2

Consider Figure 21

In Figure 21 we gave names to the attacks.

$$\begin{aligned} \alpha &= z \rightarrow x \\ \beta &= y \rightarrow \alpha \\ \gamma &= u \rightarrow \beta \\ \delta &= w \rightarrow \gamma. \end{aligned}$$

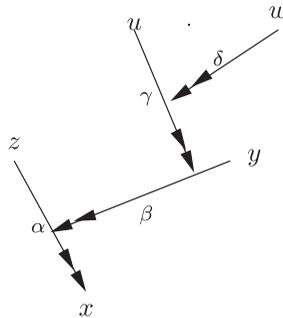


FIG. 21.

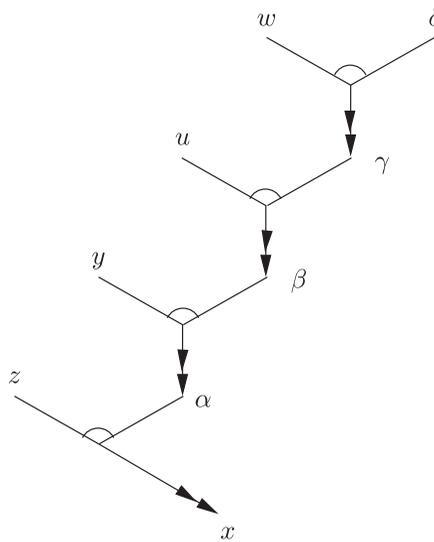


FIG. 22.

It is natural to give the double arrows names, because in higher level attacks we treat them as objects to be attacked.

Figure 21 becomes Figure 22.

This illustrates how we reduce higher level attacks to joint attacks, simply by giving names to the attacking arcs.

8 Disjunctive attacks

Our starting point is the representation of the disjunctive attack of Figure 14. We noted in Remark 6.2 that this attack is represented in CN by the formula

$$z \rightarrow \bigvee_{i=1}^n Nz_i.$$

In classical logic this formula is equivalent to the formula

$$\bigvee_{i=1}^n (z \rightarrow Nz_i).$$

On the face of it, it looks like a disjunctive attack on a set is reduced to attacking one of the elements of the set. The next example will explain what is really going on.

EXAMPLE 8.1

Consider the networks in Figures 23, 24 and 25.

The representation of the attacks of Figure 23 in CN is through (1)–(3):

(1) $a \rightarrow Nb$

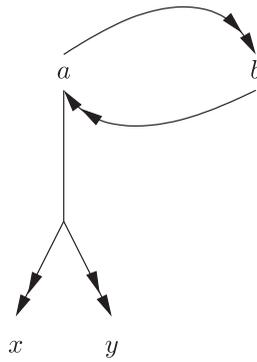


FIG. 23.

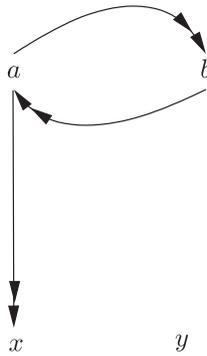


FIG. 24.

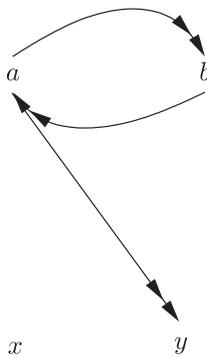


FIG. 25.

- (2) $b \rightarrow Na$
- (3) $a \rightarrow Nx \vee Ny$.

Item (3) is equivalent in classical logic to item (4):

- (4) $(a \rightarrow Nx) \vee (a \rightarrow Ny)$.

The representation of the attacks of Figure 24 is by items (1) and (2) as well as item (4.1)

$$4.1. a \rightarrow Nx$$

while the representation of the attacks in Figure 25 is by items (1), (2) and (4.2).

$$4.2. a \rightarrow Ny.$$

We recall that we said that in **CN** the attack relations of the given source network are brought forward (using N) from the meta-level in the object level. So the meaning of the equivalence in classical logic of

$$(\text{Equiv}): (a \rightarrow Nx \vee Ny) \leftrightarrow ((a \rightarrow Nx) \vee (a \rightarrow Ny))$$

is the meta-level statement (MD):

(MD): E is a complete extension of the network of Figure 23 iff E is a complete extension of at least one of the networks of Figures 24 or 25.

We make two critical comments.

(CC1) (MD) is a meta-level property. We can ask the following question (Q) for example:

(Q) Given traditional networks $(S, R_j), i = 1, \dots, k$, then under what conditions do there exist a disjunctive network (S, \mathbb{R}) such that (MD) holds?

(CC2) If our logic **CN** were not classical logic, but say intuitionistic logic (as will be investigated in Part 2 of this paper [35]), then the equivalence (Equiv) will not hold and the results would be different.

We are now ready for a formal definition of disjunctive attacks. We use Caminada–Gabbay labelling.

DEFINITION 8.2

- (1) A disjunctive argumentation network has the form (S, ρ) , where S is a non-empty set and $\rho \subseteq S \times (2^S - \emptyset)$. See Figure 26.
- (2) Let $x \in S$. Let y_1, \dots, y_k be all the elements of S which attack x directly. Let z_1, \dots, z_m all be elements of S which disjunctively attack sets which contain x , namely $z_i \rho \{x, u_1^i, \dots, u_{r(i)}^i\}$. We say z_i indirectly attacks x .
- (3) Let $\lambda : S \mapsto \{\text{in}, \text{out}, \text{und}\}$. We say that λ is a legitimate Caminada–Gabbay labelling for (S, ρ) iff the following conditions hold:
 - (D1) $\lambda(x) = \text{in}$ if $\lambda(y_i) = \text{out}, i = 1, \dots, k$ and for all $1 \leq j \leq m$, if $\lambda(z_j) = \text{in}$ then for some $k, \lambda(u_k^j) = \text{out}$.

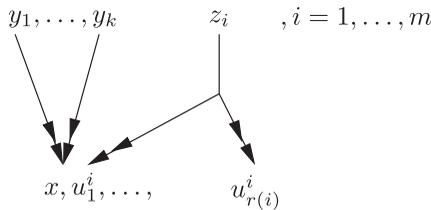


FIG. 26.

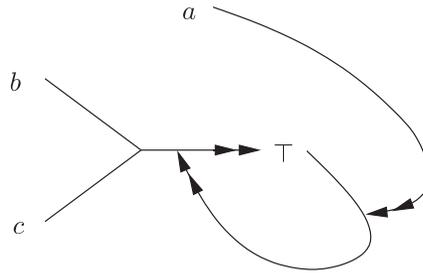


FIG. 27.

- (D2) $\lambda(x) = \text{out}$, if for some $1 \leq i \leq k, \lambda(y_i) = \text{in}$, or for some $j, (\lambda(z_j) = \text{in}$ and for all $1 \leq k \leq r(j)$ we have $\lambda(u_k^j) \neq \text{out}$.
- (D3) $\lambda(x) = \text{und}$ iff all attacks on x direct or indirect are either out or undecided, and where at least one attack is undecided, where the following define the meaning of the terms we just used.
 - A direct attack on x by y is out if $\lambda(y) = \text{out}$. It is undecided if $\lambda(y) = \text{und}$.
 - An indirect attack of z on x using the disjunctive attacked set $\{x, u_1, \dots, u_r\}$ is out if either $\lambda(z)$ is out or for some $j, \lambda(u_j)$ is out. The attack is undecided if $\lambda(z)$ and $\lambda(u_j)$ are all different from out, with at least one of them is und.

REMARK 8.3

Using higher level attacks and \top , we can implement disjunctive attacks using conjunctive attacks. See Figure 27. This figure implements the disjunctive attack $a\rho\{b, c\}$. We have that $\{b, c\} \rightarrow \top$ and $\top \rightarrow (\{b, c\} \rightarrow \top)$ and $a \rightarrow (\top \rightarrow (\{b, c\} \rightarrow \top))$

9 Comparison with literature

9.1 Comparison with the paper of S.Villata, G. Boella and L. van der Torre: Attack Semantics for Abstract Argumentation

We compare with paper [48]. The paper is different from ours, but since it has a similar name, we should address it.

Our article translates argumentation into classical logic with a symbol ‘ Na ’, meaning ‘ a is under attack’. Paper [48] regards the attack arrows as objects and discusses semantics for them. (In our article ‘ $a \rightarrow b$ ’ is a wff ‘ $a \rightarrow Nb$ ’). Our set up is completely different. Furthermore, the idea of regarding the attack arrow as an object and labelling it already appears in [29, Figure 13] and in [8]. However, we do want to make a point in this article using the machinery of [48] and so let us quote some passages from them and then make our point.

Begin quote 1:

Example 2 Consider $AF = \langle A, \rightarrow \rangle$ with $A = \{a; b; c; d; e\}$ and $\{a \rightarrow b, b \rightarrow a, b \rightarrow c, c \rightarrow d, d \rightarrow e, e \rightarrow c\}$ visualized in Figure 2. The complete extensions are $\mathcal{E}_{\text{compl}}(AF; A) = \{\emptyset, \{a\}, \{b; d\}\}$. \emptyset is the unique grounded extension, $\{a\}$ and $\{b; d\}$ are the preferred extensions, and $\{b; d\}$ is the stable extension. In the complete extension \emptyset the arguments are rejected, because they are not defended. In the extension $\{b; d\}$ the other arguments are rejected, because they are attacked by an accepted counterargument.

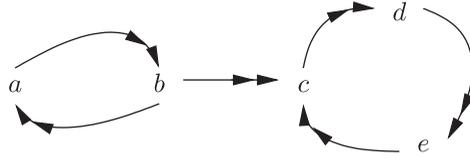


Figure 2: The argumentation framework of Example 2.

Begin quote 2:

Example 3 (Continued from Example 2 in quote 1 above): The grounded extension \emptyset has only successful attacks, the preferred extension $\{a\}$ has $\{a \leftrightarrow b, c \leftrightarrow d, d \leftrightarrow e, e \leftrightarrow c\}$ as successful attacks, and the stable extension $\{b, d\}$ has $\{b \leftrightarrow a, b \leftrightarrow c, d \leftrightarrow e\}$.

Begin quote 3:

We define attack labeling analogous to argument labeling. An attack is *in* when its attacking argument is *in*, an attack is *undecided* when its attacking argument is *undecided*, and an argument is *out* when its attacking argument is *out*. An attack is successful when it is *in* or *undecided*, whereas an argument is accepted when it is *in*. For example, if an argument is rejected, but at least one of its attacks is successful, then the argument is undecided.

Example 5 (Continued from Example 3 in quote 2 above). The grounded extension \emptyset has only undecided attacks, the preferred extension $\{a\}$ has in attack $a \leftrightarrow b$ and undecided attacks $c \leftrightarrow d, d \leftrightarrow e, e \leftrightarrow c$, and the stable extension $\{b, d\}$ has in attacks $b \leftrightarrow a, b \leftrightarrow c, d \leftrightarrow e$ and no undecided attacks.

End quotes.

Let us implement the attack network of Example 2 in quote 1 above and of its Figure 2 in our CN logic using the idea of [29, Figure 13]. Note that this idea is NOT what [48] does but we start with our own approach for comparison. We add for each arrow $x \rightarrow y$ a new node $z_{x,y}$ and translate the attack $x \rightarrow y$ into the conjunctive attack $(x, z_{x,y}) \rightarrow y$. We thus get Figure 28. This figure has conjunctive attacks and can be dealt with within CN.

Note that all the attack arrows, $z_{a,b}, z_{b,a}, z_{b,c}, z_{c,d}, z_{d,e}$ and $z_{e,c}$ are all in, because they are not attacked. This is as it should be according to the Dung traditional approach.

However, as we said, the above is not the attack semantics of [48]. According to [48] an attack $x \rightarrow y$ is out if x is out. So the attacks in Figure 28 are not always in, but depend on the extensions chosen, as discussed in [48] and quoted above in Example 5 of quote 3. So we need a new figure to implement the attack semantics of [48].

To achieve that we need to add that $\neg x$ attacks $z_{x,y}$. This is easy to write in CN: $\neg x \rightarrow Nz_{x,y}$. No new figure is really needed. However we can, if we insist, use Figure 29. It is a quite complicated higher level figure.

If we do not want to use higher level attacks, we can use Figure 30. The figures tend to be complicated because we need to add the nodes $\neg x$ on $x \rightarrow y$ (or on $z_{x,y}$). In CN we just write $\neg x \rightarrow Nz_{x,y}$, which is much simpler and *does not add any nodes!*

Independent of the problem of representing the attack semantics of [48] and the idea that *if x is out then $x \rightarrow y$ is also out*, is compatible with another idea of ours, from [28, Section 4.3]. In [28], we regard $x \rightarrow y$ as a channel, carrying an attack signal, sent by x towards y . So if x is out, no signal is sent but the channel is alive. We can say that according to [48], if x is out, the channel itself is out.

Let us now say more about our approach of [28, Section 4.3]. We want to look at the attack $a \rightarrow b$ as a signal from a to b . As such it can be considered as an object (a wave front). It therefore can split

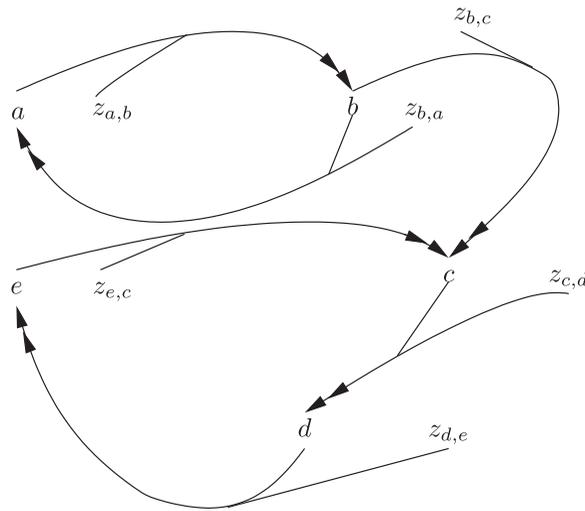


FIG. 28.

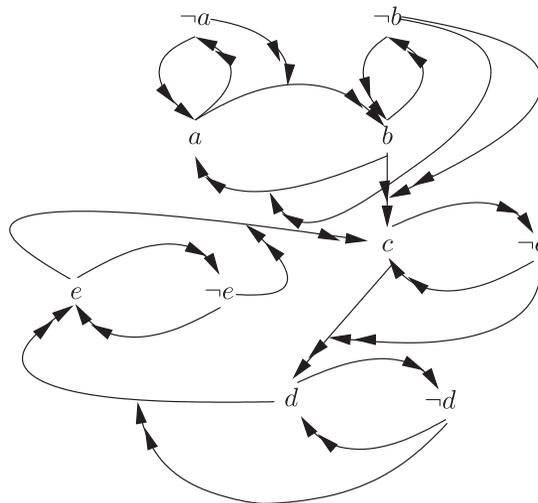


FIG. 29.

or it can interfere/join with another signal. This idea was presented in [28, Section 4.3] and will be fully developed in [26]. Figure 31 illustrates this idea. Compare with Figure 23.

In Figure 31 the attack emanates from a and disjunctively splits into two attacks one that attacks both x_1 and x_2 and one that attacks both y_1 and y_2 . Success, should a be ‘in’, is that either both x_1 and x_2 are out or both y_1 and y_2 are out.

The attacks of Figure 31 can be described in CN by the clauses:

- (1) $a \rightarrow Nb$
- (2) $b \rightarrow Na$
- (3) $a \rightarrow N(x_1 \vee x_2) \vee N(y_1 \vee y_2)$

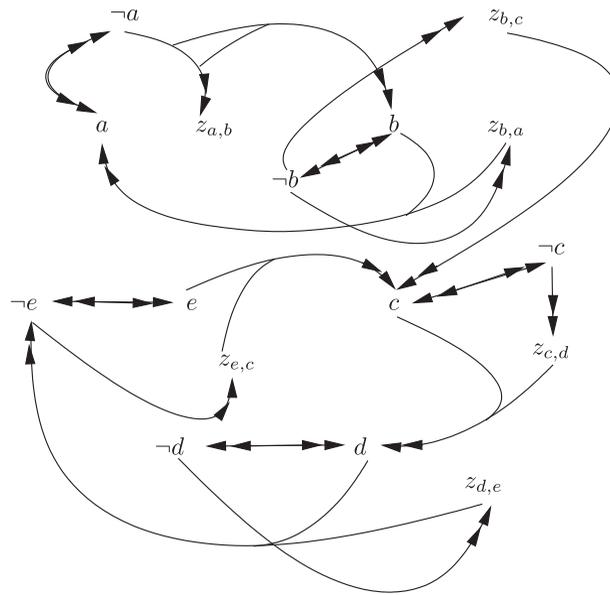


FIG. 30.

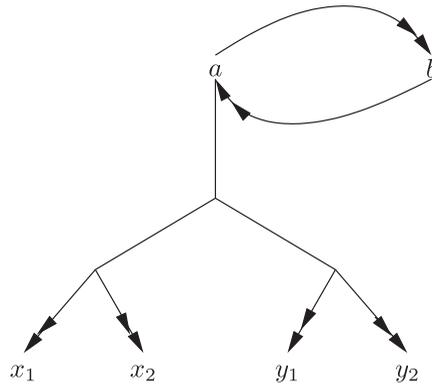


FIG. 31.

9.2 Comparison with Gabbay's paper: modal provability foundations for argumentation networks

In 2009, Gabbay [30] asked a simple question:

(Q2009) Given an argumentation network $\mathcal{A} = (S, R)$, what is its logical content $\Gamma_{\mathcal{A}}$ in terms of some reasonable logic \mathbf{L} ?

Gabbay answered this question using a version of a modal logic of provability called **LN1**. The similarity with our article is clear. We associate a modal logic **CNN** and a theory $\Gamma_{\mathcal{A}}$ with \mathcal{A} . The view of [30] is that $\Gamma_{\mathcal{A}}$ is the theory satisfying more or less (for accuracy see [30]) the equivalence

below:

$$\Gamma_{\mathcal{A}} \leftrightarrow \Box \left(\bigwedge_x (x \leftrightarrow \bigwedge_i \Diamond (\Gamma_{\mathcal{A}} \wedge \neg y_i)) \right),$$

where y_i are all the attackers of x in \mathcal{A} .

In words: the logical content Γ says that x is in iff it is possible, from the point of view of Γ , to have all the attackers of x out/false. This is a fixed point equation for Γ .

The above is the idea of [30]. Once [30] develops the semantics for the logic, we end up with a three point linear modal logic (t_1, t_2, t_3) , with t_1 the actual world and t_j accessible to $t_{j-1}, j=3, 2$. In other words we have $t_1 < t_2 < t_3$, where ' $<$ ' is the modal accessibility relation.

The labelling obtained from such a model is the following, where the triple (v_1, v_2, v_3) indicates the respective values in worlds (t_1, t_2, t_3) :

$$\begin{aligned} \text{in} &= (\top, \top, \top) \\ \text{out} &= (\perp, \perp, \perp) \\ \text{und} &= (\top, \perp, \top) \\ \text{und}^* &= (\top, \perp, \perp) \end{aligned}$$

As you can see, [30] has a completely different point of view, and furthermore, the point of view of [30] can get two types of undecided! Comparing with our current paper we have a classical point of view, and have a two worlds linear modal logic, where the assignment is restricted like in intuitionistic logic, giving rise to the following labelling with only one type of undecided:

$$\begin{aligned} \text{in} &= (\top, \top) \\ \text{out} &= (\perp, \perp) \\ \text{und} &= (\perp, \top) \end{aligned}$$

To obtain a better comparison, let us organize the worlds t_1, t_2, t_3 in different accessibility ordering, say $<^*$:

$$t_2 <^* t_3 <^* t_1.$$

The four possible values can now be seen as in Figure 32

For the ordering $t_2 <^* t_3 <^* t_1$, the assignments are clearly intuitionistic. True continues to be true in this ordering.

Consider now the loop of Figure 33.

In this loop b attacks itself, so its degree of undecided must be possibly different from the degrees of the other points in the loop and must be in the middle, equally away from the value 'in' and from the value 'out'. Indeed paper [30] can distinguish between the node b being undecided and the other nodes. There is a problem, however, in this case. There are only two values of undecided, und and und*, and so there is no middle, so what value do we give b ? Paper [30] calculates the values according to its fixed point semantics rationale and gives b the value und, while the value of a is und* and the value of c is also und. Thus according to [30], we get:

$$a = (\perp, \perp, \top), b = c = (\perp, \top, \top).$$

model	in	out	und	und*
t_1	\top	\perp	\top	\top
t_3	\top	\perp	\top	\perp
t_2	\top	\perp	\perp	\perp

FIG. 32.

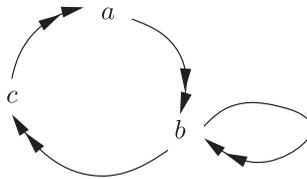


FIG. 33.

We, however, would like to give b the other value of undecided, namely $b = (\perp, \perp, \top)$, and so we have to give $c = (\perp, \top, \top)$ and have $a = b$. We can offer the following rationale for these values:

- (1) x being undecided means that x can go either way, to be in or be out, but is neither, i.e. $x \neq (\top, \top, \top), x \neq (\perp, \perp, \perp)$.
- (2) Different vectors for undecided relate to how likely x is to be in or out. So when comparing the sequences (\perp, \top, \top) with (\perp, \perp, \top) , the less ‘ \top ’ we have in the sequence, the nearer ‘out’ $= (\perp, \perp, \perp)$ is the value of the sequence.
- (3) In Figure 33, we need to choose the value for b . Paper [30] adopted a generous credulous view for letting an argument x to be in, namely that x is in if it is possible for all its attackers y to be out. Our view in this article is more skeptical and strict, it is that x is in if all attackers y are strongly out ($\mathcal{N}y$ holds). So if paper [30] calculated that $b = (\perp, \top, \top)$, and our view is different, then we might consider the other undecided value, namely $b = (\perp, \perp, \top)$. This option agrees with the equational approach, (see Remark 9.1 below) and so we choose (contrary to [30]), to take the value of b to be (\perp, \perp, \top) .
- (4) c is attacked by b , which is nearer $b =$ out, so c must be still undecided, but nearer the value in. So we expect c to have the value (\perp, \top, \top)
- (5) From (4) we expect a to be still undecided but nearer the value out and so our only option is to have $a = b = (\perp, \perp, \top)$.

The comparison and discussion of the rationale above suggests that we can look at logics like $\text{CNN}k$, where $k \geq 2$ defined by chains of worlds of the form

$$w_1 <^* w_2 <^* \dots <^* w_k.$$

The undecided values can be of the form

$$\text{und}_j^* = (\perp, \dots, \perp, \top, \dots, \top).$$

where $1 < j \leq k$ and the value is \perp until position $k - 1$ and \top afterwards.

We reserve the study of CNN_k for Part 2 of this article.

REMARK 9.1

Note that the equational approach of [27] also supports our analysis and choice of value for b . In this approach we solve the following equations of the network of Figure 33.

- (1) $b = (1 - b)(1 - a)$
- (2) $c = (1 - b)$
- (3) $a = (1 - c)$

From (2) and (3) we get

$$(4) \quad a = b$$

Substituting in (1) we get

$$(5) \quad a = (1 - a)^2$$

which solves to $a = b = 0.38$ and $c = 0.62$.

This is in agreement with our analysis provided we understand undecided as numerical value strictly between 0 and 1 and nearer out to mean a numerical value nearer to 0.

We can also view the equational approach as a CNN_∞ approach, where the possible worlds intuitionistic model is taken to be $[0, 1]$, with 0 the actual world and the value $x = k$ for $x \in S$ and $k \in [0, 1]$ means

- $y \models x$ iff $x \leq k$

Again, this connection will be investigated in Part 3 of this article.

9.3 Comparison with David Pearce logic

We compare our logic CNN with Pearce's logic of [45]. Pearce introduced his logic in 1995 in the context of studying Answer Set Programming (ASP). Pearce's logic is very similar to our logic and so we need to offer a comparison. Furthermore, Pearce did apply his logic to argumentation.³ In Pearce's own words [45], equilibrium logic N_5 provides

- (1) a general methodology for building non-monotonic logics;
- (2) a logical and mathematical foundation for ASP-type systems, enabling one to prove useful metatheroetic properties;
- (3) further means of comparing ASP with other approaches to nonmonotonic reasoning.

The semantics for Pearce's logic uses the same two possible worlds linear model which we use, namely an actual world, t , a possible world s , with the restriction on atoms q that if $t \models q$ then $s \models q$. The difference between us and Pearce is in what we actually do with this semantics.

³The following is a private communication from David Pearce:

'Equilibrium logic was designed to capture stable reasoning and, not surprisingly, you can apply it to capture stable extensions in argumentation theory. Together with other colleagues we were able to show how to capture stable extensions in assumption-based argumentation. However, including strong negation would be a new topic.'

In order to be able to compare the two logics, we need to agree on notation. Our approach and notation so far can be summarised as follows:

Comments Group 1. The model has two worlds, t and s . t is the acutral world and s is possible relative to t . For atomic q we have $t \models q$ implies $s \models q$.

- (1) Our view is that this model is a modal semantics model with two possible worlds linearly ordered whose atomic assignments satisfy a restriction.
- (2) Pearce's view is that this model is an intuitionistic Kripke model with two worlds (Gödel's logic with two worlds). This semantics is well known and can be axiomatised. See [45].

Comments Group 2. We use the connectives \neg (classical negation), \rightarrow (classical implication) and modality N . We also use \wedge and \vee .

The semantical conditions are as follows

- (1) $t \models \neg A$ iff $t \not\models A$
- (2) $t \models A \rightarrow B$ iff ($t \models A$ implies $t \models B$)
- (3) Same as above for s
- (4) $t \models NA$ iff $s \models \neg A$
- (5) $s \models NA$ iff $t \models \neg A$
- (6) $x \models A \vee B$ iff $x \models A$ or $x \models B, x \in \{t, s\}$
- (7) $x \models A \wedge B$ iff $x \models A$ and $x \models B, x \in \{t, s\}$

Comments Group 3. Pearce uses intuitionistic connectives and Nelson strong negation. He calls his system N_5 , the logic of here and there. To describe Pearce's system N_5 , let us use the notation below to distinguish the intuitionistic connectives from the classical ones and to distinguish N_5 from CNN. Let us use the following:

- (1) ' \neg ' for intuitionistic negation. Its semantic satisfaction condition is
 - $t \models \neg A$ iff $t \not\models A$ and $s \not\models A$
 - $s \models \neg A$ iff $s \not\models A$
 (Note that for atoms we have $t \models A$ implies $s \models A$!)
- (2) ' \Rightarrow ' for intuitionistic implication with the semantic conditions
 - $t \models A \Rightarrow B$ iff ($t \models A$ implies $s \models B$) and ($s \models A$ implies $s \models B$)
 - $s \models A \Rightarrow B$ iff ($s \models A$ implies $s \models B$).
- (3) Use \wedge and \vee with the same semantic conditions as Comment 2.6 and 2.7 respectively.
- (4) Note that for wffs with \wedge, \vee, \neg and \Rightarrow we have
 - $\models A$ implies $s \models A$
- (5) Pearce adds strong negation ' \sim ' to the system N_5 in a similar way to the way we added \mathbf{N} in Section 1. With each atom q , Pearce adds another atom $\sim q$ and adds the axiom $\sim q \Rightarrow \neg q$. Thus N_5 is the intuitionistic theory $\{\sim q \rightarrow \neg q\}$ based on the atoms $\{q, \sim q \mid q \text{ atomic}\}$. Note that for us in CNN, we added Nq with the axiom $Nq \rightarrow \neg q$ and we have $t \models Nq$ iff $s \models \neg q$. This implies that if $\neg q \wedge Nq$ holds at t then $\neg q$ holds at t .

Just like we did for N in allowing N to apply to any wff and pushing N through to the atoms, Pearce does a similar thing for \sim . \sim satisfies the following rules both at t and at s :

- $\sim(A \wedge B)$ iff $\sim A \vee \sim B$
- $\sim(A \vee B)$ iff $\sim A \wedge \sim B$
- $\sim A \Rightarrow B$ iff $A \wedge \sim B$

- $\sim(\neg A)$ iff A
- $\sim\sim A$ iff A .

These rules for ‘ \sim ’ allow us to push ‘ \sim ’ right in front of the atoms.

Comments Group 4. Comparison so far is as follows:

- (1) Both ‘ \sim ’ and ‘ N ’ associate with any atom q the atom $\sim q$ or Nq respectively. Both \sim and N can be pushed down to the atoms syntactically. N , however, directly changes the world it is evaluated in. It is a true explicit modality. In comparison, \sim remains in the same world. It basically adds a strong atom $\sim q$ to any atom q , but $\sim q$ is evaluated semantically in the same world. It can push the evaluation to another world only through the axioms $\sim\neg q \Rightarrow q$ and $\sim q \Rightarrow \neg q$ and this is a roundabout way of doing it. Our logic **CNN** also added with each atom q another atom Nq , but Nq connects with an accessible world where q must be false.
- (2) Pearce moves from world t to world s using \neg and \Rightarrow . We move from t to s using N . We can also move from s to t using N . Pearce has no direct connective which can move the evaluation from world s to world t .

Comments Group 5. To further compare **CNN** and N_5 , let us define the intuitionistic connectives in our logic **CNN**.

- (1) We let

$$\begin{aligned}\neg X &= \text{def. } \neg X \wedge NX \\ (X \Rightarrow Y) &= \text{def } (X \rightarrow Y) \wedge N(X \wedge \neg Y \wedge Ny).\end{aligned}$$

- (2) Let A and B be two wffs built up from the atoms and the above connectives \neg and \Rightarrow and \wedge and \vee , as defined in Comment 5.1 above. In other words, we have a sublanguage **INN** of **CNN** with wffs defined as follows:
 - atoms q are in **INN**
 - If A and B are in **INN** then so are $A \wedge B$ and $A \vee B$
 - if A and B are in **INN**, so are
 - $(\neg A \wedge NA)$, this is $\neg A$
 - and
 - $(A \rightarrow B) \wedge N(A \wedge \neg B \wedge NB)$, this is $A \Rightarrow B$.
- (3) We now check and see that the truth conditions of intuitionistic semantics hold for these translations (i.e. the condition in Comment 3.4 holds).
 - (3.1) For atomic q the condition $t \models q$ implies $s \models q$ holds.
 - (3.2) Assume by induction that the condition in Comment 3.4 ($t \models X$ implies $s \models X$) holds for $X = A$ and $X = B$. We show that it holds for $A \wedge B$ and $A \vee B$. This is immediate.
 - (3.3) Assume that the condition in Comment 3.4 holds for A and we show that the condition in Comment 3.4 holds for $\neg A$. We have
 - $t \models \neg A$ iff (by definition) $t \models \neg A \wedge NA$ iff $t \models \neg A$ and $s \models \neg A$ iff $t \not\models A$ and $s \not\models A$.
 - We also have
 - $s \not\models A$ iff $s \not\models A$ and $t \not\models A$
 - (because our inductive hypothesis was that Comment 3.4 holds, i.e. $t \models A$ implies $s \models A$)
 - iff $s \models \neg A$ and $s \models NA$ iff $s \models \neg A \wedge NA$ iff (by definition) $s \models \neg A$.

- We show now that if $t \models \neg A$ then $s \models \neg A$. We have: $t \models \neg A$ iff $t \models \neg A \wedge NA$. This implies $s \not\models A$ and we have seen before that this is equivalent to $s \models \neg A$.
- (3.4) We now check the case of $A \Rightarrow B$. Assume by induction that $t \models X$ implies $s \models X$ holds for $X = A$ and for $X = B$. We show that the semantic condition of Comment 3.2 holds for $A \Rightarrow B$ and that also if $t \models A \Rightarrow B$ holds then $s \models A \Rightarrow B$ also holds. We have
- $t \models A \Rightarrow B$ iff (by definition) $t \models (A \rightarrow B) \wedge N(A \wedge \neg B \wedge NB)$ iff $t \models A \rightarrow B$ and $s \models (\neg(A \wedge \neg B) \vee \neg NB)$ iff ($t \models A \rightarrow B$ and $s \models A \rightarrow B$) or ($t \models A \rightarrow B$ and $s \models \neg NB$). But $s \models \neg NB$ iff $t \models B$. So we have that: $t \models A \rightarrow B$ iff $t \models (A \rightarrow B) \vee B$.
- Therefore we can continue
 $t \models A \Rightarrow B$ iff ($t \models A \rightarrow B$ and $s \models A \rightarrow B$) \vee ($t \models A \rightarrow B$ and $t \models B$). But $t \models B$ implies $s \models B$ and we continue
iff $t \models (A \rightarrow B) \wedge s \models A \rightarrow B$. This is the correct condition for the case of t .
We check s . $s \models A \Rightarrow B$ iff by definition $s \models (A \rightarrow B)$ and $s \models N(A \wedge \neg B)$ iff $s \models (A \rightarrow B)$ and $t \models A \rightarrow B$.

Given the above, we now show that

- $s \models A \Rightarrow B$ iff $s \models A$ then $s \models B$

If $s \models A \Rightarrow B$ then by the above $s \models A \rightarrow B$ and so if $s \models A$ then $s \models B$.

If $s \models A$ implies $s \models B$ then $s \models A \rightarrow B$. So the only option for $s \not\models A \Rightarrow B$ is that $s \not\models N(A \wedge \neg B \wedge NB)$ and so $t \models A \wedge \neg B \wedge NB$. But then this implies by the condition of Comment 3.4 for A that $s \models A \wedge \neg B$, contradicting what we have just assumed that $s \models A \rightarrow B$.

It remains to show $A \Rightarrow B$ satisfies the condition of Comment 3.4. This is obvious, however, from the semantic condition of satisfaction for ' \Rightarrow '.

- (3.5) We summarize the results of the current item (Comment 5.3). We have just shown that **CNN** contains the intuitionistic part of N_5 as the fragment **INN** (we defined the intuitionistic negation and intuitionistic implication in our system and shown they behave correctly). The reader can verify that the law of excluded middle for this embedded intuitionistic fragment does not hold, so the fragment is not trivial, (i.e. $q \vee (\neg q \wedge Nq)$ is not a theorem of **CNN**, take $q = (\perp, \top)$).

We now ask, can we also define ' \sim ' as well using N ?

Let us check. We need a formula of **CNN**, say $\alpha(q)$, which will act as the connective \sim , namely

$$\sim q = \text{(by definition)} \alpha(q)$$

This formula must satisfy

- (a) For atomic q , $\alpha(q) \Rightarrow \neg q$
- (b) For any formula A of **INN**, $\alpha(\neg A) \Leftrightarrow A$

If the value of q is taken to be (\perp, \top) or (\top, \top) , we get because of (a). and the fact that in this case the value of $\neg q$ is (\perp, \perp) , that the value of $\alpha(q)$ must be (\perp, \perp) . To summarize we must have (c) and (d). to hold

- (c) $\alpha(\perp, \top) = (\perp, \perp)$
- (d) $\alpha(\top, \top) = (\perp, \perp)$

but if we have this, how can (b) hold? For both cases where value of q is taken to be (\perp, \top) or (\top, \top) , the value of $\neg q$ is the same and so how can we have

$$\alpha(\neg q) \Leftrightarrow q?$$

The only way out is to say that rule (b) is a syntactical reduction, making \sim not a real connective. O.K. this is acceptable but different from our logic where N is a connective. So (b) holds syntactically, this means that (a) has to hold by a restriction on the assignment of the formal syntactical atom ' $\sim q$ '. The only way to it is to take ' $\sim q$ ' to be the same as ' $\neg q$ '. This is fine except that when we iterate to have

$$\neg\neg q \Rightarrow q.$$

The first ' \neg ' is not a connective but a syntactical rewrite operator and only the second \neg is a connective and the implication is valid. This is not a problem because we will write it syntactically as $\sim\neg q \Rightarrow q$ and the ' \sim ' will rewrite the following ' \neg ' out, leaving only $q \Rightarrow q$.

Comments Group 6. We now start with N_5 as our given intuitionistic system with $\{\neg, \Rightarrow, \wedge, \vee, \sim\}$ and the semantics as described in [45] and in (Comment Group 3) above and check whether we can define CN using the connectives of N_5 . We need to define N and the classical connectives $\{\neg, \rightarrow, \wedge, \vee\}$. We show that this is not possible. Consider the formula of CN.

- $\neg(q \wedge Nq), q$ atomic.

This formula is always true at world t but can be false at world s (for $q = \top$ at s and at t).

No intuitionistic wff, even with ' \sim ' can be true at t and false at s . Therefore N and \neg are not definable in Pearce's logic.

It may still be possible to translate argumentation networks into N_5 using its strong negation. We leave this study to part 2 of our article.

9.4 Comparison with four meta-level approach papers [21,24,25,38] and [11]

To compare with papers of Dvorak *et al.*, Doutre *et al.*, and Grossi, we need some preliminary methodological discussion about interpretations and translations. All the above papers translate argumentation networks into other systems and so to explain what they are doing and compare with our article we need a framework of reference, a matrix schema for comparison.

Consider the schematic Figure 34. This figure is a schema for possible embeddings, containing four systems. We name the system and give examples to help the reader visualize the schema. We then discuss options for interpretations in general and then explain how the papers we are discussing fall into the schema of Figure 34. Once we understand how these papers fit into the schema, our comparison will be concluded.

In Figure 34, the top left circle is a system **T**. Think of **T** as classical logic and think of x, e, y' as formulas of classical logic. Think of **T**⁺ as an extension of **T**, say if **T** is classical logic then **T**⁺ extends **T** with monadic second order quantification.

The system **S**₁ can be interpreted into **T**. Say x is mapped to x' (the arrows indicate the mapping) and y is mapped to y' . Think of **S**⁺ as an extension of **S**. Say **S** = intuitionistic propositional logic and **S**⁺ is some extension of it. The element e in **T** is not a result of the mapping of **S** into **T**. It has no source in **S**. Similarly **S**₂ is mapped into **T** and **S**₂⁺ extends **S**₂. **T**⁺ can be mapped into an extension **S**₂⁺ of **S**₂. If not all of **T**⁺ is mapped, at least some of it, say α , is mapped onto α' .

Think of **S**₂ as modal logic, mapped into classical logic. Think of α as a quantifier of second order, which is realized/mapped as extending modal logic (as a new modality $\Box_\alpha?$, for example).

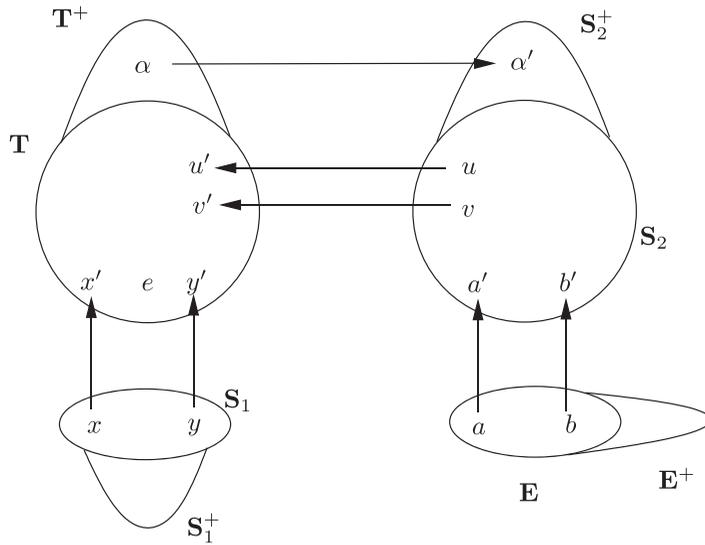


FIG. 34.

The systems **E** and **E⁺** are mapped into modal logic. Think, e.g. of **E** as a substructural logic or a default system or some normative system or as an argumentation system being interpreted in modal logic.

We now discuss possible properties of such embeddings.

Type 1: encapsulation of S in T. This type gives names in **T** to the elements of **S** and mirrors in **T** in all the movements of **S**. It is like embedding **S** in **T** using Gödel numbers. Such encapsulation is used in translating systems into classical logic in order to use Theorem Proving machines to talk about/manipulate **S**. None of the papers involved above are pure encapsulation. Encapsulation is very general, **S** can be a recipe for making pizzas and **T** can just talk about it. **T**'s role in encapsulation is that of a Turing machine.

Type 2: Meta-level embedding of S into T. This type is most common. The language **T** describes in a meaningful way the system **S**. **T** acts as a meta-level language describing **S** by internally mirroring **S**. To understand how this works, think of a mathematical theory modelling some physical phenomenon, say in an equational model governing the motion of a planet around the sun. The meta-theory **T** describing modelling **S** (e.g. argumentation) in this way is not supposed to provide a meaning (semantics) for what it describes. It is just supposed to provide a formal language of accurate description of **S**, allow one to describe variation of **S** and enable the use of properties and tools available in **T** to investigate **S** (very common is to use **T** for complexity issues for **S**). Of course, one tends to use **T** which is well known, general enough and well investigated and well endowed with tools.

To summarize Type 2.

- (P1) **T** describes **S** from the meta-level point of view but does not give **S** any meaning.
- (P2) **T** can describe at the same time several different **S₁, S₂** and so can serve as an environment for suggesting combinations of **S₁** and **S₂**. For example, if **T** is classical logic and **S₁** is modal

logic and S_2 is intuitionistic logic, then embedding S_1 and S_2 into T , may suggest how to formulate modal intuitionistic logics.

- (P3) Any tool available for T can be used for S . So e.g. any complexity studies for classical logic can be used for modal logic through the embedding of modal logic inside classical logic.
- (P4) New research ideas can be imported from T to S . Some may be interesting for S and some may be just mathematics hacks of no interest.

The paper of Dvorak *et al.*, paper [25] is an example of this. The formal mathematical language is second-order monadic first-order logic. This can serve as the modelling language for the majority, if not all, of the varieties of argumentation networks. It is intended by the authors to be to argumentation like ALGOL is to algorithms. It is an exact mathematical logic language strong enough to express whatever you want to say about argumentation networks. To quote the authors of [25] own words:

Begin quote.

We propose the formalism of monadic second order logic (MSO) as a unifying framework for representing and reasoning with various semantics of abstract argumentation. We express a wide range of semantics within the proposed framework, including the standard semantics due to Dung, semi-stable, stage, cf2, and resolution-based semantics. We provide building blocks which make it easy and straightforward to express further semantics and reasoning tasks. Our results show that MSO can serve as a *lingua franca* for abstract argumentation that directly yields to complexity results. In particular, we obtain that for argumentation frameworks with certain structural properties the main computational problems with respect to MSO-expressible semantics can all be solved in linear time. Furthermore, we provide a novel characterisation of resolution-based grounded semantics.

.

.

.

Starting with the seminal work by Dung, the area of argumentation has evolved to one of the most active research branches within Artificial Intelligence. Dung’s abstract argumentation frameworks, where arguments are seen as abstract entities which are just investigated with respect to how they relate to each other, in terms of “attacks”, are nowadays well understood and deferent semantics (i.e., the selection of sets of arguments which are jointly acceptable) have been proposed. In fact, there seems to be no single “one suits all” semantics, but it turned out that studying a particular setting within various semantics and to compare the results is a central research issue within the field. Different semantics give rise to different computational problems, such as deciding whether an argument is acceptable with respect to the semantics under consideration, that require different approaches for solving these problems.

This broad range of semantics for abstract argumentation demands for a *unifying framework* for representing and reasoning with the various semantics. Such a unifying framework would allow us to see what the various semantics have in common, in what they differ, and ideally, it would offer generic methods for solving the computational problems that arise within the various semantics. Such a unifying framework should be general enough to accommodate most of the significant semantics, but simple enough to be decidable and computationally feasible. End quote

Properties (P1)–(P4) mentioned above hold for this case. See also the discussion in Section 5.1 above. Our own interpretation is object level and gives the argumentation network a meaning in

terms of strong negation N . As we mention in Section 5.1, the meta-level interpretation has to translate ‘ x attacks y ’ as is, by providing a predicate letter ‘ R ’ for attack and write xRy . The meta-level interpretation does not give a meaning to R . Our interpretation writes $x \rightarrow Ny$ for the attack of x on y . So when x and y are instantiated by, e.g. wffs $x = \alpha$ and $y = \beta$, we get a meaning for the attack of α on β , namely the meaning our logic gives to $\alpha \rightarrow N\beta$. The paper of Dvorak cannot write ‘ $\alpha R\beta$ ’, but even if it could do so, it would have to wait for us to say to Dvorak *et al.* what we mean by ‘ α attacks β ’ and then Dvorak *et al.* would try to say it formally in monadic second-order classical logic.

The other paper of Dvorak, paper [24], also provides a more propositional meta-language to describe more tightly higher level argumentation network. Properties (P1)–(P4) above still apply. The emphasis here is on (P3).

To quote the authors’ own words:

Begin quote:

This paper reconsiders Modgil’s Extended Argumentation Frameworks (EAFs) that extend Dung’s abstract argumentation frameworks by attacks on attacks. This allows to encode preferences directly in the framework and thus also to reason about the preferences themselves. As a first step to reduction-based approaches to implement EAFs, we give an alternative (but equivalent) characterisation of acceptance in EAFs. Then we use this characterisation to provide EAF encodings for answer set programming and propositional logic. Moreover, we address an open complexity question and the expressiveness of EAFs.

End quote.

The authors of [24] use propositional encoding of the argumentation network. This is still meta-level. To explain briefly what is the difference: while predicate logic would formalize ‘ x attacks y ’ as ‘ xRy ’, using a predicate R , the propositional encoding uses a propositional atom $r_{x,y}$. You need a new atom for each pair x,y . So $r_{x,y} = \top$ means that x does attack y and $r_{x,y} = \perp$ means that x does not attack y .

So for example to formalize:

- (For all x), if (x is ‘in’ iff all its attackers are ‘out’)

We can write in predicate logic:

- $\forall x [Q_{in}(x) \leftrightarrow \forall y (yRx \rightarrow Q_{out}(y))]$

or in propositional logic

- $\bigwedge_{x \in S} [x \leftrightarrow \bigwedge_{y \in S} (r_{x,y} \rightarrow \neg y)]$.

The advantage of the use of $r_{x,y}$ is that we can say directly that x does not attack y , i.e. we can write $\neg r_{x,y}$. Our logic CNN cannot do that. Although we can represent in our logic ‘ x attacks y ’ as ‘ $x \rightarrow Ny$ ’ and use it to calculate extensions, we cannot represent ‘ x does not attack y ’ as ‘ $\neg(x \rightarrow Ny)$ ’, because x may be out, i.e. $\neg x$ is true, and so $x \rightarrow Ny$ is true, even though x does not attack y . We shall remedy this in Part 2 of this article. Note also that using direct naming of attacks via xRy or $r_{x,y}$ we can also formalize higher level attacks of the form z attacks $r_{x,y}$.

So to summarize, paper [24] is also a meta-level interpretation geared towards supplying algorithms.

The paper of Doutre *et al.* [21] is also meta-level, using propositional logic. It is also geared towards algorithms. To quote the authors:

Begin quote:

We provide a logical analysis of abstract argumentation frameworks and their dynamics. Following previous work, we express attack relation and argument status by means of propositional variables and define acceptability criteria by formulas of propositional logic. We here study the dynamics of argumentation frameworks in terms of basic operations on these propositional variables, viz. change of their truth values. We describe these operations in a uniform way with in well known variant of propositional Dynamic Logic PDF: the Dynamic Logic of Propositional Assignments, DL-PA. The atomic programs of DL-PA are assignments of propositional variables to truth values, and complex programs can be built by means of the connectives of sequential and nondeterministic composition and test. We start by showing that in DL-PA, the construction of extensions can be performed by a DL-PS program that is parameterized by the definition of acceptance. We then mainly focus on how the acceptance of one or more arguments can be enforced and show that this can be achieved by changing the truth values of the propositional variables describing the attack relation in a minimal way.

End quote.

The paper of Grossi [38] is also meta-level, but it uses modal logic as the meta-level language. The schematic situation is described in Figure 34 by the embedding schema of $[E$ and $E^+]$ into $[S_2$ and $S_2^+]$ which in turn are embedded into $[T$ and $T^+]$. Here E is argumentation networks, S_2 is modal logic, S_2^+ is an expansion which contains extra connectives of modal logic to compensate for the lack of quantifiers. T is classical predicate logic. α is the extra ‘quantifier’ connectives imported by Grossi into modal logic to enable Grossi to interpret argumentation. The basic modal accessibility is taken by Grossi to be the inverse of the attack relation:

- x attacks y means x is an accessible world to y .

The situation is best described by Grossi’s own words below. You can immediately see that Grossi focusses on the (P4) aspects of the translation:

Begin quote:

The paper presents a study of abstract argumentation theory from the point of view of modal logic. The key thesis upon which the paper builds is that argumentation frameworks can be studied as Kripke frames. This simple observation allows us to import a number of techniques and results from modal logic to argumentation theory, and opens up new interesting avenues for further research. The paper gives a glimpse of the sort of techniques that can be imported, discussing complete calculi for argumentation, adequate model-checking and bisimulation games and sketches an agent for future research at the interface of modal logic and argumentation theory.

End quote.

The paper of Besnard, Doutre and Herzig, *Encoding argument graphs in logic* is a meta-level paper discussing properties (specification) which any meta-level interpretation should satisfy. For example, the interpretations of papers [21, 24, 25, 38] should be checked to see if they satisfy the principles outlined in paper [11]. Paper [11] is a meta-meta-level paper. We quote the authors’ own description of their paper:

Begin quote.

Argument graphs are a common way to model argumentative reasoning. For reasoning or computational purposes, such graphs may have to be encoded in a given logic. This paper

aims at providing a systematic approach for this encoding. This approach relies upon a general, principle-based characterisation of argumentation semantics.

In order to provide a method to reason about argument graphs [22], Besnard and Doutre first proposed encodings of such graphs and semantics in propositional logic [10]. Further work by different authors following the same idea was published later ... However, all these approaches were devoted to specific cases in the sense that for each semantics, a dedicated encoding was proposed from scratch. We aim here at a generalisation, by defining a *systematic* approach to encoding argument graphs (which are digraphs) and their semantics in a logic \vdash . Said differently, our objective is to capture the extensions under a given semantics of an argument graph in a given logic (be it propositional logic or any other logic). We hence generalise the approach originally introduced in [10] by parametrizing the encoding in various ways, including principles defining a given semantics.

We consider abstract arguments first, and then provide guidelines to extend the approach to structured arguments (made up of a support that infers a conclusion).

End quote.

9.5 Comparison with the paper [3] of Arieli and Caminada

To explain what Arieli and Caminada are doing and to evaluate and compare it with our article, let us start with the meta-level point of view of Dvorak [25] and the discussion in Section 5.1 above. We know that the Caminada labelling has three values $\{\text{in}, \text{out}, \text{und}\}$. The meta-level approach which interprets argumentation in monadic second-order classical predicate logic, would use variables x, y, z, \dots for arguments and the binary predicate R for the attack relation and three unary predicates Q_1, Q_0 and $Q_?$ for the values in, out and undecided respectively. See Section 5.1. The axioms of $\Delta(R, Q_0, Q_1, Q_?)$ say in predicate logic, among other things, that each x gets exactly one value. We are now ready to lead, step by step, from the above to the Arieli and Caminada paper.

Let us start from a given (S, R) . For each $a \in S$, consider $Q_1(a), Q_0(a)$ and $Q_?(a)$ as atomic propositions of the classical propositional calculus. Since these are all connected with the letter ‘ a ’ we can change notation and write

- a^+ for $Q_1(a)$
- a^- for $Q_0(a)$
- $a^?$ for $Q_?(a)$

Since we know that exactly one of the above can be true, we can forget about $a^?$ and use only the pair (a^+, a^-) . We have:

- If a is in, then we have $a^+ = \top, a^- = \perp$.
- If a is out we have $a^+ = \perp, a^- = \top$.
- If $a =$ undecided, we have $a^+ = \perp, a^- = \perp$.

We must add the restriction that we can never have both $a^+ = \top$ and $a^- = \top$.

We thus have turned the logic with $Q_1(x), Q_0(x), Q_?(x)$ into a 3-valued logic with values $\mathbf{t} = (1, 0), \mathbf{f} = (0, 1)$ and $\mathbf{?} = (0, 0)$, where these values mean as follows:

- $x = \mathbf{t}$ means x is ‘in’, or equivalently $x^+ = \top$ and $x^- = \perp$ or equivalently $(x^+, x^-) = (\top, \perp)$ or equivalently $Q_1(x) \wedge \neg Q_0(x) = \top$.
- $x = \mathbf{f}$ means x is ‘out’, or equivalently $x^+ = \perp$ and $x^- = \top$ or equivalently $(x^+, x^-) = (\perp, \top)$ or equivalently $\neg Q_1(x) \wedge Q_0(x) = \top$.

- $x=?$ means x is undecided or equivalently $x^+=x^-=\perp$ or equivalently $(x^+,x^-)=(\perp,\perp)$ or equivalently $\neg Q_1(x)\wedge\neg Q_0(x)=\top$.

The value $x^+=x^-=\top$ is forbidden.

We now have a propositional calculus with three values $\{\mathbf{t}, \mathbf{f}, ?\}$ and a reduction of each proposition x into two propositions x^+,x^- , such that $x=(x^+,x^-)$ and the restriction on the assignments on $\{x^+,x^-|x\in S\}$ which says that $x^+=x^-=\top$ is forbidden.

If we add to this version of classical propositional calculus quantifiers over propositions $\forall x, \exists x, x\in S$, we get the system of [3].

We can also turn the pairs (x^+,x^-) into a lattice calculus. Note that we have

$$x=(x^+,x^-) = (\text{say}) (x_1,x_2)$$

$$\neg x=(x^-,x^+)=(x_2,x_1).$$

But also we can write similarly for y and get:

$$x=Q_1(x)\wedge\neg Q_0(x)=(x_1,x_2)$$

$$y=Q_1(y)\wedge\neg Q_0(y)=(y_1,y_2).$$

Therefore

$$\begin{aligned} x\wedge y &= Q_1(x)\wedge Q_1(y)\wedge\neg Q_0(x)\wedge\neg Q_0(y) \\ &= Q_1(x)\wedge Q_1(y)\wedge\neg(Q_0(x)\vee Q_0(y)). \end{aligned}$$

We therefore get as a rule of the calculus

$$(x_1,x_2)\wedge(y_1,y_2)=(x_1\wedge y_1,x_2\vee y_2).$$

We get other rules in a similar fashion.⁴

Now that we have reduced the idea of [3] to the meta-level system of [25], we can read the quotation word by word from [3] and understand it in a new light. We have changed the notation slightly. [3] uses ‘ \perp ’ for undecided, we use ‘?’ so as not to confuse our readers.

Begin quote.

Three-valued Semantics and Signed Formulas. As indicated previously, our purpose in this paper is to provide a this, logic-based, perspective on argumentation frameworks, and to relate it to the two other points of view presented in the two previous subsections. [Gabbay note: these are 2.1: Extension based semantics and 2.2: labelling based semantics.] In this section we define the framework for doing so, using *signed theories*. Following [1], we introduce these theories in the context of three-valued semantics (see leo [4]).

Consider the truth values t (‘true’), f (‘false’) and $?$ (‘neither true nor false’). A natural ordering, reflecting differences in the ‘measure of truth’ of else elements, is $f < ? < t$. The meet (minimum) \wedge , join (maximum) \vee and the order reversing involution \neg , define by $\neg t = f$, $\neg f = t$ and $\neg ? = ?$, are taken to be the basic operators on \leq for defining eh conduction, disjunction, and the negation connectives (respectively) of Kleene’s well-known three-valued logic (see [42]). Another operator which will be useful in the sequel is defined as follows: $a \supset b = t$ if $\in \{f, ?\}$ and $a \supset b = b$ otherwise (see [2] for some explanations why this operator is useful for defining

⁴In Part 2 we shall compare with Gabbay’s 1985 (published in 2012 as [31]).

an implication connective). The truth tales of these basic connectives are given below.

\vee	t	f	$?$	\wedge	t	f	$?$	\supset	t	f	$?$	\neg	
t	t	t	t	t	t	f	$?$	t	f	f	$?$	t	f
f	t	f	$?$	f	f	f	f	f	t	t	t	f	t
$?$	t	$?$	$?$	$?$	$?$	f	$?$	$?$	t	t	t	$?$	$?$

The truth values may also be represented by pairs of two-valued components of the lattice $(\{0, 1\}, 0 < 1)$ as follows: $t = (1, 0)$, $f = (0, 1)$, $? = (0, 0)$. This representation may be intuitively understood as follows: If a formula ψ is assigned the value (x, y) , then x indicates whether ψ should be accepted and y indicates whether ψ should be rejected. As shown in the next lemma, the basic operators considered above may also be expressed in terms of this representation by pairs.

Lemma 11. Let $x_1, x_2, y_1, y_2 \in \{0, 1\}$. Then:

$$\begin{aligned} (x_1, y_1) \vee (x_2, y_2) &= (x_1 \vee x_2, y_1 \wedge y_2), \\ (x_1, y_1) \wedge (x_2, y_2) &= (x_1 \wedge x_2, y_1 \vee y_2), \\ (x_1, y_1) \supset (x_2, y_2) &= (\neg x_1 \vee x_2, x_1 \wedge y_2), \\ \neg(x, y) &= (y, x). \end{aligned}$$

In our context, the three values above are used for evaluating formulas in a propositional language \mathcal{L} , consisting of a set of atomic formulas $\mathbf{Atoms}(\mathcal{L})$, the propositional constants t and f , and logical symbols $\neg, \wedge, \vee, \supset$. We denote the atomic formulas of \mathcal{L} by p, q, r , formulas by ψ, ϕ and sets of formulas (theories) by \mathcal{T}, \mathcal{S} . the set of all atoms occurring in a formula ψ is denoted by $\mathbf{Atoms}(\psi)$ and $\mathbf{Atoms}(\mathcal{T}) = \{\mathbf{Atoms}(\psi) \mid \psi \in \mathcal{T}\}$ is the set of all the atoms occurring in the theory \mathcal{T} . Now, a *valuation* v is a function that assigns to each atomic formula a truth value from $\{t, f, ?\}$, and $v(t) = t, v(f) = f$. Any valuation is extended to complex formulas in the obvious way. In particular, $v(\psi \circ \phi) = v(\psi) \circ v(\phi)$ for every $\circ \in \{\neg, \wedge, \vee, \supset\}$. A valuation v *satisfies* ψ iff $v(\psi) = t$. A valuation that satisfies every formula in \mathcal{T} is a *model* of \mathcal{T} . The set of models of \mathcal{T} is denoted by $\text{mod}(\mathcal{T})$.

Definition 12. Let \mathcal{L} be a propositional language with set of atoms $\mathbf{Atoms}(\mathcal{L})$. A *signed alphabet* $\mathbf{Atoms}^\pm(\mathcal{L})$ is a set that consists of two symbols p^+, p^- for each atom $p \in \mathbf{Atoms}(\mathcal{L})$. The language over $\mathbf{Atoms}^\pm(\mathcal{L})$ is denoted by \mathcal{L}^\pm . A value v for \mathcal{L}^\pm is called *coherent*, if there is no $p \in \mathbf{Atoms}(\mathcal{L})$ such that both $v(p^+) = 1$ and $v(p^-) = 1$.
End quote.

We now compare with our system **CN**. We have a two world modal intuitionistic logic and our values are as follows

- $x = \text{in}$ means $x \wedge \neg Nx$
- $x = \text{out}$ means $\neg x \wedge Nx$
- $x = \text{undecided}$ means $\neg x \wedge \neg Nx$

If we let $x^+ = x$ and $x^- = Nx$, we get the system of Arieli and Caminada

- $x = \text{in}$ iff $(x^+, x^-) = (\top, \perp)$ iff $x \wedge \neg Nx = \top$
- $x = \text{out}$ iff $(x^+, x^-) = (\perp, \top)$ iff $\neg x \wedge Nx = \top$.

- $x = \text{und}$ iff $(x^+, x^-) = (\perp, \perp)$ iff $\neg x \wedge \neg Nx = \top$
- $(x^+, x^-) = (\top, \top)$ is not allowed. This means that $x \wedge Nx$ is not allowed. This is our intuitionistic restriction $\vdash Nx \rightarrow \neg x$.

To summarize the comparison, Arieli and Caminada turn argumentation into a Kleene 3 valued logic and manipulate it in the meta-level using Arieli’s [1, 2, 4] lattice theoretic methods. Our article manipulates it in the object level in a modal intuitionistic logic.

The perceptive reader might ask whether the paper of Arieli and Caminada carries any message or point of view, beyond the mathematical manipulation of truth values? In comparison, the message of our article is that it is object level, translating the attack relation xRy into $x \rightarrow Ny$, where N is strong negation embedded in a modal intuitionistic logic. What is the message of [3]? What can it be used for? Our answer to this question is that the Arieli–Caminada paper is very important in the context where a calculus of degrees of undecided is needed. If we need to build an argumentation method for an application area with many degrees of undecided, then we need a calculus for undecided values and we can follow the lead of [3].

Consider Figure 35. We need to be able to assign a value to x in Figure 35, given the variety of undecided values of the attackers of x , namely y_1, \dots, y_m .

Our system **CNN** is not general enough for this purpose. We are committed to the intuitionistic restriction, so we can have undecided values like $(\perp, \top, \top, \top), (\perp, \perp, \top, \top), (\perp, \perp, \perp, \top)$ and no more, for a 4-world modal model. The Arieli–Caminada approach can handle more, many more values, for example they can have $(\perp, \top, \perp, \top)$, etc.

Imagine a highly divergent group of witnesses of a tragic event. They are all undecided and contradicting and unreliable for a multitude of reasons. The traditional extension-based semantics, as well as the Caminada labelling based semantics will give them all undecided. This is *not* what we need. We need a calculus of undecided that will tell us something better. The equational approach [27] will give numbers, the modal provability [30] might give us several options for undecided, but these papers put forward fixed approaches derived from a fixed points of view serving some other methodological aims. The fact that they also yield several undecided values is only a side effect. Same applies to our logic **CNNk**. These system are not intended as calculi for undecided values and they offer no flexibility. We need an approach which can generate alternative calculi of undecided which can be tailored for different applications. The Arieli–Caminada paper is a start. We leave this research to a future paper.

It is regrettable that Arieli and Caminada took their paper in the (traditional) direction of generating extensions. It is time for all of us to get out of this way of thinking, which is nothing but a barrier to our imagination.

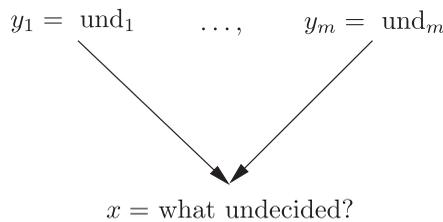


FIG. 35.

9.6 Comparing with paper [18]—Methods for solving reasoning problems in abstract argumentation. A survey

[18] is an excellent survey of methods for computing extensions. It divides the approaches into two:

- (1) Direct methods.
- (2) Reduction methods, where the argumentation systems are translated into other systems which have good computational facilities which can be used to calculate extensions.

So the paper [18] considers translations only as means of calculating extensions. In comparison, we consider translations as means of giving new and different meaning to the attack relation.

In the authors own words, for example, they say

Begin quote.

The underlying idea of the reduction approach is to exploit existing efficient software which has originally been developed for other purposes. To this end, one has to formalize the reasoning problems within other formalisms...In this approach, the resulting argumentation systems directly benefit from the high level of sophistication today's systems reached...In the remainder of Section 3 we shall present the concepts behind other reduction-based approaches, for instance, the equational approach as introduced by Gabbay in [27] and the reductions to monadic second order logic as proposed in [25].

We will not consider the vast collection of extensions to Dung's frameworks, like,⁵ value-based, bipolar, extended, constrained, temporal, practical, and fibring argumentation frameworks, as well as argumentation frameworks with recursive attacks, argumentation context systems, and abstract dialectical frameworks. We also exclude abstract argumentation with uncertainty or weights here.

End quote.

So to continue our comparison with paper [18], we highlight the fact that the interest of the authors of paper [18] in the current paper (as well as in papers [25] and [27], e.g.) is limited solely to the question of how these papers provide means to compute extensions.

10 Conclusion and future research

In this article, we introduced a modal logic **CNN** with strong negation N playing the role of modality. The semantics has two possible worlds, here (actual world t) and there (possible world s) with

$$\begin{aligned} t \models NA &\text{ iff } s \models \neg A \\ s \models NA &\text{ iff } t \models \neg A \end{aligned}$$

and with the restriction that for atoms q we have

$$t \models q \text{ implies } s \models q.$$

We showed how to translate any abstract argumentation frame \mathcal{A} into **CNN** via a theory $\Delta_{\mathcal{A}}$ of **CNN** in such a way that all the models of $\Delta_{\mathcal{A}}$ give exactly the complete extensions of \mathcal{A} .

Many properties and operations on \mathcal{A} (such as joint attacks, higher level attacks, bipolar argumentation, and more) become very simple when done in **CNN**.

⁵Editorial comment: we omit the references in this list

Further research postponed to Part 2 of this article includes the following topics:

- (1) Translate Assumption Based Argumentation (ABA) into CNN.
- (2) Use intuitionistic logic as a basis for CNN (call it INN) and see how to obtain intuitionistic based argumentation. In fact, given any logic L defined by rules of the type used in ASPIC, we can form LNN and translate ASPIC (for L) into LNN.
This will give us a better understanding of ASPIC and in fact also allow us to define a generalization of Assumption Based Argumentation.
- (3) Fully investigate ADF (Abstract Dialectical Frameworks) in CNN.
- (4) It is easy to define predicate argumentation using predicate CNN. Simply use predicate CNN and allow instantiations of any (S, R) with classical predicate wffs. Translate $\varphi \rightarrow \psi$ as $\varphi \rightarrow N\psi$. The semantics of predicate CNN will give us the complete extensions for the network.
- (5) Investigate bipolar networks and CNN, as discussed in CC7.
- (6) Investigate in detail iterated disjunctive attacks in CNN of the kind of Figure 31.
- (7) The logic CNN is based on two possible worlds t (actual world) and s (s is accessible to t). We can investigate the possibility of adding more worlds to the chain, say use 3 worlds t (actual), s (accessible to t) and s' (accessible to s).
Define

$$\begin{aligned} t \models NA & \text{ iff } s \models \neg A \\ s \models NA & \text{ iff } s' \models \neg A \\ s' \models NA & \text{ iff } t \models \neg A. \end{aligned}$$

This new system (call it CNN3) might allow us to do the following:

- (a) have several grades of undecided, see the comparison in Section 9.2
- (b) allow for attacks of a node x on another network A_x .
- (c) allow us to use truth in the middle node as a meta-level vehicle to express properties of extensions, (since NN refers to the actual world from the middle node). We can force the models to yield, e.g., only preferred extensions.
- (d) Investigate the connection with Logic Programming.
- (e) allow us to formalize in the object level the notion of ‘ x does not attack y ’.
- (8) Following the connection with Logic Programming and David Pearce’s Equilibrium Logic [45], we investigate whether Pearce’s logic can implement argumentation network similarly to the way CNN does.
- (9) Extend the negation as failure approach of [31] to argumentation and compare with this paper and with [3].

We shall investigate these possibilities in Part 2.

Acknowledgements

We are grateful to the referees and to Philippe Besnard, P. M. Dung, Davide Grossi, Beishui Liao, David Pearce, Odinaldo Rodrigues, Matthias Thimm and Leon van der Torre, for their valuable comments.

References

- [1] O. Arieli. Paraconsistent reasoning and preferential entailments by signed quantified Boolean formula. *ACM Transactions on Computational Logic*, **8**, Article 18, 2007.
- [2] O. Arieli and A. Avron. Reasoning with logical bilattices. *Journal of Logic, Language, and Information*, **5**, 25–63, 1996.
- [3] O. Arieli and M. W. A. Caminada. A QBF-based formalization of abstract argumentation semantics. *Journal of Applied Logic*, **11**, 229–252, 2013.
- [4] O. Arieli and M. Denecker. Reducing preferential paraconsistent reasoning to classical entailment. *Journal of Logic and Computation*, **13**, 557–580, 2003.
- [5] P. Baroni, F. Cerutti, M. Giacomin and G. Guida. Encompassing attacks to attacks in abstract argumentation frameworks. In *ECSQARU*, Springer, pp. 83–94, 2009.
- [6] P. Baroni, F. Cerutti, M. Giacomin and G. Guida. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, Vol. 52, 19–37, 2011.
- [7] P. Baroni, M. Giacomin and G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence*, **168**, 162–210, 2005.
- [8] H. Barringer, D. M. Gabbay and J. Woods. Temporal dynamics of support and attack networks: from argumentation to zoology. In *Mechanizing Mathematical Reasoning*, Springer, pp. 59–98, 2005.
- [9] H. Barringer, D. Gabbay and J. Woods. Temporal, numerical and meta-level dynamics in argument networks. *Argumentation and Computation*, **3**, 143–202, 2012.
- [10] P. Besnard and S. Doutre. Checking the acceptability of a set of arguments. In *Proceedings of NMR 2004*, AAAI Press, J. P. Delgrande and T. Schaub, eds, pp. 59–64, 2004.
- [11] P. Besnard, S. Doutre and A. Herzig. Encoding argument graphs in logic. In *Proceedings of the 15th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'2014)*, A. Laurent, O. Strauss, B. Bouchon-Meunier and R. R. Yager, eds, pp. 345–354. Springer, 2014.
- [12] P. Besnard, É. Grégoire and B. Raddaoui. A conditional logic-based argumentation framework. In *7th International Conference on Scalable Uncertainty Management (SUM'13)*, Springer, Lecture Notes in Computer Science (LNCS) 8078, pp. 44–56, 2013.
- [13] A. Bochman. Propositional argumentation and causal reasoning. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, Edinburgh, 2005.
- [14] G. Brewka. *Dialectical Frameworks*. Bologna, 2013.
- [15] G. Brewka, P. Dunne and S. Woltran. Relating the semantics of abstract dialectical frameworks and standard AFs. In *Proceedings of IJCAI-11*, AAAI press, pp. 780–785, 2011.
- [16] G. Brewka and S. Woltran. Abstract dialectical frameworks. In *Proceedings of the 20th International Conference on the Principles of Knowledge Representation and Reasoning (KR 2010)*, pp. 102–111, 2010.
- [17] M. Caminada and D. M. Gabbay. A logical account of formal argumentation. *Studia Logica*, **93**, 109–145, 2009.
- [18] G. Charwat, W. Dvořák, S. A. Gaggl, J. P. Wallner and S. Woltran. Methods for solving reasoning problems in abstract argumentation A survey. <http://dx.doi.org/10.1016/j.artint.2014.11.008>
- [19] C. Cayrol and M. C. Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *ECSQARU*. Lecture Notes in Computer Science, Godo, L. ed., Vol. 3571, pp. 37–389. Springer, 2005.
- [20] C. Cayrol and M. C. Lagasquie-Schiex. Coalitions of arguments: a tool for handling bipolar argumentation frameworks. *International Journal of Intelligence System*, **25**, 83–109, 2010.

- [21] S. Doutre, A. Herzig and L. Perrussel. A dynamic logic framework for abstract argumentation. In *International Conference on Principles of Knowledge Representation and Reasoning - KR 2014*, 20–24 July 2014 (Vienna, Austria), 2014.
- [22] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, **77**, 321–357, 1995.
- [23] P. M. Dung, R. A. Kowalski and F. Toni. Assumption-based argumentation. In *Argumentation in Artificial Intelligence*, G. Simari and I. Rahwan, eds, pp. 199–218, Springer US, 2009.
- [24] W. Dvořák, S. A. Gaggl, T. Linsbichler and J. P. Wallner. Reduction-based approaches to implement modgil’s extended argumentation frameworks. In *Brewka Festschrift*, LNAI 9060, T. Eiter et al. eds, pp. 249–264. Springer International Publishing, 2015.
- [25] W. Dvořák, S. Szeider and S. Woltran. Abstract argumentation via monadic second order logic. In *SUM 2012*, LNAI 7520, E. Hüllermeier et al. eds, pp. 85–98. Springer, 2012.
- [26] D. Gabbay. A theory of Disjunctive attacks in argumentation networks, to appear in *Logic Journal of IGPL*, OUP, 2015.
- [27] D. M. Gabbay. An equational approach to argumentation networks. *Argument and Computation*, special issue on the equational approach to argumentation, **3**, 87–142, 2012.
- [28] D. M. Gabbay. Fibring argumentation frames. *Studia Logica*, **93**, 231–295, 2009.
- [29] D. M. Gabbay. Semantics for higher level attacks in extended argumentation frames. Part 1: Overview. *Studia Logica*, **93**, 355–379, 2009.
- [30] D. Gabbay. Modal provability foundations for argumentation networks, *Studia Logica*, **93**, 181–198, 2009.
- [31] D. M. Gabbay. What is negation as failure, paper written in 1985, revised version in A. Artikis et al. eds, *Sergot Festschrift*, LNAI 7360, pp. 52–78. Springer, 2012.
- [32] D. M. Gabbay. The equational approach to CF2 semantics, short version. In *Proceedings COMMA 2012, Computational Models of Argument*, B. Verheij, S. Szeider and S. Woltran, eds, pp. 141–153. IOS Press, 2012.
- [33] D. M. Gabbay. Theory of semi-instantiation in abstract argumentation. To appear *Logica Universalis*.
- [34] D. M. Gabbay. *Meta-Logical Investigations in Argumentation Networks*. Research Monograph College Publications, 2013, 770 pp.
- [35] D. Gabbay and M. Gabbay. The attack as strong negation, Part II. *In Preparation*.
- [36] D. Gabbay and M. Gabbay. The attack as intuitionistic negation, Submitted to *Logic Journal of IGPL*, OUP.
- [37] D. M. Gabbay and O. Rodrigues. Probabilistic argumentation, an equational approach. To appear in *Logica Universalis*, arxiv.org/abs/1503.05501.
- [38] D. Grossi. Argumentation in the view of modal logic. In *ArgMAS 2010*, LNAI 6614, P. McBurney, I. Rahwan, and S. Parsons eds, pp. 190–208, Springer, 2011.
- [39] D. D. Hanh, P. M. Dung and P. M. Thang. Inductive defense for Modgil’s extended argumentation framework. *J Logic Computation* doi: 10.1093/logcom/exq018. First published online: June 16, 2010.
- [40] A. Hazen. Comments on the logic of constructible falsity (strong negation). *Bulletin of the Section of Logic*, **9**, 10–13, 1980.
- [41] A. Kakas, F. Toni and P. Mancarella. Argumentation logic. In *Proceeding COMMA 2014*, S. Parsons, N. Oren, C. Reed and F. Cerutti, eds, pp. 345–356. DOI10.3233/978-1-61499-436-7-345
- [42] S. C. Kleene. *Introduction to Metamathematics*. Van Nostrand, 1950.

- [43] S. Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, **173**, 901–934, 2009.
- [44] S. H. Nielsen and S. Parsons. A generalisation of Dung’s abstract framework for argumentation: arguing with sets of attacking arguments. In *LNCS Vol. 4766*, pp. 54–73. Springer.
- [45] D. Pearce. Equilibrium logic. *Annals of Mathematics and Artificial Intelligence*, **47**, 3–41, 2006.
- [46] H. Prakken. An abstract framework for argumentation with structured arguments, *Argument and Computation*, **1**, 93–124, 2011.
- [47] S. Villata, G. Boella, D. M. Gabbay and L. van der Torre. Modelling defeasible and prioritized support in bipolar argumentation. *Annals of Mathematics and Artificial Intelligence*, **66**, 163–197, 2012.
- [48] S. Villata, G. Boella and L. van der Torre. Attack semantics for abstract argumentation. In *22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, IJCAI/AAAI, pp. 406–413, 2011.

Received 1 April 2015

Appendix: Formal definition of the modal logic CNN

In this section, we change the syntax of CNN slightly. Besides the classical connectives and the connective N , we add a constant t for the actual world. This helps with completeness proofs. We should call, if we want to be strict, the modified system with the slightly different name, say, CNN*, but we are not going to bother.

DEFINITION A.1

A *model* for CNN is a pair $\langle \{1, 2\}, \nu \rangle$ where ν assigns subsets of $\{1, 2\}$ to atomic propositions such that if $1 \in \nu(p)$ then $2 \in \nu(p)$

DEFINITION A.2

If $M = \langle \{1, 2\}, \nu \rangle$ is a model of CNN then define truth in M as follows, (where $m = 1, 2$ and t is the propositional constant for world 1).

- $M \models_m p$ iff $m \in \nu(p)$
- $M \models_m \top$
- $M \not\models_m \perp$
- $M \models_m t$ iff $m = 1$
- $M \models_m \neg A$ iff $M \not\models_m A$
- $M \models_m A \wedge B$ iff $M \models_m A$ and $M \models_m B$
- $M \models_m A \vee B$ iff $M \models_m A$ or $M \models_m B$
- $M \models_m A \rightarrow B$ iff $M \not\models_m A$ or $M \models_m B$
- $M \models_m NA$ iff either $m = 1$ and $M \not\models_2 A$, or $m = 2$ and $M \not\models_1 A$

Write $M \models A$ when $M \models_m A$ for all m (i.e. $m = 1$ or $m = 2$); write $\models_m A$ when $M \models_m A$ for all M ; and write $\models A$ when $M \models_m A$ for every m and every M .

Intuitively, a model is a finite linear ordering of worlds—in the sense of familiar possible world semantics for modal logic—where logical connectives are defined as usual and an atomic proposition p is true at a world, then it is true at all ‘later’ worlds. The new connective N has the semantics that NA is true at a world if and only if A is false as the ‘next’ world where, in the case of interpreting

N , ‘next’ cycles back to the first world at the last world (so NA is true at the last world when A is false at the first).

The condition on atomic propositions ensures that N is a kind of negation, at least for atomic propositions interpreted at the first world. If $M \models_1 Np$ then $M \not\models_2 p$ and so $M \not\models_1 p$. Moreover the condition on interpreting N ensures that $M \models_1 NNp \rightarrow p$ as $M \models_1 NNp$ implies $M \not\models_1 Np$ implies $M \models_1 p$.

DEFINITION A.3

The theorems of **CNN** are defined as follows:

$$\begin{array}{ll} \text{tautologies} & \\ (K) \quad N(A \wedge B) \leftrightarrow NA \vee NB & (A) \quad t \rightarrow (p \rightarrow N\neg p) \\ (F) \quad \neg NA \leftrightarrow N\neg A & (T) \quad t \leftrightarrow Nt \\ (C) \quad A \rightarrow NNA & \end{array}$$

$$\frac{A \quad A \rightarrow B}{B} (MP) \quad \frac{A}{N\neg A} (N)$$

For a set Γ we write $\Gamma \vdash A$ when there are $A_1 \dots A_n \in \Gamma$ such that $\vdash (A_1 \wedge \dots \wedge A_n) \rightarrow A$.

LEMMA A.4

$\vdash A \rightarrow B$ implies $\vdash NB \rightarrow NA$, and $\vdash A \leftrightarrow B$ implies $\vdash NA \leftrightarrow NB$,

PROOF. The first part follows by the following derivation:

$$\begin{array}{ll} \vdash A \rightarrow B & \text{assumption} \\ \vdash \neg(A \wedge \neg B) & \text{tautologies and (MP)} \\ \vdash N\neg\neg(A \wedge \neg B) & (N) \\ \vdash N(A \wedge \neg B) & \text{tautologies, (MP) and (F)} \\ \vdash NA \vee N\neg B & (K) \\ \vdash NA \vee \neg NB & \text{tautologies, (MP) and (F)} \\ \vdash NB \rightarrow NA & \text{tautologies and (MP)} \end{array}$$

It is worth expanding one of the steps further: $\vdash N\neg\neg C$ implies by (F) that $\vdash \neg N\neg C$; also, $\vdash \neg NC \rightarrow N\neg C$ is a consequence of (F). So, using tautologies and (MP), $\vdash \neg N\neg C \rightarrow \neg\neg NC$ thence $\vdash \neg N\neg C \rightarrow NC$. Therefore $\vdash N\neg\neg(A \wedge \neg B)$ implies $\vdash N(A \wedge \neg B)$.

The rest of the lemma follows from the fact that $A \leftrightarrow B$ is short for $(A \rightarrow B) \wedge (B \rightarrow A)$. ■

THEOREM A.5

$\vdash B \leftrightarrow C$ implies $\vdash A[p/B] \leftrightarrow A[p/C]$

PROOF. By induction on A using A.4. ■

THEOREM A.6

$\models A$ iff **CNN** $\vdash A$

PROOF. For the right-left direction it is a simple matter to verify, by induction on derivations, that **CNN** $\vdash A$ entails that for any M and any m , $M \models_m A$.

For the right left direction we argue that if Σ is consistent then there is an M such that, for every $A \in \Sigma$, $M \models_m A$ for $m = 1$ and $m = 2$. First, since **CNN** $\not\vdash A$ then using familiar methods, we can find a

set Γ that is consistent with respect to **CNN** and maximal in the sense that $B \in \Gamma$ or $B \notin \Gamma$ for every B . Moreover, we have that $\Gamma \vdash A$ implies $A \in \Gamma$

Now, consider the set $\Delta = \{B \mid N\neg B \in \Gamma\}$. If $B, \neg B \in \Delta$ then both $N\neg B \in \Gamma$ and $N\neg\neg B \in \Gamma$, and so by A.5, (F) and tautological reasoning, $\neg NB, NB \in \Gamma$ which is impossible as Γ is consistent. Also, since $\neg NB \in \Gamma$ or $NB \in \Gamma$, it follows that $N\neg B \in \Gamma$ or $NB \in \Gamma$, and so $B \in \Delta$ or $\neg B \in \Delta$. We conclude that Δ , like Γ , is maximal and consistent.

Now, by the definition of Δ and the fact it is maximal consistent, $NB \in \Gamma$ implies $B \notin \Delta$; and if $NB \notin \Gamma$ then $N\neg B \in \Gamma$ and so $B \in \Delta$. Also, if $B \in \Gamma$ then, by (C), $NNB \in \Gamma$ and so $N\neg\neg B \in \Gamma$ which implies that $\neg NB \in \Delta$ thence $NB \notin \Delta$; and conversely, if $B \notin \Gamma$ then $\neg B \in \Gamma$ and so $NN\neg B \in \Gamma$ which implies $N\neg NB \in \Gamma$ thence $NB \in \Delta$. We conclude that

$$(\dagger) \quad B \in \Gamma \text{ iff } NB \notin \Delta \quad \text{and} \quad B \in \Delta \text{ iff } NB \in \Gamma$$

We now describe a model M using Γ and Δ . Suppose that $t \in \Gamma$, and set $v(p) = \{1, 2\}$ iff $p \in \Gamma$ and $v(p) = \{2\}$ iff $p \in \Delta$. Then, as $t \in \Gamma$, if $p \in \Gamma$ then by (A), $N\neg p \in \Gamma$ and so $p \in \Delta$. Thus M is indeed a model as defined in A.1.

We now verify, by induction on A , that $M \models_1 A$ iff $A \in \Gamma$ and $M \models_2 A$ iff $A \in \Delta$

- If A is atomic the result follows by the definition of M .
- If A is t then, by assumption, $t \in \Gamma$. Moreover, making use of (T), $t \in \Gamma$ iff $Nt \in \Gamma$ iff $\neg Nt \notin \Gamma$ iff $N\neg t \notin \Gamma$. It follows that $t \in \Gamma$ iff $t \notin \Delta$.
- If A is a truth functional connective (i.e. is not N), then the result follows easily by the induction hypothesis and the maximal consistency of Γ and Δ .
- Suppose A is NB . Then using (\dagger) above we have:

$$\begin{array}{ll} M \models_1 A \text{ iff } M \models_1 NB & M \models_2 A \text{ iff } M \models_2 NB \\ \text{iff } M \not\models_2 B & \text{iff } M \not\models_1 B \\ \text{iff } B \notin \Delta & \text{iff } B \notin \Gamma \quad \text{ind. hyp.} \\ \text{iff } NB \in \Gamma & \text{iff } NB \in \Delta \quad (\dagger) \\ \text{iff } A \in \Gamma & \text{iff } A \in \Delta \end{array}$$

This completes the induction.

The argument is similar if we suppose $t \in \Delta$. Therefore, if Σ is consistent, then there is a model M such that $M \models A$ for all $A \in \Sigma$. ■

THEOREM A.7

If A does not contain t , nor any subformula of the form NB unless B is atomic, then $\text{CN} \vdash A$ iff $\text{CNN} \vdash t \rightarrow A$.

PROOF. Given A.6, $\text{CNN} \vdash t \rightarrow A$ iff $\models_1 A$.

For any M , the function h over the atomic formulae of **CN** such that:

$$h(p) = 1 \text{ iff } M \models_1 p \quad \text{and} \quad h(Np) = 1 \text{ iff } M \models_1 Np$$

is clearly a truth assignment satisfying item 4 of 2.1, and moreover by an easy induction on A , $M \models_1 A$ implies $h \models A$ (as $h \models \dots$ is characterised in 2.1).

On the other hand, if h is a truth assignment according to 2.1, define v – and so M – as follows:

$$v(p) = \begin{cases} \{1, 2\} & \text{if } h(p) = 1 \\ \emptyset & \text{if } h(Np) = 1 \\ \{2\} & \text{otherwise} \end{cases}$$

(note that the same numbers have different functions here: the 1 of the assertion $h(p)=1$ acts as a propositional truth value of 2.1; the 1 of the assertion $v(p)=\{1,2\}$ acts as a possible world). An easy induction on A verifies that $h \models A$ implies $M \models_1 A$.

Therefore, there is an h such that $h \models A$ iff there is an M such that $M \models_1 A$. It follows from this that $\mathbf{CN} \vdash A$ iff $\mathbf{CNN} \vdash t \rightarrow A$. ■