Argumentation-based Semantics for Attack-Defense Networks

D. M. Gabbay, R. Horne, S. Mauw, and L. van der Torre

University of Luxembourg

Abstract. This paper connects between the areas and communities of abstract argumentation and attack-defense trees in the area of security. Both areas deal with attacks and defense/support and both areas rely on human applications dealing with human aggressive activities. Our idea is to examine the different points of view of these two communities and come up with new unifying ideas which can solve existing problems in each area.
The unifying idea we use in this paper is to regard the attacking argument as collections of wepons. The definition of a basic network which is acceptable for both communities is a pair $(\mathbf{S}, \mathbf{R})$, where **S** is a non-empty set of nodes and **R** is a binary relation on **S** (the basic "attack" relation). Each node **x** in **S** is a multi-set which is a collection of weapons. A node **x** attacking a node **y** needs to use its weapons to neutralize all of the weapons of the target. So we need another relation on the weapons to tell us what can neutralize what. This technically means we need another network saying what are the weapons which can be used and how they can attack other weapons. It also brings numerical annotations of costs and firepower/intensity into our considerations. Note that this other network is not exactly an argumentation network. A collection of weapons needs not be conflict free. Think of a collection of medicines. Some cannot be given together but you have them all available to combat various attacks on your health.

1 Background and Orientation

This section deals with orientation towards our semantical and syntactical study of attack and defense networks in the area of security, safety and fault tolerance. We begin with Table 1 listing some of the major areas of research and communities where Attack and defense networks are relevant and where they are either already used in some form or should be used. Included are our comments on the state of play in these areas as well as our expectations for these areas and future research and contributions to logic of these areas in the 21st century. This will give us a context for what we want to do in this paper, and we then develop better networks in the area of security safety and fault tolerance.

For now let us motivate and explain what we our doing in this paper. Our approach is to compare existing treatments of attack and defense networks in the security area, with existing treatments of such networks in the argumentation area. We hope to export ideas and technical tools from the argumentation area into the security area.

As mentioned in Table 1, the security area can benefit from (among other things) the following.

1. Ways of Handling loops.

Table 1. Table of major research areas and communities which can benefit from theories of attack and defense networks.

Area	**Comments**
1. Attack and defense trees/networks in security and safety areas. (AD networks)	No handling of loops. No temporal aspects. Need improved semantics.
2. Argumentation	Need better temporal models. Can be too mathematical and inwards looking.
3. Ecology	Emphasis on loops and dynamic equilibrium. Strong community well worth working with.
4. Medical	Very complex area. No network research. The modelling is similar to AD networks.
5. Legal (logic)	Not similar to any of the other areas. This area is well developed and the researchers in the area are working very well with the argumentation community.
6. Sex offenders, therapy etc.	Similar to AD networks. There is no awareness of logic or of network possibilities in this community.
7. Numerous other areas.	Natural language, informal logic, argument mining, fallacies, etc., not directly related to the present paper.

2. Ways of integrating temporal aspects
3. Improved affine linear logic semantics addressing resource limitations
4. Addressing the role of failed attacks or attacks which only weaken the target.

We take as a starting point the material of reference [1] especially Figure 1 of [1]. We study this figure and compare it, bit by bit with argumentation networks, and try to see how to understand it in a new improved more detailed point of view.

Let us begin:

Viewed as a bipolar argumentation network (i.e., a network with attack and support) this Figure 1 has the following characteristics.

1. The graph has no cycles. (The handling of cycles is still an open problem in Attack and Defense context and is more central in the argumentation context.)
2. The graph has a single top node (let us call it the goal g) to be defended and it is layered as a tree with layer 1 defending/protecting g and each later $n+1$ attacking the previous layer n and or defending layer $n-1$.
3. The graph uses joint attacks and/or joint defense/support.
4. The nodes have internal meaningful contents. They are not atomic letter nodes. This should be taken into account when offering semantics for the tree.

There are several ways of looking at Figure 1.

1. As a traditional formal argumentation network. This is not the right view, as we shall discuss.

2. As a graph for a game between two players (the defender/protector of g and the attacker of g) the levels/layers are moves and countermoves of the players. This view is better but still not exactly right. We shall also discuss this. The graph can be flattened to a mini-max matrix. The defence can put forward in layer 1 all possible best strategic defense moves and the attacker can attack all possible attacks and the net result is the solution. The problem with this view is that we need to address more features of the application, for example the temporal evolution of moves, the availability and cost of resources and the local reasoning and aim of each player and the treatment of cycles.
3. As action counter action temporal sequence between two agents, the one protecting g and the other in principle attacking g. This is a much better view but it needs to be fine-tuned to various applications, Figure 1 being one of them and Figure 8 is another.

We now ask how do we proceed, and where do we find the connection and use of argumentation in the attack and defense trees context?

We can choose to follow one of two approaches.

*1. Present a general theory of networks of trees, encompassing in some way both elements/components of argumentation and properties of attack and defense trees and fine tune the general theory and zero in on our own more specific theory with examples.
*2. Start with examples from both areas and step by step, using a Socratic method, add components and generalise to the specific theory we want to present.

We first follow approach *2, which is more educational for the general community of practitioners. However, the general approach *1 is more logical and methodological and should also be presented, and the formal mathematical presentation of this we leave to the appendices.

Let us now look at formal argumentation networks and find a network familiar from the formal argumentation community, (Figure 2) which may be, on the face of it, similar to what Figure 1 seems to be. We then continue our analysis of Figure 1. Consider Figure 2. In this figure we use a single arrow for support "$\rightarrow$" and a double arrow for attack "$\twoheadrightarrow$". To start our comparison, the nodes in Figure 2 are explained and exemplified by nodes in Figure 1 in paraenthesis below.

Explanation of the nodes of Figure 2:

– g is the goal to protect (data confidentiality)
– a, b, c are supports (physical security, network security, etc.)
– α, β, γ attack the support (break in, dictionary attack, corruption)
– x, y, z support a, b, c by attacking the attacks (security guard, strong password, etc.).

Comparing Figure 1 and Figure 2, let us make some observations.

Observation 1. In the Figure 1 of the attack and defence paper, consider the subpart of the figure represented by Figure 3. In this figure the node y does not attack β in the sense of "killing" β but makes b stronger so that it can withstand the attack of β.

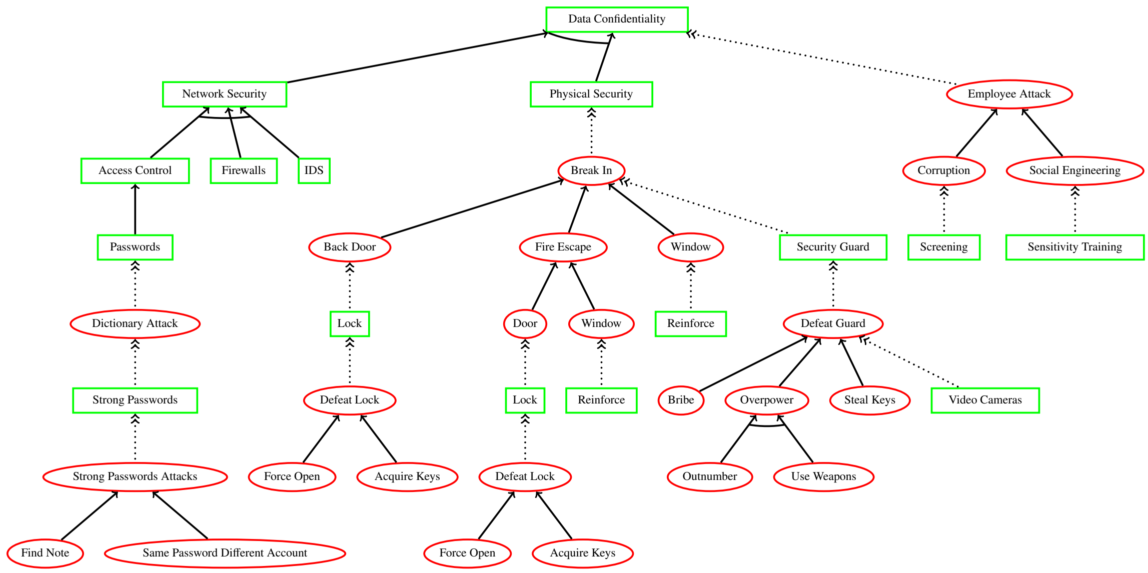


Fig. 1. An ADTree for protecting data confidentiality of reference [1]

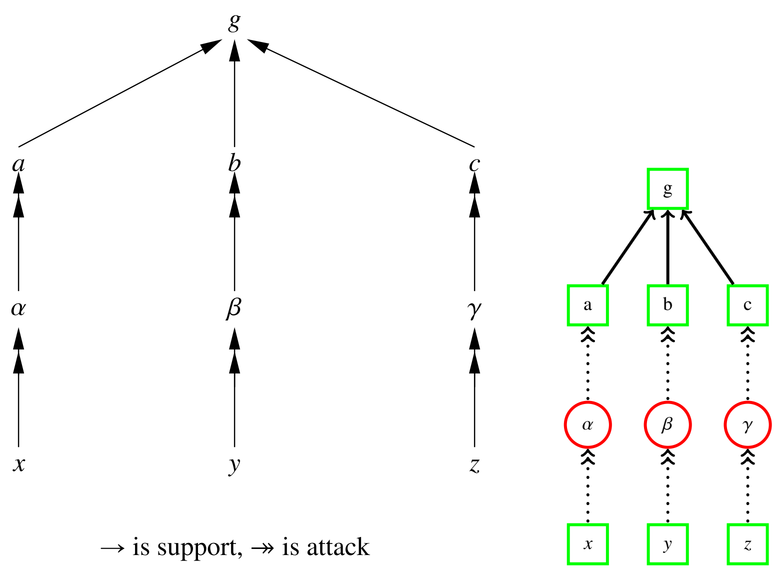


Fig. 2. A scenario with goals, attacks and defenses, in terms of attack relations and as an attack defense tree.

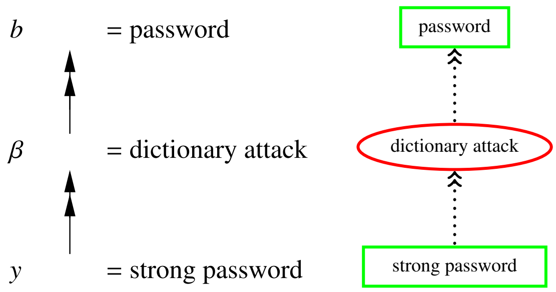


Fig. 3.

In other words, the part of the figure (namely the formal attack and defense subfigure Figure 4) is really the formal argumentation figure, Figure 5.

Figure 5 represents a bipolar argumentation network, that is a network with attack and defense (in Argumentation terminology). One of the interpretations of such networks, from the argumentation point of view, is that to attack and kill a node b, we need also to kill all of its supporters (i.e. we need to attack y as well). In the terminology of Gabbay's paper [4], the set $\{b, y\}$ forms a Support Group. Indeed this is also the security view of the the attack and defense Figure 1, in that the attacks must continue on node $y =$ strong password. Indeed in Figure 1, $y =$ strong password is attacked by "Strong password Attacks" (i.e. Find Note, Same Password Different Accounts).

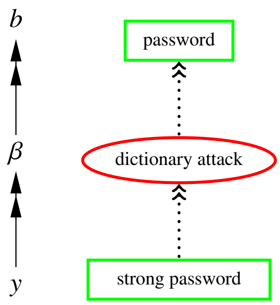


Fig. 4.

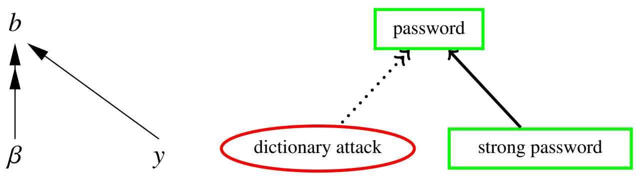


Fig. 5.

Observation 2. On the other hand, the part of the figure with a, α, x with the additional options of u and w as in Figure 6 below is different, it has additional new features:

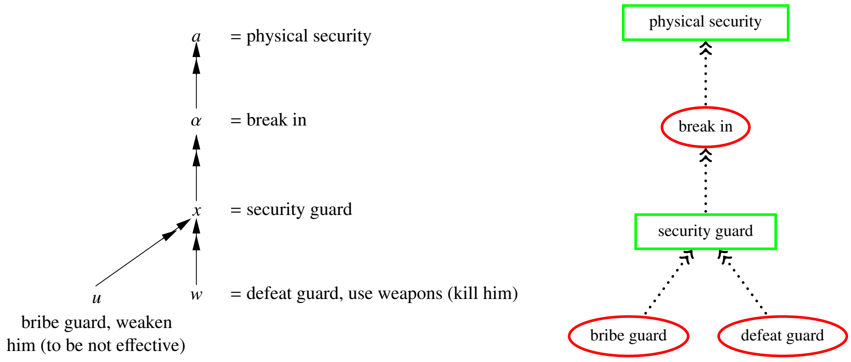


Fig. 6.

The rewrite is the following Figure 7

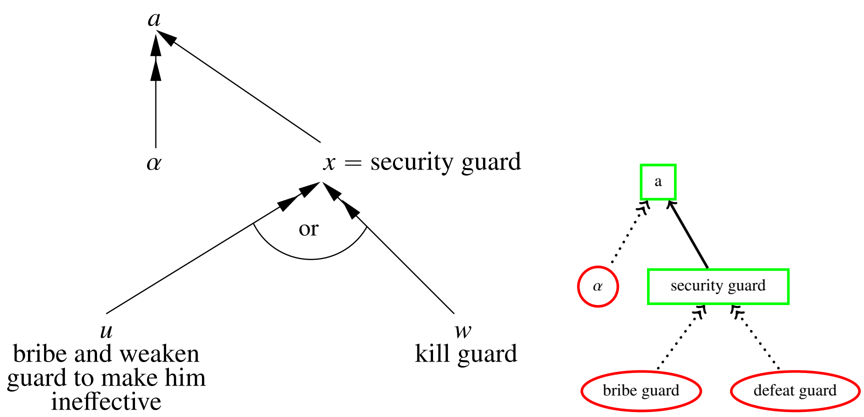


Fig. 7.

To understand the difference between "killing" and "weakening", consider the following[1]:

Observation 3. Consider the medical network, Figure 8.

This Figure is a medical case. It is similar to the data case, we have Dov's health to protect. Dov is attacked by Cholesterol and in turn Cholesterol is attacked by Statins. But Statins do something new; they activate Cramps as a side effect.

[1] In argumentation networks, If a node x attacks a node y (i.e. $x \twoheadrightarrow y$), then if x is alive then the attack on y is always successful and x kills y and y is dead. There is no intermediate result such as weakening y or failing to kill y.

We do not have side effects in Figure 1 but it is possible to add examples of side effects. In Figure 6, killing the guard may activate a Murder Investigation as a side effect and we might not want that.

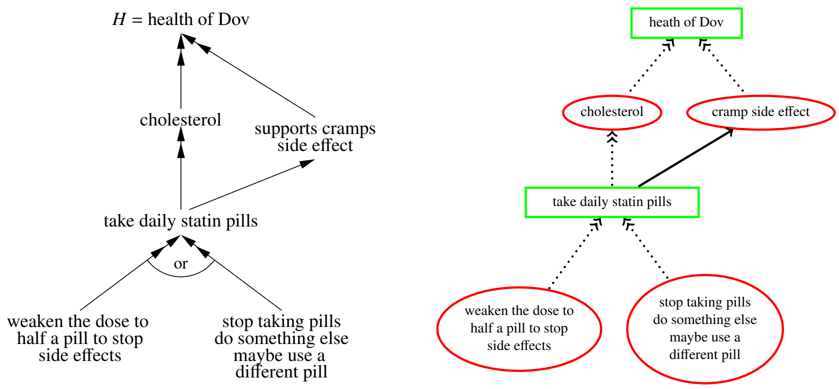


Fig. 8.

Observation 4. In formal argumentation networks, a node x attacking several targets attacks them all of them in the same way. There is no option for different attacks for different targets. This is not the case in Figure 1, "defeat lock" attacking the back door is most likely not the same as the one attacking the front door. The attacks are directional.

Observation 5. In Security there is heavy stress on resources, hence the use of Linear Logic in the attempted semantics. Formal argumentation is based on classical logic.

Remark 1 (Summary of discussion in Section 1). We summarise the points learnt from our discussion in this section.

To give good argumentation like semantics for Figure 1 of security, we need to enrich argumentation with the following features:

1. And/or attacks and defence (this we have already).
2. Allow converting attack to support and support to attacks (Dov Gabbay [3] in 2005 did this but for only the numerical case).
3. Allow for weakening attacks (as well as attacks which fail) in a directional way. (This means that for the same live x and different targets get y, say for example, $x \twoheadrightarrow y_1, x \twoheadrightarrow y_2$, and $x \twoheadrightarrow y_3$, and the attack of x on y_1 will succeed, the attack on y_2 will fail and the attack on y_3 will only weaken y_3. Compare this with numerical attacks which change the strength of the target by a numerical factor.)
4. Allow for a mechanism of replacing (i.e. activating) nodes when attacked by other nodes in the object level, i.e. Figure 9.

The notation we use in Figure 9 is "$\multimap$" for activation.

So nodes can attack other nodes, can support other nodes, or can activate other nodes. This means that our networks have three arrows, for attack, support and activation and that nodes can be either

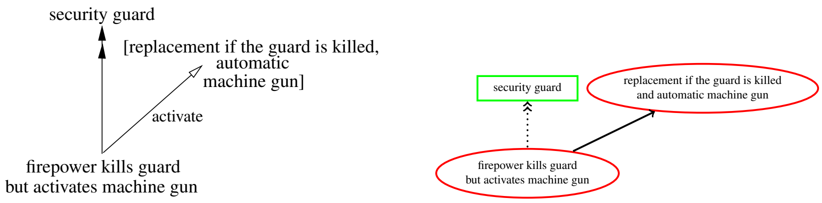


Fig. 9.

- alive and active
- alive but not active
- dead (and cannot be activated).

Alive and active nodes can attack other nodes and can activate other nodes provided the other nodes are alive (but not active).[2]

5. Deal with side effects in the formal argumentation level, because in practice for example when you hack into a server you may cause unexpected side effects.
6. We need one more principle: Consider Figure 10, where we have nodes $a_1, \dots, a_n$ supporting g. To make sure we successfully kill g we need to kill *all* of $a_1, \dots, a_n$. This is for the case where all the a_i are independent supports.

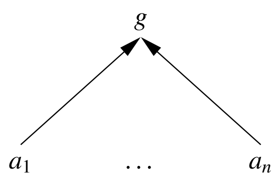


Fig. 10.

If we are dealing with joint supports, as in Figure 11, where each a^i is a set $\{a^i_1, a^i_2, \dots\}$, then to kill g we must kill at least one member from each $\{a^i\}$.

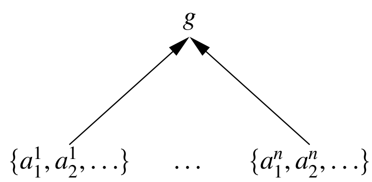


Fig. 11.

[2] Activation in argumentation is thoroughly studied in our paper 616, reference [8].

This is not like how it goes in logical and legal argumentation. If we have

$$\begin{array}{c} a_1 \vdash g \\ \vdots \\ a_n \vdash g \end{array}$$

Then attacking or falsifying all a_i does not mean that g is false. There may be some new $x \vdash g$.

7. **Running Global Side effects.** Each node costs money. Guards need to be paid, Keys need to be acquired, etc., etc. We have a global budget node which needs to be treated as a special weapon node.
8. Local support. This principle has to do with supporting local nodes in the middle of the tree. We note that in Figure 1 all the support nodes actually support the security of the data. There is a sequence of nodes.
   Break in through door $\Leftarrow$ lock door $\Leftarrow$ acquire keys.
   So let us add support to acquire key the support we add is " increase budget to buy keys".
   This support is not for server security, it supports locally the attack of acquire keys.
9. The emphasis on resources requires the use of Linear Logic in our models.

2 Introductory schematics towards a formal Model

2.1 General schematic view

Let us address the features we have observed in Section 1, Remark 1, and give a quick schematic model for a new possible formal attack and defense network.

Our starting point is the server security Figure 1 and the medical example of say Figure 8.

What is common to these two cases is the following:

1. We have in the medical case two players, Doctor and Nature and a patient. Nature wants to attack the patient with a multitude of illnesses, viruses, diseases, etc.
   The doctor wants to protect the patient with a wealth of medicines, inoculations, antibiotics, exercises, etc.
2. In the security case we have two players, security and hackers and the data. Security wants to protect the data and hackers want to acquire the data.
3. We regard the two players as part of a simulation. Neither the Doctor nor the security officer wait for the patient or data to be successfully attacked before reacting with a counter attack.
   They do not even necessary assume the attacks will be successful. We may counterattack just because we are being attacked. In argumentation every live attack is always successful. So from the argumentation point of view the answer to the question of whether the data is compromised in Figure 1 may be different from the answer given by the security point of view.

Let us in this subsection 2.1, take a schematic view of how we are going to model these two cases. By a schematic view we mean a map overview of components and their functionality without going into detail. More details in subsection 2.2.[3]

What is common schematics to the two cases is as follows:

1. goal g = (patient or data)
2. protecting player (doctor and resp. security company) and attacking player (nature and resp. hackers))
3. A set of atomic "arguments" $S = \{a, b, c, \ldots\}$ which are what the attackers and protectors use as the weapons.
   It can be viruses and medicines or security measures or hacking and attack measures. Some of the weapons are attacking weapons which destroy other weapons, some are defensive weapons, used to protect or boost the capabilities of other weapons and some are mixed, weapons which can do both.
4. The basic relations $R_a \subseteq S \times S$ (attack) and $R_s \subseteq S \times S$ (support). The exact properties of these relations is not relevant to this subsection, since we are giving a general schematic structural view. Note we are simplifying and not including the activation relation. The detailed treatment of activation is complex and relatively recent, see [8].
5. We note that (S, R_a, R_s) is what is known in formal argumentation theory as "bipolar network", i.e. a network with attack R_a and support R_s. For bipolar networks see Gabbay's [4].
   We must note that the weapons network cannot be fully viewed as an argumentation network. The reason is that a key principle in the investigations (by the argumentation community) of argumentation networks is the requirement for a set of arguments to be conflict free. This is not a reasonable requirement for a set of weapons. We can stock for examples two different medicines which are incompatible and can kill a patient if given together.
6. The players have 'logic" as a guide and they choose each a subsets $E_1, E_2 \subseteq S$ of "arguments", (which may not be conflict free), they are going to use. In formal argumentation there are several options for this kind of choice (called semantical extensions). For example, a doctor might give the patient travelling to, say, Africa the most common inoculations and antibiotics to fend off all the types of diseases he might encounter (including a variety of emergency capsules to be taken if needed, with a warning of what is not compatible with what). The choice is based on cost, convenience and logical judgement on what the patient is likely to encounter. If, for example, the patient is from UK and the doctor represents a travel insurance company then the the choice may be just enough protection to keep the patient OK until he is flown back to UK and becomes the responsibility of the UK NHS.
   In the case of security there are cost considerations and expectations of what kind of attacks to protect against. In Figure 1, i.e. the case of security the protector might choose level 1 of Figure 1. This is the top green layer of the Figure. These are the items {physical security, passwords, firewalls and IDS}. These seem to be

[3] It is like buying a flat which has not been built yet, but you are sold it on a map. For example: a kitchen, 3 bedrooms, 3 bathrooms, lounge, study, etc. You are not given details about what is in the kitchen or in the bathrooms, etc. This comes later.

standard. Level 2 is the red attackers and the attackers seem to have decided to break in physically and take the actual server and the weak points are the back door fire escape and window. This was probably not anticipated by the protector, otherwise the protector would have already reinforced the window and put special locks on the door already in the first level.[4]

The perceptive reader can see already how we view the layers in the figure. We see the process as a simulated temporal progression (not real time but imaginary) sequence of action and counter actions of the players.

2.2 Schematic example semantical model

This subsection gives a schematic sample model to show how the final model is going to be constructed in Section A. We will explain, in a Socratic way, the principles and the assumptions made.

Component 1. **Borrow from formal argumentation**. Let (S, R) be a formal argumentation network. This means $S \neq \varnothing$ is a set of abstract formal arguments and $R \subseteq S \times S$ is the attack relation. We are not including the support relation to keep it all very simple. Note however that our handling of sets of arguments does not necessarily require that they are conflict free.

It is helpful to keep a real argumentation (where conflict freeness is key) example interpretation in mind.

Imagine John and Mary are planning a wedding. They want to decide who to invite. They make a comprehensive list of all the people they might invite. These include: relatives, friends, colleagues, neighbours, etc., etc. It is clear immediately that the set of invited guests to the wedding must be conflict free.

There is a problem. They know, for example, that if they invite x, they cannot invite y. This is the attack relation xRy. To make the possible conflict more specific, let us imagine that John's father may have divorced John's mother and married his long-time mistress. So on the one hand they would like to invite both — the current mistress/wife to John's father and also to invite John's biological mother but the mother "attacks" the mistress:

mother R mistress
saying: "I don't want to see this slut (salope) in my daughter's wedding"

So logic must be used to decide how to construct the invitation list. Maybe inviting the mother is unreasonable, under the circumstances, but maybe nevertheless mother is a mother, she must be invited!

So the *semantics* for (S, R) is taken to be, according to the argumentation community, the choice of various maximal subsets[5]

$$E_1, E_2, \ldots, E_1 \subseteq S$$

[4] Figure 1 is probably a simulated planning figure, explaining various levels of expected attacks. If we think the data might invite serious attacks then several levels of green (defensive) levels will be used immediately as level 1.

[5] We are describing here what is generally understood by the argumentation community as "Semantics" for Argumentation networks. It is a function assigning to any (S, R) possible conflict free subsets of S. This is not the view held by D. Gabbay [6] and L. van der Torre,

such that each subset E satisfies:

1. E is conflict free, i.e. for no $x, y \in E$ do we have xRy.
2. E is maximal
3. Some other conditions are fulfilled. For example we do not want to have a situation where the father is invited, the father hates the mother (father R mother) (maybe because she gave him a very hard time with the divorce) the father is invited (he pays for the Wedding), the mother therefore is not invited, and yet the mistress/wife is not invited.

In the case of Security or Medicine, (S, R) represents the weapons. The sets of weapons can be representing security agents or doctors and their favourite available "weapons" and these are the parallel to the "arguments" in an argumentation networks. So a set of agents ("arguments") is a team of doctors or a team of security agents and here we can talk about conflict free in the sense that operationally (in an algorithm to protect the data/patient) they do not prescribe conflicting medicine but have a cooperating defensive strategy.

So, to sum up, the first component is a formal (S, R).

With a machinery (algorithm) for generating subsets, $E_1, E_2, \ldots$ each one has some logical reasons behind its construction. For example if S is the set of *all* security measures and counter-measures, and if R means

$$xRy \text{ if intuitively } x \text{ and } y \text{ are not compatible}$$

then a logic of *cost* and social/political considerations might choose:
E_1 = set of standard security measures (recommended via Google)
E_2 = standard tools of viruses to use in attack (recommended via Google)
etc.

Component 2. **This component is built out of component 1 and comprises a finite set/multiset of weapons.** We now use (S, R) with S viewed as a set of weapons and use (S, R) to define another attack and defense network. We enlist the help of Figure 8. We denote this network by $(\mathbf{S}, \twoheadrightarrow)$. (Its elements $\mathbf{m} \in \mathbf{S}$ are also denoted in bold.) Think of these elements $\mathbf{m}$ as mercenaries; as agents $\mathbf{m}$ carrying each a variety of weapons, these weapons being a subset $\mathbf{a}(\mathbf{m})$ of S. Thus we are relying in our attack and defense mercenary network on the weapons network (S, R). For example

$$\mathbf{a}(\mathbf{m}) = \{\{a_1, a_2\}, b_1, b_2\}, \text{ where } a_1, a_2, b_1, b_2 \in S, \mathbf{m} \in \mathbf{S}$$

We now explain the meaning of $\mathbf{a}(\mathbf{m})$.

*1) The meaning of $\mathbf{a}(\mathbf{m})$ is a collection of three weapons. The first weapon is a composite weapon built up of two component weapons $\{a_1$ and $a_2\}$. We denote it by $\{a_1, a_2\}$ but maybe better to use a special constructor $*$ and denote it by $a_1 * a_2$, as we do in Section A. The second weapon is b_1 and the third is b_2.

*2) So, if we want to attack $\mathbf{a}(\mathbf{m})$ we need to attack all three components weapons and leave $\mathbf{m}$ without weapons.

[5, 7], each for their own (complementary) reasons. A better view is that Semantics is a process assigning for each (S, R) another (S', R'). This view makes it easier for us in this paper to give up the conflict freeness requirement on sets of "arguments".

So how do we use $\{(a_1 * a_2), b_1, b_2\}$? We use them as resource weapons, which can be used for attack or for defence. $(a_1 * a_2)$ is a composite machine weapon which has two components. So to neutralise the composite weapon $(a_1 * a_2)$ we need to kill at least one of its components, and to attack **a(m)** we must attack each of its weapons. So if

$$\alpha = \{\{\alpha_1, \alpha_2, \alpha_3\}, \delta_1, \delta_2\}$$

is the set of weapons of another mercenary $\mathbf{m}'$, keen to attack **a(m)**, then for $\mathbf{m}'$ to use its α to attack **a** it must attack each of **m**'s weapons. So

*3)

$$\begin{aligned} \alpha \twoheadrightarrow \mathbf{a}(\mathbf{m}) \ \text{ iff } \ & \alpha \twoheadrightarrow (a_1 * a_2) \\ & \text{and } \alpha \twoheadrightarrow b_1 \\ & \text{and } \alpha \twoheadrightarrow b_2 \end{aligned}$$

So for α to attack any single weapon x (such as one of the weapons in $\mathbf{a}(\mathbf{m})$), we need a weapon in α to attack x. So we follow rule *4):

*4)

$$\begin{aligned} &\{z, y\} \twoheadrightarrow x \text{ iff def. } \ zRx \vee yRx \\ &u \twoheadrightarrow z * y \ \text{ iff def. } \ uRz \vee uRy \end{aligned}$$

where $z * y$ is a weapon with two components. So, for example

$$u \twoheadrightarrow \{\{z, y\}, w\} \text{ iff } u \twoheadrightarrow (z * y) \text{ and } uRw) \text{ iff } ((uRz \vee uRy) \wedge uRw).$$

Therefore we have

*5) $\{\{\alpha_1 * \alpha_2 * \alpha_3\}, \delta_1, \delta_2\} \twoheadrightarrow x$ iff $[\delta_1 Rx$ or $\delta_2 Rx$ or $(\alpha_1 * \alpha_2 * \alpha_3)Rx]$

and

*6) $\alpha \twoheadrightarrow \{(a_1, a_2), b_1, b_2\}$ iff $\alpha \twoheadrightarrow (a_1 * a_2)$ and $\alpha \twoheadrightarrow b_2$ and $\alpha \twoheadrightarrow b_2$.

This gives a full meaning to $\alpha \twoheadrightarrow \mathbf{a}(\mathbf{m})$. This is a weapons meaning. Not a logical meaning. We should *not* interpret "$*$" as a logical conjunction '$\wedge$". For comparison, let us see what logical meaning for $*$ (taken as "$\wedge$") would look like.

Example 1. Example of how the attack formation $\alpha \twoheadrightarrow \mathbf{a}(\mathbf{m})$ behaves when we interpret the $*$ as logical conjunction $\wedge$.

So let α attack $\mathbf{a}(\mathbf{m})$.

$$\alpha = \{\{\alpha_1, \alpha_2, \alpha_3\}, \delta_1, \delta_2\}.$$

So we read α as a formula of logic:

$$(\alpha_2 \wedge \alpha_2 \wedge \alpha_3) \vee \delta_1 \vee \delta_2$$

and similarly we read $\mathbf{a}(\mathbf{m})$ as the formula of logic

$$(a_1 \wedge a_2) \vee b_1 \vee b_2$$

So we want

(♯1) $\alpha \vdash \neg\mathbf{a}(\mathbf{m}) = \neg b_1 \wedge \neg b_2 \wedge (\neg a_1 \vee \neg a_2)$

but $\alpha = ((\alpha_2 \wedge \alpha_2 \wedge \alpha_3) \vee \delta_1 \vee \delta_2)$.

So we get that $(\sharp 1)$ means $(\sharp 2)$

$(\sharp 2)$ $(\alpha_1 \wedge \alpha_2 \wedge \alpha_3) \vdash \neg \mathbf{a}(\mathbf{m})$ and $\delta_1 \vdash \neg \mathbf{a}(\mathbf{m})$ and $\delta_2 \vdash \neg \mathbf{a}(\mathbf{m})$.

but logically

$(\sharp 3)$ $\neg \mathbf{a}(\mathbf{m}) = (\neg a_1 \vee \neg a_2) \wedge \neg b_1 \wedge \neg b_2$

So in terms of weapons if $*$ is interpreted as $\wedge$ and the collection of elements is interpreted as disjunction we get that *all* weapons in α must kill/attack each of the weapons in $\mathbf{a}(\mathbf{m})$.

This is *not* the interpretation we want.

Let us now look at Figure 4. Here we have, according to our weapon interpretation Figure 12, where to simplify let (we abuse notation, identifying the mercenary with its set of weapons):

$$\mathbf{b} = \{b_1, b_2\}, \beta = \{\alpha_1, \alpha_2\}, \mathbf{x} = \{x_1, x_2\}.$$

We get

$$\begin{tikzcd} \{b_2, b_2\} \\ \{\alpha_1, \alpha_2\} \arrow[u, twoheadrightarrow] \\ \{x_1, x_2\} \arrow[u, twoheadrightarrow] \end{tikzcd}$$

Fig. 12.

α_1, α_2 are used as weapons to kill $\{b_1, b_2\}$. So the meaning of $\{\alpha_1, \alpha_2\} \twoheadrightarrow \{b_1, b_2\}$ is $(\alpha_1 R b_1 \wedge \alpha_1 R b_2) \vee (\alpha_1 R b_1 \wedge \alpha_2 R b_2) \vee (\alpha_2 R b_1 \wedge \alpha_2 R b_2) \vee (\alpha_2 R b_1 \wedge \alpha_1 R b_2)$.

We would expect the meaning of

$$\{x_1, x_2\} \twoheadrightarrow \{\alpha_1, \alpha_2\}$$

to be similar, namely

$$(x_1 R \alpha_1 \wedge x_1 R \alpha_2) \vee (x_1 R \alpha_1 \wedge x_2 R \alpha_2) \vee (x_2 R \alpha_1 \wedge x_2 R \alpha_2) \vee (x_2 R \alpha_1 \wedge x_1 R \alpha_2).$$

However, our analysis of Figure 1 together with the meaning that $\mathbf{x}$ contains weapons x_1 and x_2 which can be used for defense, as in Figure 5, allows us to have more options. We can lend these weapons to $\mathbf{b}$ and strengthen $\mathbf{b}$'s defense. For example we can form as one option

$$\mathbf{b}' = \{b_1 * x_1, b_2 * x_2\}$$

and Figure 12 is transformed to Figure 13

$$\begin{tikzcd} \{b_1 * x_1, b_2 * x_2\} \\ \{\alpha_1, \alpha_2\} \arrow[u, two heads] \end{tikzcd}$$

Fig. 13.

Remark 2. We see here that the weapon interpretation uses weapons as a resource. If we have only one copy of x_1, we cannot form

$$\mathbf{b}'' = \{b_1 * x_1, b_2 * x_1, x_2\}$$

Here **x** strengthens **b** by making b_1 and b_2 stronger and even gives it x_2 as additional weapon. So β must kill each of these 3 items. Note that we need two copies of x_1. So we could move to linear logic and multisets.

Note that this Weapons point of view allows for loops. If we have $\{a, b\} \twoheadleftarrow\!\!\twoheadrightarrow \{c, d\}$, this is a loop which is stable if we only have aRc and dRb. The surviving weaponry is $\{a\}$ and $\{d\}$.

It is similar to discrete equilibrium of an ecology of two species. Imagine certain trees and certain grass growing under the trees. The grass processes the dead leaves from the tree and making better earth to enable the tree to make leaves.

Remark 3. Let us verify how the schematic model of this Section 2 addresses the required features listed in Remark 1. We check the items as listed one by one:

- Items 1. and 2. can be addressed as seen from the discussion in Example 1.
- Item 3. is not fully addressed. We can weaken our target by killing some of its weapons. We have not addressed directional attacks . This will be done later by offering a more sophisticated model along the same schematic lines.
- Items 4. and 5. are not addressed yet.
- Item 6. was discussed and can be addressed.
- Item 7. was not addressed.
- Item 8. was addressed.
- Item 9. will be addressed in the appendices where we discuss linear implication and weaponise it directly.

The appendices continue the Socratic development of a formal model from the requirements of an attack-defence network as developed in this paper.

References

1. B. Kordy, S. Mauw, S. Rdomirovic and P. Schweitzer. Attack and Defense trees. *Journal of Logic and Computation*, pp. 55–87, 2012.
2. W. Wideł, M. Audinot, B. Fila and S. Pinchinat. Beyond 2014: Formal method for attack tree-bsed security modelling. *ACM Comput. Surv.*, 1, 1, Aricle 1 (January 2019) 35 pages. `https://doi.org/10.1145/3331524`

3. D. Gabbay, H. Barringer and J. Woods. Temporal dynamics of argumentation networks. In *Mechanising Mathematical Reasoning*, a volume dedicated to J. Siekmann, D. Hutter and W. Stephan, eds. Springer Lecture Notes in Computer Science 2605, pp. 59–98, 2005. Expanded version published as: Temporal, Numerical and Metalevel Dynamics, in *Argumentation Networks; Argument and Computation*, 2012, Vol 3, issues 2-3, pp. 143–202.
4. D. Gabbay. Logical foundations for bipolar argumentation networks, in special issue of the Journal of Logic and Computation in honour of Arnon Avron, *J Logic Computation* (2016) 26(1): 247-292. doi: 10.1093/logcom/ext027 First published online: July 22, 2013
5. Ryuta Arisaka, Ken Satoh, and Leendert van der Torre. Anything You Say May Be Used against You in a Court of Law? Abstract Agent Argumentation (Triple-A)
6. D. Gabbay, G. Rozenberg and Students of CS Ashkelon; Introducing Abstract Argumentation with Many Lives: To appear in a refereed edited volume of Springer's series on Outstanding Contributions to Logic https://www.springer.com/series/10033), dedicated to the work of Arnon Avron (refereed and accepted for publication)
7. Pietro Baroni, Guido Boella, Federico Cerutti, Massimiliano Giacomin, Leendert W. N. van der Torre, and Serena Villata. On the input/output behavior of argumentation frameworks. *Artif. Intell.*, 217:144, 2014.
8. Dov Gabbay, Leon van der Torre, H Barringer and D Rydeheard. Reactive Activation and De-activation of Attacks in Formal Argumentation. Submitted to *Journal of Applied Logics*.
9. Barbara Kordy, Ludovic Piètre-Cambacédès, and Patrick Schweitzer. 2014. DAG-based attack and defense modeling: Don't miss the forest for the attack trees. *Computer Science Review* 13–14 (2014), 1–38.
10. D. Gabbay and John Woods. The laws of evidence and labelled deduction Published in *Phi-news*, pp. 5–46, October 2003, `http://phinews.ruc.dk/phinews4.pdf`. Updated version in D.M. Gabbay; Canivez, P.; Rahman, S.; Thiercelin, A. (Eds.) A*pproaches to Legal Rationality, Logic, Epistemology, and the Unity of Science* 20, 1st Edition., Springer 2010, pp 295–331 DOI 10.1007/978-90-481-9588-6_15,
11. Douglas Walton, Christopher Reed, Fabrizio Macagno *Argumentation Schemes*. Cambridge University Press, 2008. ISBN: 0521723744,9780521723749
12. Edmund Landau. Foundations of Analysis. The Arithmetic of Whole, Rational, Irrational and Complex Numbers. A Supplement to Text-Books on the Differential and Integral Calculus. 1 Jan 1951
13. Edmund Landau; translated by F. Steinhardt. *Differential and Integral Calculus*. 1 Jan 1960
14. Pietro Baroni, Dov Gabbay and Massimilino Giacomin (Editors). *Handbook of Formal Argumentation* College Publications, 28 Feb 2018, 1028 pp.
15. D. Gabbay . An Equational Approach to Argumentation Networks, in *Argument and Computation*, 2012, vol 3 issues (2-3), pp 87–142
16. `https://en.wikipedia.org/wiki/Cross-ratio`, accessed 19 march 2020, 1649 hours UK
17. Meyer R. K., McRobbie M. A., Belnap N. (1995) Linear analytic tableaux. In: Baumgartner P., Hähnle R., Possega J. (eds) Theorem Proving with Analytic Tableaux and Related Methods. TABLEAUX 1995. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), vol 918. Springer, Berlin, Heidelberg
18. D. Gabbay. *Labelled Deductive Systems; Principles and Applications*. Vol 1: Basic Principles (Monograph) Oxford University Press, 1996. 500pp (1st draft Manuscript 1989, 2nd intermediate draft, University of Munich, CIS Bericht 90-22, 1990. Third intermediate draft, Max Planck Institute, Saarbrucken, Technical Report, MPI-I 94-223, 1994.)
19. D. Gabbay , O. Rodrigues and J. Woods: Belief Contraction, Anti-formulas , and Resource Overdraft: Part I *Logic Journal of the IGPL*, 10, 601-652, 2002.
20. D. Gabbay , O. Rodrigues and J. Woods: Belief Contraction, Anti-formulae and Resource Overdraft: Part II In *Logic, Epistemology and the Unity of Science*, D. M. Gabbay, S. Rahman, J. Symons and J-P van Bendegem, eds. Pp. 291-326, Kluwer, 2004

Appendices

A A first design draft of a formal model

Following our discussions in Sections 1 and 2, we proceed now towards a more sophisticated model. We aim to study a first draft of a model with joint attacks, joint support, resource multisets, linear logic and weapons interpretation semantics.

We begin by collecting various building blocks towards constructing a model.

Block $\sharp 1$. *Point of view.* Given a set of nodes and a binary relation on the set, the traditional argumentation view of this system is that the nodes are (logical) arguments and the binary relation is the relation of logical attack. So imagine two lawyers arguing for and against the guilt of the accused.

This point of view is highly applicable in the legal domain. In contrast we are aiming for the Security and for the Medical domains, and for such domains we need a different point of view.

Our point of view, as discussed in Sections 1 and 2, is that the nodes are collections of weapons. We imagine two armies attacking each other with weapons. For example a doctor with a collection of medicines attacking various medical problems in a patient. Another example is a data security officer installing various measures to defend his data against *Chaos International* hackers.

Block $\sharp 2$. *The nature of our weapons.* Our choice of weapons is inspired by our analysis of the attack and defense in Figure 1; the figure of the attack and defense tree of paper [1]. The figure contains the details of many kinds of weapons. We need to simplify this assortment, for the sake of simplicity.

We envisage two kinds of weapons:

1. Attack weapons which can kill other attack weapons.
2. Supporting weapons which can enhance or protect other weapons or other defense means.

Example 2.

1. Weapon – Fighter plane. This is an attack weapon.
   Support enhancement - better radar and electronic communication
   Support protection - coat the exterior with stealth capability.
2. From Figure 1:
   Defense weapon – Password
   Attack Weapon (on passwords) – Dictionary attacks
   Support weapon (supporting passwords) – Strong password
   Support (of Dictionary Attacks) – Stronger password attacks.

Remark 4. Note that in this first design model we use only discrete weapons. So a cannister of poisonous gas can either be used in its entirety or not be used at all. We do not model, at this stage, the use of some of the poison (say 35%). Allowing for this option requires a special logical operators in the modelling.

Definition 1. *1. A weapons system has the form* $(W, W_a, W_s, \rho, *)$ *where*

(a) W_a is a multiset of atomic weapons, including the empty weapon **e***.*
(b) W_s is a multiset of atomic support weapons.
(c) $*$ is a binary concatenation algebraic operator, which is commutative and associative on W_s. It is used to form composite support weapons.
  - *any atomic s in W_s is also considered a composite support weapon.*
  - *if x and y are composite support weapons so is $x * y$.*
  - *$*$ can also form expressions of the form $z * (s_1 * s_2 \ldots * s_k)$, where $z \in W_a$ and $s_i \in W_s$.*
    See the exact role of $*$ in (d) below.

(d) W is the multi-set of composite attack weapons defined inductively using W_a, W_s, and $*$ as follows:
  - *An atomic attack weapon is a composite weapon (so* **e** *is also a composite attack weapon).*
  - *If x is a composite weapon and s is atomic support weapon then $x * s$ is a composite weapon. In particular so is $\mathbf{e} * s$. (Think of $\mathbf{e} * s$ as a sort of shield. It cannot attack and kill but it can protect.)*
  - *$*$ is commutative and associative, namely*

$$(x * s_1) * s_2 = x * (s_1 * s_2) = (x * s_2) * s_1 = x * (s_2 * s_1)$$

  - *If $x * (s_1 * \ldots * s_{k+1})$ is an attack weapon, so is $x * (s_1 * \ldots * s_k)$. (This is to allow moving enhancements such as s_{k+1} from one weapon to another.)*

2. *An arsenal of weapons* **A** *is a multiset of weapons from W and supports from W_s. (Think of* **A** *as a node in the attack and defense tree, representing a mercenary, or a doctor or a hacker, carrying the arsenal of weapons* **A***.)*
3. *ρ is a binary relation on W, satisfying the following conditions (ρ is the attack/kill relation). (We read $x\rho y$ as saying the weapon x is capable of attacking and damaging/killing the weapon y, should the mercenary choose to use it.)*
   (a) If $x\rho y$ holds then x is not of the form $\mathbf{e} * (s_1, \ldots, s_k)$. (I.e. a multiset of supports cannot kill anyone. Think of it as a shield.) We may, however, allow for attacks of the form $\mathbf{e} * s_1, \ldots, *s_m \rho \mathbf{e} * t_1, \ldots, *t_m$. See the table of Figure 2 item 4.
   (b) If $x\rho y$ and $s \in W_s$ then $x * s\rho y$.
   (c) $\forall x \in W \exists y \in W(y\rho x)$ (Spinoza principle)
   (d) If $z \in W_a$ and $y \in W$ and $y\rho z * (s_1, \ldots, s_k)$ then also $y\rho\mathbf{e} * (s_1, \ldots, s_k)$.
   Item (d) combined with item (b) imply that y kills the weapon z as well as each of its individual enhancements $\{s_i\}$.
   (e) It may be of interest to investigate the following principle. If x attacks y ($x\rho y$), then y can be enhanced to a y' such that x no longer attacks y'. (This is the opposite of the Spinoza principle.)
4. *Let* **A** *be an arsenal. Define a new arsenal called* **strip** *(***a***) as follows:*
   (a) For $z \in W_a$ such that $z * (s_1, \ldots, s_k)$ is in **A***, we let $z, s_1, \ldots, s_k$ be in* **strip** *(***A***).*
   (b) If $s \in W_s$ and $s \in$ **A** *then we let $s \in$* **strip***(***A***).*
   (c) We say that **A** *is equivalent to* **B** *iff (def)* **strip** *(***A***) =* **strip** *(***B***).*
   We write $\mathbf{A} \equiv \mathbf{B}$.

Remark 5. **Strip**(**A**) is the multi-set of resources used to build up **A** from elements of W_a, W_s and $*$. So when **A** is equivalent to **B**, then it means they use the same resources. We shall see below that if **A** wants to attack or defend it can re-organise itself into an equivalent better weapons deployment **B** using it resources.

Definition 2.

1. *Let* $\mathbf{A}$ *be an arsenal and let* $w = z * (s_1 *, \ldots, *s_k)$ *be a weapon* $\in W$. *We say that* $\mathbf{A}$ *is capable of killing* w *iff for some* $y \in \mathbf{strip}(\mathbf{A}) \cap W_a$ *and some* $t_1, \ldots, t_m \in \mathbf{strip}(\mathbf{A}) \cap W_s$, *we have that* $y * (t_1 * \ldots *_m)\rho w$.
   We further say that $\mathbf{A}$ *used the resources* $\{y, t_1, \ldots, t_m\}$ *to kill* w *and after the successful attack the arsenal* $\mathbf{A}$ *is depleted into* $\mathbf{strip}(\mathbf{A}) - \{y, t_1, \ldots, t_m\}$.
2. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be two arsenals. We say* $\mathbf{B}$ *is capable of surviving an all out attack from* $\mathbf{A}$ *iff there exists a* $\mathbf{B}' \equiv B$ *such that* $\mathbf{B}' = \{w_1, \ldots, w_m\}$ *and for any weapons* $y_1, \ldots, y_m$ *such that* $\bigwedge_i y_i\rho w_i$ *we have that*

$$\bigcup_i \mathbf{strip}(\{y_i\}) \subsetneq \mathbf{strip}(\mathbf{A}).$$

   In other words, $\mathbf{A}$ *does not have the resources* $(\mathbf{strip}(\mathbf{A}))$ *to kill every element in the defense formation* $\mathbf{B}'$ *(which is a rearrangement of the resources of* $\mathbf{B}$*).*
3. *We say* $\mathbf{A}$ *can weaken* $\mathbf{B}$, *if for any* $\mathbf{B}' \equiv B$ *there exists a* $w \in \mathbf{B}$ *and a* $y \in W$ *such that* $y\rho w$ *an* $\mathbf{strip}(\{y\}) \subseteq \mathbf{strip}(\mathbf{A})$.
   We say that $\mathbf{B}'$ *is weakened into* $\mathbf{B}' - \{w\}$ *and* $\mathbf{A}$ *is weakened by loosing its resources* $(\mathbf{strip}(\mathbf{A}))$ *by the amount* $\mathbf{strip}(\{y\})$.
4. *We say* $\mathbf{A}$ *can kill* $\mathbf{B}$ *iff* $\mathbf{B}$ *cannot survive an all out attack from* $\mathbf{A}$.

Remark 6. In Definition 2 we were concerned on how an arsenal $\mathbf{A}$ can mount an attack on a target $\mathbf{B}$. Basically what $\mathbf{A}$ has to do is to re-organize itself into its basic resources $\mathbf{strip}(\mathbf{A})$, and construct a weapon w which can attack (some weapon or resource t in the target, i.e. we have $w\rho t$.

The point we want to make here is that the possibility of attack depends on the attack relation ρ.

So if no matter how $\mathbf{A}$ re-organize its resources it cannot construct a weapon w which can attack anything, then $\mathbf{A}$ cannot mount any attack on any target.

Now we ask, how can $\mathbf{A}$ support a target $\mathbf{B}$? The answer it can always support any $\mathbf{B}$. It can always look at $\mathbf{strip}(\mathbf{A})$ and send some of its resources to $\mathbf{B}$.

Of course the geometry of the network dictates whom $\mathbf{A}$ wants to attack and whom it wants to support, but to attack – $\mathbf{A}$ needs to consult ρ while to support $\mathbf{A}$ can always send resources to the destination it wants.

Example 3. 1. Let $w = \{z * s, y * s\}$. $x = \{\alpha\}$.
   It may be that $\alpha\rho z * s, \alpha\rho y * s, \alpha\rho\mathbf{e} * s * s$, but $\neg\alpha\rho(z * s * s)$.
   Let $\mathbf{A} = \{\alpha\}, \mathbf{B} = \{z, y, e, s, s\}$.
   Then $\mathbf{A}$ can kill $\mathbf{B}$ but if $\mathbf{B}$ reorganises its defense into $\mathbf{B}' = \{\mathbf{e}, z * s * s, y\}$ then it can survive.
   Note that we may have an interesting situation here if we have $(z * s * s)\rho\alpha$. Then $\mathbf{B}$ can also kill $\mathbf{A}$ by reorganising into $\mathbf{B}'$.
2. Note that if we have $\mathbf{A}' = \{\alpha, \alpha\}, \mathbf{B}_1 = \{z, s\}, \mathbf{B}_2 = \{y, s\}$.
   Then $\mathbf{A}'$ can kill both $\mathbf{B}_1$ and $\mathbf{B}_2$ but if they create an alliance $\mathbf{B}_1 \oplus \mathbf{B}_2$ then they can not only survive but weaken $\mathbf{A}'$.
   Furthermore, if we look at $\mathbf{A}$, then $\mathbf{A}$ can kill one of $\{\mathbf{B}_1, \mathbf{B}_2\}$ but again if they form $\mathbf{B}_1 \oplus \mathbf{B}_2$ they can kill $\mathbf{A}$.

Definition 3. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be arsenals. Let* $\mathbf{C}$ *be an arsenal. We say* $\mathbf{C}$ *can protect* $\mathbf{B}$ *from killing attack from* $\mathbf{A}$*, if for every* $\mathbf{B}' \equiv \mathbf{B}$ *and very* $\mathbf{A}' \equiv \mathbf{A}$ *which can kill* $\mathbf{B}'$*, there exists a* $\mathbf{C}' \equiv \mathbf{C}$ *such that* $\mathbf{C}'$ *has enough resources to support* $\mathbf{B}'$ *to be* $\mathbf{B}''$ *and also weaken* $\mathbf{A}'$ *to* $\mathbf{A}''$ *so that* $\mathbf{B}''$ *can survive* $\mathbf{A}''$*.*

Block $\sharp 3$*. Reinterpreting the weapons of Block* $\sharp 2$ *s arguments (legal).* This move is to show that our weapons model constructed intuitively by looking at security attack and defense can also be used as a model in argumenation in the legal domain. Such a move will reinforce our confidence in the model and will also benefit the argumentation research area and community and maybe also mobilise them to take an interest in security attack and defense networks and in resource and linear logic.

We present our comparison in a table, see Table 2. The table refers to the components of a model of the form $(W, W_a, W_s, \rho, *)$. We use examples from the literature both in argumentation and in security.

Block $\sharp 4$*. Defining attack and defense networks and their semantics.* This block gives a schematic view. The next block $\sharp 5$ will give a more detailed mathematical view.

We present our schematic view by looking at Figure 14.

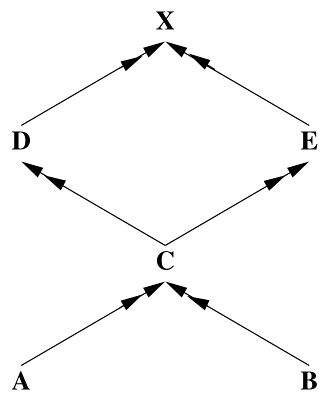


Fig. 14.

Note the following:

1. (a) We use only attack arrows "$\twoheadrightarrow$". The defense/support comes as part of the attack as we explain later. This is compatible with Figure 1.
   (b) Assume the graph is finite acyclic. We deal with loops later.
2. Consider node $\mathbf{C}$. Its resources are $\mathbf{strip}(\mathbf{C})$.
3. Its semantic "desire" in terms of our model (to be agreed and finalised) is as follows:
   (a) Top priority — protect itself against attacks from $\mathbf{A}$ and $\mathbf{B}$.
   (b) These attacks will diminish $\mathbf{C}$ resources, because it might weaken $\mathbf{C}$ or maybe even kill it.

Table 2.

Component	**Meaning in security (block ♯2)**	**Meaning in argumentation**
1. atomic weapon $z \in W_a$	use weapon z (from Figure 8	fact z: the accused murdered the victim
2. Atomic support s (forming $z * s$)	s enhances the weapon z.	s is witness to fact z. (i) s is written as label $s : z$, see [10]. (ii) or s written as $s \to z$ forming part of support group of z. See [4]. (iii) or s is part of argumenation scheme, see [11, p. 90]. (iv) or we can use a single support letter s and let the number of concatenated s's indicate the strength of the argument or how many lives it has etc. So $z * s * s * s$ indicates that z has strength 3 we can also write it as $3 : z$
3. Note about item 2 above.	Our algebra of resources in the security case allows us to strip a support from a $z * s$ and add it to a $y * s$, for $z \neq y$. In argumentation this is not allowed. A witness for z cannot become a witness for $y, y \neq z$. Of course in a completely corrupt society where evidence can be faked and witnesses can be bought or hired, the algebra will be the same.	
4. $z\rho y$ or $z\rho s$ is allowed	A weapon z can attack another weapon y or a support s but we do not have $s\rho s'$. If we allow $\mathbf{e} * s\rho e * s'$ then OK.	A fact or argument can attack another argument but can also attack a supporting witness. In fact more witnesses can attack a witness saying for example "how can you testify to fact z when you wee abroad that same day!"
5. The way attack and support are conducted and what semantics we have.	This comparison will continue after we present below *Block* (♯4)*: the semantics of attack and defense*.	

Question. Do we assume **A** and **B** join resources?? I.e. form $\mathbf{strip}(\mathbf{A}) \cup \mathbf{strip}(\mathbf{B})$.
Answer. To simplify, assume no cooperation.

(c) With its diminished resources **C** wants to
(c1) attack **D** and **E** and weaken them.
(c2) At the same time send remaining resources to support **X**.
We need to formulate as part of our semantics a policy for nodes like **C** to do this, (c1) and (c2).

4. Following the above principles and policies, we propagate our attacks along the finite acyclic graph from the endpoints (points not attacked). When we reach the end of the process, we stop.
5. So our semantics, a process to be defined and finalised, is a functional $\mathbb{F}$ which produces, for each finite acyclic graph of attacking arsenals, another graph where each arsenal $\mathbf{Z}$ is reduced to $\mathbf{Z}'$.

Block ♯5. *A more detailed mathematical model.* We start by simplifying the definition of a weapon system $(W, W_a, W_s, \rho, *)$ (given in Definition 1), which will enable us to give a better semantics which will more directly connect with concepts from the argumentation area.

Remark 7. Consider the weapon system $(W, W_a, W_s, \rho, *)$ of Definition 1. We want to discuss how to simplify this definition with a view of making the connection with argumentation more transparent and with a view of making it easier for the security community to understand it and use it. This remark explains the simplification and the next series of definitions will lead to a formal definition of the simplified weapon system.

We propose the following simplifications:

1. Assume $W_a = W_s = W_0$ with W_0 being a set of free generators for a free Abelian group. so W_0 has elements of the form
$$\{\mathbf{e}, g_1, g_1^{-1}, g_2, g_2^{-1}, \ldots\}.$$
The elements g_i are the positive generators and we also form their group inverses namely we form the elements g_i^{-1} which are considered negative (the group is multiplicative, the words Positive and Negative express our view of how they are used as resource).
2. $*$ is the free group multiplication, which is associative and commutative.
The element $\mathbf{e}$ is the unit element and all other elements w are generated as products of the form $w = w_1 * w_2 * ... * w_n$, with each $w_i \in W_0$.
3. So the elements of W have the form $\alpha = \alpha_1 * \alpha_2 * \ldots * \alpha_k, \beta = \beta_1 * \ldots * \beta_m$ with the following holding:
$$\alpha * \beta = \text{def.}\alpha_1 * \ldots * \alpha_k * \beta_1 * \ldots * \beta_m$$
$$\alpha^{-1} = \alpha_k^{-1} * \alpha_{k-1}^{-1} * \ldots * \alpha_1^{-1}.$$
We have $\mathbf{e} * \alpha = \alpha * \mathbf{e} = \alpha$ and $\alpha * \alpha^{-1} = \mathbf{e}$.
4. ρ is a binary relation on W saying which element is permitted to attack which other element

5. (a) From commutativity and (3) above we get a normal form for each $\alpha \in W$, namely
$$\alpha = g_1 * \ldots g_k * h_1^{-1} * \ldots * h_m^{-1}$$
with $g_i \neq h_j$ for all i, j.
   (b) We say that the elements g_i and h_j^{-1} appear as atomic components in the word α.
6. (a) An arsenal **A** is a set of elements from the Abelian group.
   (b) Let **strip**(**A**) be the mutiset of all atomic elements appearing as components in any word of **A**.
   (c) we can define that **A** is Equivalent to **B** iff **strip**(**A**) = **strip**(**B**).
   (d) **A** can re-organise its resources into an equivalent **B** depending on what it wants to do, attack or defend or support
7. According to Definition 2 the question of whether a word x can attack y is determined by ρ. However, in view of Remark 6 and in order to simplify the model, we do not use ρ. Any x can attack or support any y, all we need to say is how to do it!
8. We now explain how we are going to define attack and support.
   (a) For α to attack β we form the word $\alpha * \beta^{-1}$.
   (b) For α to support β we form the word $\alpha * \beta$.
9. Now consider the sequence of attacks of Figure 15.
   Because this is only an example, let us assume that the nodes do not re-organise themselves for better attack or support (or maybe assume that what you see in the figure is after a suitable re-organisation has taken place).

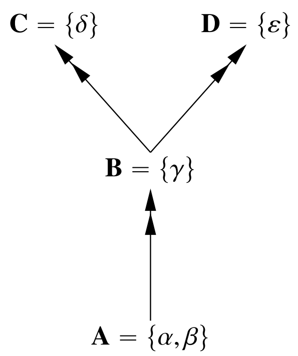


Fig. 15.

The options for **A** are to divide its resources between attacking **B** and supporting **C** and/or **D**. Let us examine several scenarios:

Scenario (a). **A** attacks **B** with α and supports **D** with β.
Let us follow (execute) this scenario following the arrows of the tree.

Step 1 of Scenario a. **A** uses all of its resources and becomes $\mathbf{A}' = \{\mathbf{e}\}$. **B** becomes $\mathbf{B}' = \{\gamma * \alpha^{-1}\}$ and **D** becomes $\mathbf{D}' = \{\varepsilon * \beta\}$, **C** is unchanged.

Step 2 of Scenario a. Now $\mathbf{B}'$ has the option of continuing the scenario and suppose it decides to attack $\mathbf{D}'$. So $\mathbf{D}'$ becomes

$$\mathbf{D}'' = \{\varepsilon * \beta * (\gamma * \alpha^{-1})\} = \{\varepsilon * \beta * \alpha * \gamma^{-1}\}.$$

C remains unchanged.

We now change Scenario a slightly to Scenario b.

Scenario (b). Suppose **A** decided to support **D** completely without attacking **B**. So **D** becomes $\mathbf{D}_1 = \{\alpha*\beta*\varepsilon\}$ and **B** attacks $\mathbf{D}_1$ and it becomes again $\mathbf{D}' = \{\alpha*\beta*\varepsilon*\gamma^{-1}\}$. **C** remains unchanged.

Scenario (c). Now suppose **A** attacks **B** with α to form $\mathbf{B}' = \{\gamma * \alpha^{-1}\}$. **A** supports **C** to form $\mathbf{C}' = \{\delta * \beta\}$ and $\mathbf{B}'$ attacks $\mathbf{C}'$ (not **D**) to form $\mathbf{C}'' = \{\delta * \beta * (\gamma * \alpha^{-1})\} = \{\delta * \beta * \alpha * \gamma^{-1}\}$. **D** remains unchanged.

We need to stipulate the following law:

(λ) **Law of conservation of resources**.

This law needs to be properly formulated. The problem is that if we have $\gamma\rho\gamma$, that is resource γ attacks resource γ, we get $\gamma * \gamma^{-1} = \mathbf{e}$, and it looks like resources are being destroyed.

We need a remedy to this problem. Perhaps what we need to do it to modify item 5 of Definition 7 and say that the normal form of any word α is the form

$$\alpha = (g_1 * * * g_k) * (h_1^{-1} * * * h_m^{-1}) * ((r_1 * r_1^{-1}) * (r_2 * r_2^{-1}) * ....)$$

with $g_i \neq h_j$ for all i, j.

To accommodate the group rule of $w * w^{-1} = \mathbf{e}$, we define the equality class of α to be all β with the same initial part $(g_1 * * * g_k) * (h_1^{-1} * * * h_m^{-1})$. So by looking at the equality class of α we ignore the tail of α, namely we ignore $((r_1 * (r_1^{-1}) * (r_2 * (r_2^{-1}) * **)$, which is in the group equal to $(\mathbf{e} * \mathbf{e} * **) = \mathbf{e}$. So the equality classes do carry a record of the resources. This way the resources remain recorded.[6]

Let us now compute resources for our scenarios.

The final resources of Scenario (a):

$\mathbf{A}' = \mathbf{B}'' = \{\mathbf{e}\}$ (no resources)

$\mathbf{D}'' = \{\varepsilon * \alpha * \beta * \gamma^{-1}\}$. $\mathbf{C} = \{\delta\}$.

Final resources of Scenario (c)

$\mathbf{A}' = \mathbf{B}'' = \{\mathbf{e}\}$, no resources.

$\mathbf{C}'' = \{\delta * \beta * \alpha * \gamma^{-1}\}, \mathbf{D}' = \{\varepsilon\}$.

We must have by conservation of resources that

$$\mathbf{strip}(\{\varepsilon * \alpha * \beta * \gamma^{-1}\} \cup \{\delta\}) = \mathbf{strip}(\{\delta * \beta * \alpha * \gamma^{-1}\} \cup \{\varepsilon\})$$

[6] Note that we are saying "remain recorded", it does not imply that resources are not lost. This just help record our losses. However when we get to more precise mathematical definitions such as Definition 12, where the approach is that when x supports u, then x gives u resources, and when x attacks u, then x steals/takes resources from u. So either way resources are conserved. The result of a complex attack defense tree is a redistribution of resources. This will be discussed at length in Section **??**.3.

Remark 8. Let us follow Remarks 1 and 3 and check which additional items have been addresses in this Section A.

We have addressed Directional attacks and generally expanded the model with more details.

B Formal argumentation meets attack and defense graphs from security

Following our developments in Sections 1–A we are now ready to continue with a more specific model. This time we move from the direction of argumentation towards the goal of security. In this spirit this section develops a model of argumentation inspired by attack and defense graphs from the security area. Our exposition is mathematical, following the method introduced by Edmund Landau a century ago. See [12, 13]. However, for the sake of the non mathematical reader, we add explanations in the footnotes. This way we do not interrupt the mathematical flow but still recognise the perceptive students of argumentation.

B.1 Weaponising the attack and support (defense) nodes in a bipolar network

Definition 4. *An attack and defense (AD) graph has the form (S, R_a, R_s), where $S \neq \emptyset$ is the set of nodes, R_a and R_s are binary relations on S, R_a is referred to as the attack relation denoted also by $\twoheadrightarrow$ and R_s is referred to as the support relation and is denoted by $\rightarrow$. We also let $R = R_a \cup R_s$ and assume that (S, R) is a finite acyclic graph.*

Definition 5. *Let $\mathcal{N} = (S, R_a, R_s)$ be an AD network. We define four rewrite transformation rules on $\mathcal{N}$.*

1. **Attack elimination rule**
   If $x \twoheadrightarrow y \twoheadrightarrow z$ is in $\mathcal{N}$ (i.e., $(x, y) \in R_a \wedge (y, z) \in R_a$) then transform "$x \twoheadrightarrow y$" into "$x \rightarrow z$". That is, let
   $$\begin{aligned} R'_a &= R_a - \{(x, y)\} \\ R'_s &= R_s \cup \{(x, z)\} \end{aligned}$$
   and let $\mathcal{N}' = (S, R'_a, R'_s)$.[7]
2. **Support elimination rule**
   If $x \rightarrow y \twoheadrightarrow z$ is in $\mathcal{N}$ (i.e., $(x, y) \in R_s \wedge (y, z) \in R_a$), then we transform "$x \rightarrow y$" into "$(x \twoheadrightarrow z)$". That is, let
   $$\begin{aligned} R'_s &= (R_s - \{(x, y)\}) \\ R'_a &= R_a \cup \{(x, z)\} \end{aligned}$$
   Let $\mathcal{N}' = (S, R_a, R'_s)$.

[7] Note that following Definition 12, (this definition is given later in this Subsection) when x attacks y, x takes resources from y, therefore technically, it looks like y (reluctantly) supports x.

This will be discussed in Section **??**.3. To see the logic of it, suppose a hacker takes control of your computer and changes all the passwords and encrypts the data. Then the hacker asks for ransom. The net result of this cycle (if you pay) is that you are technically giving a cash support to the attacker. If the attack only destroys resources and does not steal/them, then the above scenario is not modelled.

3. **Support chain reduction rule**
   If $x \to y \to z$, (i.e., $(x, y) \in R_s \wedge (y, z) \in R_s$) then let $R'_s = (R_s - \{(x, y)\} \cup \{(x, z)\}$ and let $\mathcal{N}' = (S, R_a, R'_s)$.
4. **Attack chain reduction rule**
   If $x \twoheadrightarrow y \to z$, (i.e., $(x, y) \in R_a \wedge (y, z) \in R_s$) then transform "$x \twoheadrightarrow y$" into $(x \twoheadrightarrow z)$. That is, let
   $$R'_a = (R_a - \{(x, y)\}) \cup \{x, z)\}$$
   and let $\mathcal{N} = (S, R'_a, R_s)$.

Definition 6. *Let $\mathcal{N} = (S, R_a, R_s)$ be an AD network. Let $R = R_a \cup R_s$.*

1. *$x \in S$ is called a top point iff there is no $y \in S$ such that $(x, y) \in R$.*
2. *$x \in S$ is called a bottom point iff there is no $z \in S$ such that $(z, x) \in R$.*
3. *$\mathcal{N}$ is said to be in normal form iff every node $x \in S$ is either a top point or a bottom point or both.*

Lemma 1. *Let $\mathcal{N} = (S, R_a, R_s)$. Then using the transformation rules of Definition 6, we can transform $\mathcal{N}$ into a unique normal $\mathcal{B}' = (S, R'_a, R'_s)$.*

Proof. Since (S, R) is acyclic, we can prove by induction using the transformation rules from the bottom points.

Remark 9.

1. The AD graph of Definition 4 is also known in the argumentation community (COMMA, see [14]) as bipolar network (see Gabbay, [4]).
2. Note that we allow for an $x \in S$ to both attack and support a $y \in S$ (i.e., $(x, y) \in R_a \cap R_s$).
3. Note that we shall manipulate a AD graph in a different novel way than in traditional (Dung) argumentation.

Definition 7.

1. *A finitely generated free Abelian group (with n generators) has the form $\mathcal{A} = (A, G, *, E, \mathbf{e})$, where G is a finite set of distinct atomic generators, $G = \{g_1, \ldots, g_n\}$.*
   - *A is the set of elements of the group if the form $\alpha = g_1^{m_1(\alpha)} * g_2^{m_2(\alpha)} * \ldots * g_n^{m_n(\alpha)}$ where m_i are integers.*
   - *$*$ is a commutative and associative multiplication.*
   - $\mathbf{e} = g_1^0 \ldots * g_n^0$
   - *We let*
     $$\alpha * \beta = g_1^{m_1(\alpha)} * \ldots * g_n^{m_n(\alpha)} * g_1^{m_1(\beta)} * \ldots * g_n^{m_n(\beta)}$$
     $$\alpha^{-1} = g_1^{-m_1(\alpha)} * \ldots * g_n^{-m_n(\alpha)}$$
   - *E is the set of equations of the form*
     $$(g_i^m * g_i^k) = g_i^{m+k}$$
     and
     $$\alpha * g_i^0 = \alpha$$

for all generators $g_i, i = 1, \ldots, m$.[8]

Lemma 2. *Let $\mathcal{A}$ be an Abelian group $(A, G, *, E, \mathbf{e})$ and let $\alpha \in A, \alpha \neq \mathbf{e}$, then there exist generators $h_1, \ldots, h_k \in G, k \leqslant n$ and integers $m_1, \ldots, m_k$ with $m_j \neq 0, j = 1, \ldots, k$ such that $\alpha = h_1^{m_1} * \ldots * h_k^{m_k}$.*

Proof. Follows from implementing the commutativity of * and the equations E.

Definition 8. *Let $\mathcal{A} = (A, G, *, E, \mathbf{e})$ be an Abelian group. Let $\alpha \neq \mathbf{e}$ be in canonical form as in Lemma 2, namely*

$$\alpha = h_1^{m_1} * \ldots * h_k^{m_k}.$$

Let $1 \leqslant j \leqslant k$, then

1. *If $m_j > 0$, then we say α has a positive resource of m_j copies of h_j respectively for each j.*
2. *If $m_j < 0$ we say that α lacks (is short of) m_j copies of h_j respectively for each j.*
3. *If $\alpha = \mathbf{e}$ then we say that α has 0 resources of any kind.*[9]

Definition 9. *Let α,β be two elements of the Abelian group. Let α have $m > 0$ resource copies of g. Let β have m' resource copies of g.*

1. *We define the notion of α attacks β with $g^k, k \leqslant m$, to means that α executes an attack on β, using k copies of its positive g resource, and as a result of this attack β becomes $\beta' = \beta * g^{-k}$. So β' has $m' - k$ resources of g and α becomes $\alpha' = \alpha * g^k$ and so α gains (it steals from β, through the attack) g^k resources.*
2. *We say α supports β with $g^k, k \leqslant m$ of its resources to mean that β becomes $\beta'' = \beta * g^k$, and so β'' has now $m' + k$ resource of g and α loses (it gives to β through the support) g^k resource to become $\alpha' = \alpha * g^{-k}$.*

[8] Let us explain the argumentation meaning of this definition. The set of arguments is the set G of generators. Each argument g in G can have a strength m, (where m is an integer) expressed by the number of resource copies of g is available. We present this by writing g^m. If $m < 0$, understand this as strength "overdraft".

What is the attack relation? Let us take the attack to be strength reduction, and we allow any element g to attack only itself (i.e., in terms of item 3 of Definition 1, we have that only $g\rho g$ holds for any g in G).

Viewed as elements of the Abelian group generated by G, attack and support (with $\rho = \{(g, g) | g \in G\}$ can be defined using group multiplication *, as done in Definition 9.

[9] The argumentation reader should compare this definition with a bipolar network enriched with strength of arguments. This has the form $(S, R_a, R_s, \mathbf{f})$, where $\mathbf{f}$ is a function on S giving for each $x \in S$ an integer number (positive, negative or zero) which is its strength. If the strength of x is positive than x is considered "in", otherwise x is "out". If xR_ay holds and x is "in", then x can actually mount an attack on y and the result of the attack is to subtract the strength of x from that of y. If xR_sy holds and x is "in" then x can actually mount a support of y and add its strength to that of y (y may have a large negative strength and y may still remain out).

The novelty in what we do in this section is that the attacker or supporter arguments looses strength when executing its attack and support and is allowed to make strategic choices of where to invest his resources. These new complications are not arbitrary new ideas but come from observing considerations in Security networks.

Remark 10.

1. If for example $\alpha = g^{-2} * h^3$ then α can attack or defend with resources up to h^3 but it cannot do anything with g^{-2}. Our current model does not allow "borrowing resources".
2. Also if $\beta = h$, then it makes no sense for α to spend resources and attack β with h^2, because β will become $\beta' = h^{-1}$. It is enough to attack with h only, neutralising β to be $\beta' = h^0 = \mathbf{e}$.
If β can be supported by a $\gamma = h^2$ then to make sure β is neutralised it may be worth for α to attack β with h^3.
3. Mathematically α can both attack and support β. For example it can support β by h^1 and attack β by h^2. So after this combined attack and support effort of α, α will gain one h resource and β will become $\beta''' = \beta * h * h^{-2} = h^0 = \mathbf{e}$.
There may be applications where this is done. For example the UK government gives retired people a state pension of say £5000, but they also consider it as income and tax it.
Other governments (e.g., Israel) do not tax state pensions.

Definition 10. *Let (S, R_a, R_s) be an AD graph and let $\mathcal{A}$ be an Abelian group. A system $(S, R_a, R_s, \mathcal{A}, \mathbb{F})$ is a weaponised argumentation network if $\mathbb{F}$ is a function from S into A associating with each $t \in S$ an element $\alpha = h_1^{m_1} * \ldots * h_k^{m_k}$ of the group, considered as a resource/weapon for attack or support.*[10]

Definition 11.

1. *Let (S, R_a, R_s) be an AD network. Let $x \in S$ be a point. We define certain sets of points having to do with x as follows:*
   (a) *Attack* $(x) = \{y \in S \,|\, x \twoheadrightarrow y\}$.
   (b) *Support* $(x) = \{z \in S \,|\, x \rightarrow z\}$
   (c) *Constellation* $(x) = \mathit{Attack}(x) \cup \mathit{Support}(x) \cup \bigcup_{y \in \mathit{Attack}(x) \cup \mathit{Support}(x)} (\mathit{Attack}(y) \cup \mathit{Support}(y))$

Definition 12.

1. *Let $\mathcal{N} = (S, R_a, R_s)$ be an AD network.*
Let $\mathcal{A} = (A, G, *, E, \mathbf{e})$ be a Abelian group, with its element $\alpha \in A$, considered as resources. We can use $\mathcal{N}$ and $\mathcal{A}$ to define a weaponised argumentation network $(\mathcal{N}, \mathcal{A}, \mathbb{F})$ by letting $\mathbb{F}$ to be any function from S into A

$$\mathbb{F} : S \mapsto A$$

[10] From now on the definitions become mathematically complicated. Consider the system $(S, R_a, R_s, \mathbf{f})$ of Footnote 9, and let us form the product of k copies of it. So the elements of the product will be vectors of k elements from S and the attacks and supports are executed coordinate wise. Of course if we do not allow cross coordinates attacks , then having a product of k copies is nothing new, since each copy/coordinate operates independently , without any cross coordinate interaction. This subsection and the next go down this path and indeed the Abelian group view is simpler mathematically to follow. We set out some algorithms which apply also to cross coordinate interaction but can be more easily understood in this simple case. We do however take an argumentation view of these subsections in the following SubSection **??**.4, where we add various complications as well as address cross coordinate interaction. The perceptive argumentation reader is invited to go to that subsection.

with $\mathbb{F}(x) = \alpha$, where $x \in S$ and α an element in normal form

$$\alpha = h_1^{m_1} * \ldots * h_k^{m_k}$$

where $m_j \neq 0, j = 1, \ldots, k$ and where $h_j^{m_j}$ can be used to attack or support only if $m_j > 0$. We can assume that the first n generators, $n \leqslant k$ are the ones with exponent $m_j > 0$.

2. *For a given x and $\mathbb{F}(x) = \alpha$ as in item 1 above, we say the following:*
   (a) *The resources available for x to attack and or support are products of elements of the form h_j^r where $m_j > 0$ and $0 < r \leqslant m_j$. Thus we say that x has the resources to attack or support several nodes u_i with the respective elements β_i, if*
   (a1) *Each β_i is a product of the first n generators of α (recall that these are the generators with exponent $m_j > 0$). We further assume that all the generators appearing in the product of each β_i have positive exponents > 0.*
   (a2) *The exponent of each generator h_j in the $*$ product of all the β_i does not exceed m_j.*
   (b) *x is allowed to attack or support a node $u \in S$ only as specified in the options listed in Definition 13.*
   (c) *If x has the resources β_i available in $\mathbb{F}(x)$ and chooses to attack or support several nodes u_i respectively and is allowed to attack or support each u_i (as per item (2b) above) then to attack each u_i with β_i we get a new $\mathbb{F}'(u_i) = \mathbb{F}(u_i) * \beta_i^{-1}$. To support u_i with β_i we get a new $\mathbb{F}''(u_i) * \beta_i$ and as a result of all of these attacks we get a new $\mathbb{F}'(x) = \mathbb{F}(x) * \beta^{-1} * \beta'$ where β is the $*$ product of all the β_i which support u_i and β' is the $*$ product of all the β_i which attack u_i respectively.*

Definition 13. *Let $(\mathcal{N}, \mathcal{A}, \mathbb{F})$ be a weaponised argumentation network. Let $\mathcal{N} = (S, R_a, R_s)$ and let $x \in S$. We now list the nodes $u \in S$ which x is allowed to attack or support. This is a purely geometrical notation and is independent of $\mathbb{F}$. $\mathbb{F}$ only gives resources but does not give permission to attack or support.*

Let $x \in S$ and let $C(x)$ be the constellation for x as defined in Definition 11. Consider the network

$$NC_x = (C(x), R_a \upharpoonright C(x), R_s \upharpoonright C(x)).$$

This is an acyclic finite network (since $\mathcal{N}$ is finite acyclic). There are the following options for a path from the bottom point x to a top point z possibly through a middle point y. We list the possibilities.

1. *x (is both top and bottom point)*
2. *$x \twoheadrightarrow z$*
3. *$x \rightarrow z$*
4. *$x \twoheadrightarrow y \twoheadrightarrow z$*
5. *$x \twoheadrightarrow y \rightarrow z$*
6. *$x \rightarrow y \twoheadrightarrow z$*
7. *$x \rightarrow y \rightarrow z$.*

For each of the above geometric options, the node x has the following attack and/or support permissions. (We assume x, y, z have resources to attack, support, etc.)

1. *No options available.*
2. *x can attack z*
3. *x can support z*
4. *x can attack y or x can support z or both (if resources are available)*
5. *x can attack y or attack z or both*
6. *x can support y or attack z or both*
7. *x can support y or support z or both.*

Definition 14. *Let* $(\mathcal{N}, \mathcal{A}, \mathbb{F})$ *be a weaponised argumentation network. Let* $x \in S$ *be a node and let* $\mathbb{P}(x)$ *be all the nodes which x is geometrically permitted to attack or support according to Definition 13. Let* $\mathbb{P}_0(x) \subseteq \mathbb{P}(x)$ *be the set of nodes (may be* $\varnothing$*) which x chooses to attack or support with resources* $\beta(u)$ *for each* $u \in \mathbb{P}_0(x)$*.*

We assume that x has the resources in $\mathbb{F}(x)$ *sufficient for all the* $\beta(u), u \in \mathbb{P}_0$ *as defined in Definition 12. In such a case we say that "x makes the* **choice**$(x, \mathbb{F}) = \{(u, \beta(u)) | u \in \mathbb{P}_0\}$.

This choice depends on $\mathbb{F}$ *because* $\mathbb{F}$ *says how much resources x has, i.e.* $\mathbb{F}(x)$ *has.*

Example 4. [11] Consider the constellation in Figure 16 together with resources $\mathbb{F}_1$ available for attack or support for some of the nodes.

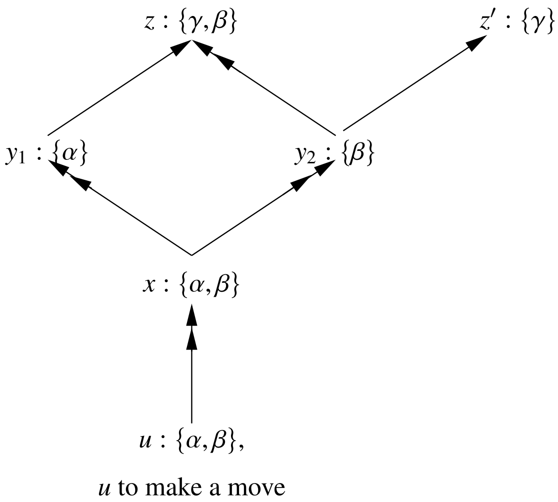


Fig. 16. The resource annotation in this Figure is $\mathbb{F}_1$

1. u is a bottom point. Its resources will not change because it is not being attacked or supported. So u has to make a move first.
2. u can attack x, with some of its resources. It can also support y_1 and or support y_2. It can divide its resources between all three options. =

[11] This example illustrates the general Definition 17 of the next subsection.

3. x is allowed to attack y_1 or attack y_2 or support z or support z'. x cannot make a move until u makes a move because it does not know what resources it is going to have.
4. (1)–(3) above prompt us to describe a sequence of moves. So let us describe one possible scenario:

   Step 1. u decides to support y_2 with β. This move is allowed. The resources of the figure following the execution of this attack become $\mathbb{F}_2$ as in Figure 17.
   In this figure x needs to make a move. y_1 and y_2 cannot move before x makes his move.

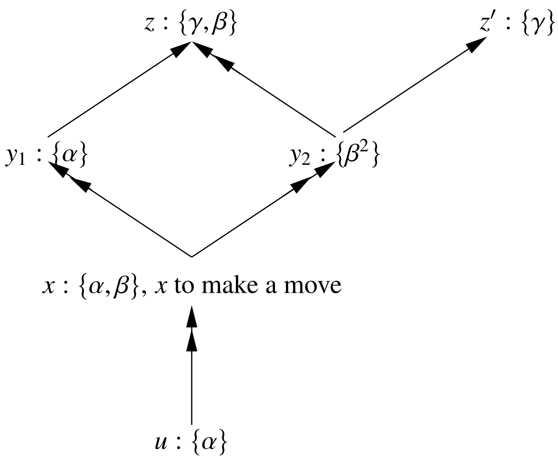


Fig. 17. The resource annotation in this Figure is $\mathbb{F}_2$

   Step 2. Now that u has made his move, x can now move. x chooses to attack y_1 with α and attack y_2 with β. We get the situation in Figure 18 with resources $\mathbb{F}_3$.

   Step 3. Now that x has made his move, y_1 and y_2 can make their move. y_1 has no resources. y_2 decides to attack z. The new figure is Figure 19 with resources $\mathbb{F}_4$.

   Step 4. Now z and z' can make a move but since they are top nodes, they have no moves to do and we stop.
5. What happened in the above scenario/sequence of steps done in item (4)? We started with Figure 16. This figure was annotated with weapons $\mathbb{F}_1$ as in the figure, together with annotation of which points to make a move (point U). Step 1 was to describe u make one of its allowable moves, moving to fFigure 17, where $\mathbb{F}_1$ becomes $\mathbb{F}_2$ as shown in this figure, together with the annotation that x can now make a move using the $\mathbb{F}_2$ resources.
   This process went on through $\mathbb{F}_3$ of Figure 18 and $\mathbb{F}_4$ of Figure 19 and there the process stopped.

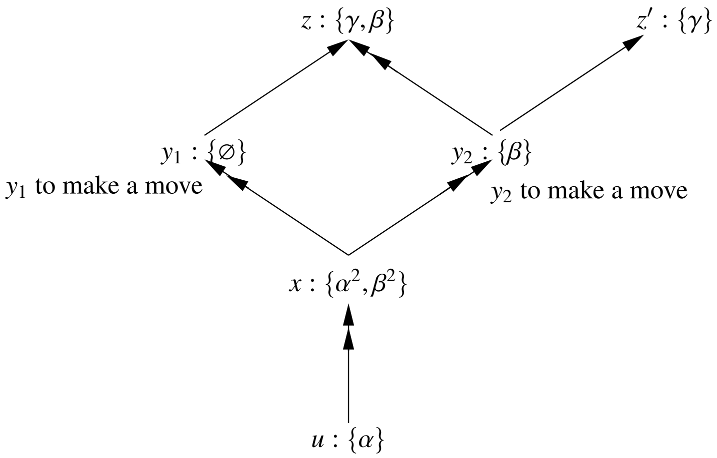


Fig. 18. The resource annotation in this Figure is $\mathbb{F}_3$

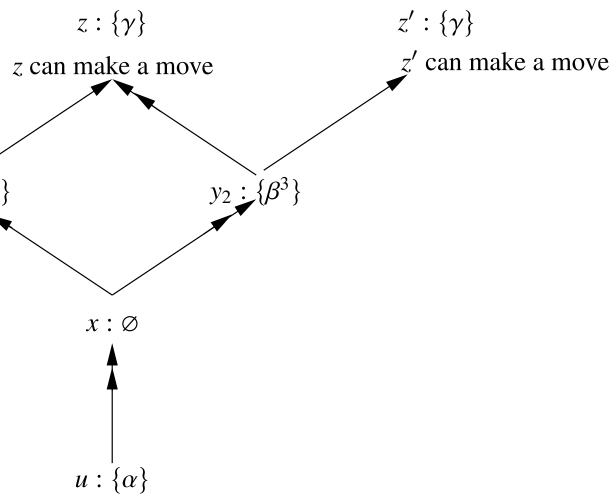


Fig. 19. The resource annotation in this Figure is $\mathbb{F}_4$

6. We can therefore say that the figure with $\mathbb{F}_1$ was transformed to the same figure with $\mathbb{F}_4$.
7. To compare with classical Dung like bipolar network, let us look at the figure again, without annotation. Consider Figure 20.

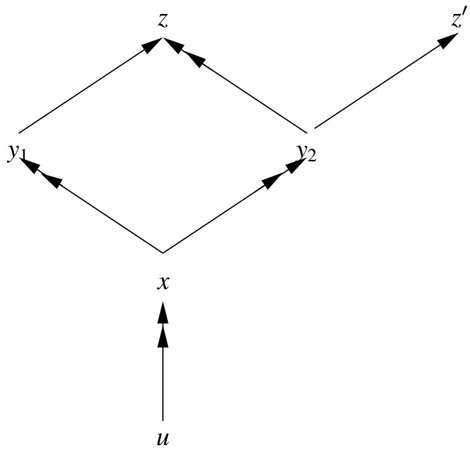


Fig. 20. This figure is viewed as traditional bipolar argumentation figure and the possible annotations are {in, out, undecided}.

Step 1. Since u is not attacked $u = \text{in}$

Step 2. Therefore $x = \text{out}$

Step 3. Therefore $y_1 = \text{in}$ and $y_2 = \text{in}$.

Step 4. Therefore $z = \text{out}, z' = \text{in}$.

Note that there are different theories for semantics of bipolar networks, some might say that also $y_1 = \text{out}$ because $z = \text{out}$.

This is not important here. We are showing process similarity.

B.2 A formal definition of the evaluation process

This subsection gives the a formal definition of the evaluation process.

Definition 15.

1. *Let $\mathcal{N} = (S, R_a, R_s)$ be a network. By a motion function $\mathbb{M}$ we mean a function of the form:*

$$\mathbb{M} : S \mapsto \left\{ \begin{array}{l} \textit{“already made a move”} \\ \textit{“ready to move now”} \\ \textit{“waiting to move”} \end{array} \right\}$$

2. $\mathbb{M}$ *is said to be a legitimate motion function on* S *iff the following conditions hold:*
   - *If* $\mathbb{M}(x)$ = *"ready to move now" and* $x \to y$ *or* $x \twoheadrightarrow y$ *then* $\mathbb{M}(y)$ *is "waiting to move".*
   - *If* $\mathbb{M}(x)$ = *"ready to move now" and* $y \to x$ *or* $y \twoheadrightarrow x$ *then* $\mathbb{M}(y)$ = *"already made a move".*
   - *If* $\mathbb{M}(x)$ = *"waiting to move" and* $x \to y$ *or* $x \twoheadrightarrow y$ *then* $\mathbb{M}(y)$ = *"waiting to move".*
   - *If* $\mathbb{M}(x)$ = *"already made a move" and* $y \to x$ *or* $y \twoheadrightarrow x$ *then* $\mathbb{M}(y)$ = *"already made a move".*

Definition 16. *Let* $(\mathcal{N}, \mathcal{A}, \mathbb{F})$ *be a weaponised argumentation network with* $\mathcal{N} = (S, R_a, R_s)$.

Let $\mathbb{M}$ *be a legitimate motion function for* $\mathcal{N}$*, as defined in Definition 15. We shall use* $\mathbb{F}$ *and* $\mathbb{M}$ *to define a new* $\mathbb{M}_1$ *and a new* $\mathbb{F}_1$ *on* S *as follows:*

1. *Let* $\{x_1, x_2, \ldots\}$ *be all points in* S *which* $\mathbb{M}$ *says they are "ready to move now". Since* $\mathbb{F}$ *is given and the geometry of* (S, R_a, R_s) *can tell each* x_i *what nodes* $u_{i,j}$*,* x_i *can attack or support (as in Definition 13), and* $\mathbb{F}(x_i)$ *can tell* x_i *how much resources* x_i *has to attack, then* x_i *can make a* **choice** $(x_i, \mathbb{F})$ *as defined in Definition 14.*
2. *Let each* x_i *attack or support each* $u_{i,j} \in$ **choice**$(x_i, \mathbb{F})$ *as described in Definition 12. Let* $\beta_{i,j}$ *be the resource which* x_i *uses to attack or support* $u_{i,j}$*. We can calculate a new* $\mathbb{F}'(x_i)$ *and a new* $\mathbb{F}'(u_{i,j})$ *as follows (using the recipe defined in item (2c) of Definition 12).*
3. *Let* y *be any node. If* y *is a target (for support or for attack) of several* x_i *(i.e.,* $y = u_{i,j}$ *for several* (i, j)*), with a resource* $\beta_{i,j}$ *let:*

$$\beta'_{i,j} = \begin{cases} \beta_{i,j} \text{ if } y \text{ is a } u_{i,j} \text{ which is a support target of } x_i \\ \beta_{i,j}^{-1}, \text{ if } y \text{ is a } u_{ij} \text{ which is an attack target of } x_i \end{cases}$$

   Let β' *be the * product of all of these* $\beta'_{i,j}$*. Then*

$$\mathbb{F}''(y) = \mathbb{F}(y) * \beta'.$$

4. *For any of the* x_i *ready to move (note that* x_i *has chosen to either attack or support each of the* $u_{i,j}$ *of its choice, respectively with* $\beta_{i,j}$*). Hence in the process of executing it choice, the node* x_i *can gain and or lose resources depending respectively on whether it chooses to attack or respectively support it target* $u_{i,j}$*. So*$\mathbb{F}(x_i)$ *lost the resource* $\beta_i = *$ *product on* j *(i.e. index* i *is fixed and* j *is the running index) of all the* $\beta_{i,j}$ *of the* $u_{i,j}$ *which it supported and gained the resource* $\beta' =$ *the* $*$ *product of all the* $\beta_{i,j}$ *of the* $u_{i,j}$ *which it attacked, respectively.*
   Let $\mathbb{F}'(x_i) = \mathbb{F}(x_i) * \beta_i^{-1} * \beta'$.
5. *Let* $\mathbb{F}_1(z), z \in S$ *be defined as follows:*

$$\mathbb{F}_1(z) = \begin{cases} \mathbb{F}'(x_i), \text{ if } z = x_i \text{ for some } x_i \\ \mathbb{F}'(u_{i,j}), \text{ if } z = u_{i,j} \text{ for some } u_{i,j} \in \textbf{choice}(x_i, \mathbb{F}) \\ \mathbb{F}(z), \text{otherwise.} \end{cases}$$

   Note that this is a good definition because $(S, R = R_a \cup R_s)$ *is finite acyclic, so* $x_i \notin$ **choice**$(x_i, \mathbb{F})$.

6. *Let $\mathbb{M}_1$ be changed from $\mathbb{M}$ as follows:*

$$\mathbb{M}_1(z) = \begin{cases} \text{"already made a move", if } z = x_i \text{ for some } x_i \\ \text{"ready to move now" if either } x_i \to z \text{ or } x_i \twoheadrightarrow z\text{, for some } x_i \\ \mathbb{M}(z), \text{ otherwise.} \end{cases}$$

Lemma 3. *$\mathbb{M}_1$ of Definition 14 is legitimate.*

Proof. Obvious from the construction.

Lemma 4. *Let $(\mathcal{N}, \mathcal{A}, \mathbb{F}_0)$ be a weaponised argumentation network and let $\mathbb{M}_{\text{bot}}$ (respectively $\mathbb{M}_{\text{top}}$) be the motion function defined by $\mathbb{M}_{\text{bot}}(x)$ = "ready to move now" if x is a bottom node (respectively top node) and $\mathbb{M}_{\text{bot}}(x)$ = "waiting to move" (respectively "already made a move") otherwise.*

Then $\mathbb{M}_{\text{bot}}$ (respectively $\mathbb{M}_{\text{top}}$) is legitimate.

Proof. Obvious.

Definition 17. *Let $\mathbf{W}_0 = (\mathcal{N}, \mathcal{A}, \mathbb{F}_0, \mathbb{M}_0)$ be a weapon system with $\mathbb{M}_0$ equal to $\mathbb{M}_{\text{bot}}$ of Lemma 4.*

A sequence $\mathbf{W}_0, \mathbf{W}_1, \ldots, \mathbf{W}_n$ is said to be a legitimate sequence if for each i, $(\mathbb{F}_{i+1}, \mathbb{M}_{i+1})$ are obtained from $(\mathbb{F}_i, \mathbb{M}_i)$ as in Definition 16.[12]

Lemma 5. *Let $W_0, W_1, \ldots, W_n \ldots$ be a legitimate sequence as in Definition 17. Then for some N $\mathbb{M}_N$ is the following*

$$\mathbb{M}_N(x) = \begin{cases} \text{"ready to move" for } x \text{ top point} \\ \text{"has already moved" otherwise} \end{cases}$$

Proof. Obvious since the "ready to move" nodes shift forward and (S, R_a, R_s) is fixed finite acyclic network.

B.3 Semantics for Attack and defense trees

This subsection will connect between the machinery of the previous two subsections and linear logic.

We will work through a series of observational remarks in first modifying linear logic and then modifying Abelian groups into a "new kind of logic" and reach a meeting point between the two (modifying) directions.

---

[12] Note that the moves of this Subsection and Definition 17 look very much like modal and temporal tableaux and the proof theory from Databases in Labelled Deductive Systems.

For the reader from argumentation with give a quick example. Suppose we have three world points, $S = \{t_1, t_2, t_3\}$ and the earlier-later relation $R = \{(t_1, t_2), (t_1, t_3), (t_2, t_3)\}$. This means that t_1 is earlier than t_2, which is earlier than t_3. Suppose we have a function $\mathbb{F}$ on S saying what is temporally true at each point of S. Let $\mathbb{M}$ be a function saying where we are, i.e. where "now" is in S. Then if "now" is t_1 and $\mathbb{F}(t_1) = \{$always in the future the formula A is true$\}$, we can add to $\mathbb{F}(t_3)$ the formula A to form the new $\mathbb{F}'(t_3) = \mathbb{F}(t_3) \cup \{A\}$. After this move, $\mathbb{F}(t_1)$ does not change.

In our case we are dealing with elements of Abelian group as resources but in the next subsection we shall show how to interpret them as linear logic.

Once we understand our systems as some sort of linear resource logic we can give it semantics.

Observational Remark 1 *Consider a logical language with the binary connective $\Rightarrow$ (intended to be intuitionistic implication or relevance implication or linear implication) and a set $G = \{g_1, g_2, \ldots\}$ of atomic generators (for generating formulas).*

1. *A wff in normal form is defined inductively as follows:*
   (a) *Any $g \in G$ is a formula. We define also the following additional meta-predicates about formulas:*
      - $L(g)$ = *the level of* $g = 0$
      - $H(g)$ = *the head of* $g = g$
      - $\mathbf{B}(g)$ = *the body of* $g = \varnothing$. *(**B** is a multiset).*
      - $\mathbf{Sub}(g)$ = *the multiset of all proper subformuas of* $g = \varnothing$.

   (b) *If $A_1 = (\mathbf{B}_1 \Rightarrow g_1), \ldots, A_n = (\mathbf{B}_n \Rightarrow g_n)$ are formulas with*
      - $\mathbf{B}(A_i) = \mathbf{B}_i, i = 1, \ldots, n$
      - $L(A_i) = m_i, i = 1, \ldots, n$
      - $H(A_i) = g_i, i = 1, \ldots, n.$

      Then we let $A = \{A_1, \ldots, A_n\} \Rightarrow g$ be a wff of level $L(A)$, where
      - $L(A) = \max\{L(A_i)\} + 1$,
        and we also let
      - $\mathbf{B}(A) = \{A_1, \ldots, A_n\}$
      - $H(A) = g$
      - $\mathbf{sub}(A) = \bigcup_{i=1}^{n} \mathbf{sub}(A_i) \cup \mathbf{B}(A)$.
2. *Let **B** be a multiset of wffs. Use the notation $(A)^n$ to indicate n copies of A. Then we can present **B** also as a set $\mathbf{B}' = \mathbf{B}$ where $\mathbf{B}' = \{(A)^n | A \text{ appears in } \mathbf{B}\ n \text{ times}\}$.*

Observational Remark 2 *Recall that implicational logic (i.e. logic for $\Rightarrow$) for intuitionistic, relevance and affine linear logic can be defined as the smallest set of theorems which satisfy the following three rules:*

- **Deduction rule**
  Database $\vdash A \Rightarrow B$ iff Database $+A \vdash B$
- **Modus ponens (detachment rule)**
  A and $A \Rightarrow B \vdash B$.
  We say A and $A \Rightarrow B$ are used in the application of this rule.
- **Restriction on resources**
  - *Intuitionistic logic. No restriction*
  - *Relevance logic. All items of data (assumptions) must be used, but can be used more than once.*
  - *Linear affine logic. Whatever assumptions are used must be used exactly once.*
  - *See [17] for examples.*

Example 5.

1. Let $A = a \Rightarrow (a \Rightarrow b)$. This formula will be written in normal form as $\{a, a\} \twoheadrightarrow b$, where $a, b \in G$. Equivalently we can write $\{(a)^2\} \Rightarrow b$.
2. We can use traditional deduction theorem and modus ponens to check whether[13] $a \Rightarrow b \vdash ?a \Rightarrow (a \Rightarrow b)$.

[13] We have not defined proof theory yet. We suggest you use what you know intuitively. This is an example.

Step 1.
- $a \Rightarrow b$ assumption
- $a \Rightarrow (a \Rightarrow b)$ to prove

Step 2. Use the deduction rule (theorem) and get
- $a \Rightarrow b$, assumption, from step 1
- a, assumption from the deduction rule
- To prove $a \Rightarrow b$.

Step 3. Use the deduction rule again
- $a \Rightarrow b$, assumption
- a, assumption
- a, assumption
- To prove b.

Step 4. We show b. From the first two assumptions, $a, a \Rightarrow b$ we get b by modus ponens. The third assumption "a" is not used.
Thus the proof is acceptable in intuitionistic logic and in affine linear logic (not all assumptions need to be used!). It is not acceptable in relevance logic (all assumptions need to be used, does not matter how many times), and not in linear logic (all assumptions need to be used exactly once).

3. Consider
$$a \Rightarrow (a \Rightarrow b) \vdash^? a \Rightarrow b.$$

Proof. Step 1.

- $a \Rightarrow (a \Rightarrow b)$, assumption
- To prove $a \Rightarrow b$

Step 2. From the deduction rule:

- $a \Rightarrow (a \Rightarrow b)$, assumption
- a, assumption
- To prove b

Step 3. Use modus ponens

- $a, a \Rightarrow (a \Rightarrow b)$, yields $a \Rightarrow b$.
- again $a, a \Rightarrow b$ yields b.

So assumption "a" was used twice. This proof is acceptable in intuitionistic and relevance logic but not in (affine) linear logic.

Observational Remark 3 *Consider item 3 of Example 5. We use the notation of item (2) of Observational Remark 1*

1. *Rewrite the example of item 3, using a different notation as follows:*
   - $(a \Rightarrow (a \Rightarrow b))^1 \vdash^? (a \Rightarrow b)$
   - *use deduction rule and get* $(a)^1, (a \Rightarrow (a \Rightarrow b))^1 \vdash^? b$

- *use modus ponens and subtract the number 1 from the exponent of $(a)^1$, i.e. because a was used. Thus we get*

$$(a)^0, (a \Rightarrow (a \Rightarrow b))^0, (a \Rightarrow b)^1 \vdash^? b$$

- *Let us not stop here, but carry on and execute another modus ponens with $(a)^0$ and $(a \Rightarrow b)^1$ by "***borrowing***" another "a" and subtracting for its use. So we get that $(a)^0$ becomes $(a)^{-1}$.*

$$(a)^{-1}, (a \Rightarrow (a \Rightarrow b))^0, (a \Rightarrow b)^0, (b)^1 \vdash^? b$$

- *success (subject to borrowing), we proved b.*

2. *What is the meaning of $(a)^{-1}$? We are operating in item 1 above in affine linear logic, so our question is in the context of linear logic. Fortunately we have encountered the meaning of this expression in the past in a different context. We now explain:*
Consider for example in linear logic, we can say that the database $\mathbf{B} = \{(a)^{-1}, b^1\}$. $(a)^{-1}$ is an anti-formula. This concept was introduced and applied by Gabbay in 2002, see Gabbay's papers [19] and [20], for completely different reasons. In fact, Gabbay was not working at the time on argumentation or security or attack and defense. Gabbay started working on argumentation around 2005). Gabbay's 2002 meaning for $(a)^{-1}$, was that of an assassin/hole in the database ready to kill/delete one "a" if it comes into the database. So $\mathbf{B} \cup \{a\} = \{b\}$, because $(a)^{-1} + a =$ nothing. Indeed Gabbay used linear logic to realise his anti-formula as $a \Rightarrow \mathbf{e}$, where $\mathbf{e}$ is the linear logic symbol for nothing. We indeed have that $(a \Rightarrow \mathbf{e}) + a = \mathbf{e} =$ nothing.
3. *If we can write $(a)^{-1}$, then we can go all the way and also write $(a)^{-2}, (a)^{-3}$, etc. In other words, use a as a generator for an Abelian group over the integers. So our database $\mathbf{B}$ in item (2) above can be viewed as a database of linear logic formulas but it can also be viewed as an element of a two generator $G = \{a, b\}$ Abelian group. Thus in Abelian group notation we have that $\mathbf{B} = \{a^{-1}, b^1\} = \{a^{-1} * b^1\}$.*
4. *Now observe carefully what is $\mathbf{B} + \{a\}$? Viewed as a linear logic database it is the database $\mathbf{B}' = \{a \Rightarrow \mathbf{e}, b\} \cup \{a\} = \{a \Rightarrow \mathbf{e}, b, a\} = \{\mathbf{e}, b\} = \{b\}$. As an element of an Abelian group it is $\mathbf{B}' = \{a^{-1}, a^1, b^1\} = \{a^{-1} * a^1, b^1\} = \{a^0, b^1\} = \{b^1\}$. So we see that addition to a database $\mathbf{B}$ is an Abelian multiplication or "support". "Borrowing" or "attacking" is multiplication by the inverse.*
5. *We now show how it is done:*
We have established that a linear logic database can be also read as an element of Abelian group. In previous sections we did various attack and defense operations with Abelian groups. Do these operations have logical meaning? can we do them in linear logic? The answer is yes. Let us start:
Consider a database $\mathbf{B}_1 = \{\beta\}$ deciding whether to attack or support another database $\mathbf{B}_2 = \{\beta, \gamma\}$.
If the decision is to support, then $\mathbf{B}_1$ sends the resource β to $\mathbf{B}_2$. So $\mathbf{B}_2$ becomes $\mathbf{B}_2' = \{\beta^2, \gamma\} = \mathbf{B}_2 * \beta$ and $\mathbf{B}_1$ loses the resources and becomes $\mathbf{B}_1' = \varnothing$.
Now suppose $\mathbf{B}_1$ decides to attack $\mathbf{B}_2$ with β. According to Definition 12, $\mathbf{B}_2$ loses β and becomes $\mathbf{B}_2'' = \{\gamma\}$ and $\mathbf{B}_1$ gains β and becomes $\mathbf{B}_1'' = \{\beta^2\}$.
The question is what does $\mathbf{B}_1$ **send** *to $\mathbf{B}_2$ to achieve this end result, and how does it gain from the action of sending? We are talking databases now and we must*

demonstrate that databases attack or defend by sending formulas. So what does $\mathbf{B}_1$ *send? Here is what* $\mathbf{B}_1$ *does:*

$$\mathbf{B}_1 = \{\beta\} = \{\beta,\beta,\beta \Rightarrow \mathbf{e}\}$$

(because $\{\beta + (\beta \Rightarrow \mathbf{e})\} = \{\mathbf{e}\} = \varnothing$*).*
$\mathbf{B}_1$ *now sends* $\{\beta \Rightarrow \mathbf{e}\}$ *into* $\mathbf{B}_2$ *which becomes*

$$\mathbf{B}_2'' = \{\gamma,\beta,\beta \Rightarrow \mathbf{e}\} = \{\gamma,\mathbf{e}\} = \{\gamma\}.$$

Having sent the damaging $\beta \Rightarrow \mathbf{E}$*,* $\mathbf{B}^1$ *is left with two copies of* β*, which is how* $\mathbf{B}^2 = \{\beta^2\}$ *gained a* β*.*

6. *Conservation of resources.*
Assume x attacks y and y attacks z Let the resources be $x = \{\alpha^2\}, y = \{\alpha^3\}, z = \{\alpha\}$*. Total resources is* α^6*.*
If x attacks y, with α^2 *resource then x steals* α^2 *from y and so the new resource* $x' = \{\alpha^4\}$ *and* $y' = \{\alpha\}$*.* y' *can attack z with* α *and we get* $y'' = \{\alpha^2\}$ *and* $z'' = \varnothing$*.*
Total resources are now $\alpha^4 * \alpha^2 = \alpha^6$*.*
If x decides to support z with α^2 *instead of attacking y we get* $x^{IV} = \varnothing, z^{IV} = \{\alpha^3\}$ *and* $y^{IV} = \{\alpha^3\}$*. y now attacks* z^{IV} *say with* α^3 *and we get* $y^v = \{\alpha^6\}$ *and* $z^V = \varnothing$*.*
Again total resources are conserved.

Having sent the damaging $\beta \Rightarrow \mathbf{e}, \mathbf{B}^1$ is left with two copies of β, which is how $\mathbf{B}^2 = \{\beta^2\}$ gained a β.

B.4 Attack and defense on *n*-dimensional real vector spaces and the real *n*-Cube

This subsection will proceed through a series of technical remarks which lead to argumentation networks on n-dimensional vector spaces and equivalently on the real cube $[0,1]^n$.

Remark 11.

1. Let $\mathcal{A} = (A, G, *, E, \mathbf{e})$ be a free Abelian group with n generators $G = \{g_1, \ldots, g_n\}$.
Then $\mathcal{A}$ can be viewed as a vector space of dimension n over the integers I.
We associate with each vector $V = (v_1, \ldots, v_n) \in I^n$ the element $\mathbf{H}(v) = g_1^{v_1} * \ldots * g_n^{v_n}$.
2. Let
   – $-V = (-v_1, \ldots, -v_n)$
   – $v_{\mathbf{e}} = (0, \ldots, 0)$
   – $V + V' = (v_1 + v_1', \ldots, v_n + v_n')$.
3. The correspondence $\mathbf{H}$ persists, namely:
   – $\mathbf{H}(-V) = \mathbf{H}(V)^{-1}$
   – $\mathbf{H}(V + V') = \mathbf{H}(V) * \mathbf{H}(V')$.

Remark 12. Taking the vector space view of Remark 11 we can talk about a finitely generated Abelian group over the real numbers, R whose elements have the form $g_1^{m_1} * \ldots * g_n^{m_n}$ where each m_i is a real number. The operations follow $\mathbf{H}$ of Remark 11.

In this case all the machinery of Section **??** works because all we use is addition, subtraction and order (> 0).

Remark 13.

1. Let m be a real number. Consider the number x

$$x = \frac{1}{1 + \frac{1}{e^m}}$$

and equivalently

$$e^m = \frac{x}{1 - x}$$

Note the following intuitive correspondence:
   - $m = +\infty$ iff $x = 1$
   - $m = 0$ iff $x = \frac{1}{2}$
   - $m = -\infty$ iff $x = 0$

   Some of these equalities are in the limit!
   So the entire $(-\infty, +\infty)$ line is in one to one correspondence with the $(0, 1)$ real interval.
   Note that the e in e^r is the exponential function (not related to the unit **e** of our Abelian group).
2. Also note the following.
   When in the Abelian group we "attack" g^{m_1} by g^{m_2}, we get as the result of the attack the element $g^{m_1 - m_2}$. We can let $r_1 = e^{m_1}$ and $r_2 = e^{m_2}$ then $r = e^{m_1 - m_2} = e^{m_1}/e^{m_2} = \frac{r_1}{r_2}$.
   So to "attack" is to divide the "r" numbers. To support would be to multiply the "r" numbers.
   But from (1) we have:

$$x_1 = \frac{1}{1+\frac{1}{e^{m_1}}}$$

$$x_2 = \frac{1}{1+\frac{1}{e^{m_2}}}$$

and when x_2 attacks x_1 we get the result

$$y = \text{ new } x_1 = \frac{1}{1 + \frac{1}{e^{m_1 - m_2}}} = \frac{1}{1 + \frac{1}{e^{m_1}/e^{m}}} = \frac{1}{1 + \frac{1}{\frac{x_1}{1-x_1}/\frac{x_2}{1-2}}}$$

   Let us write this in terms of r.
   We recall, $r_1 = e^{m_2}, r_2 = e^{m_2}$. Let $r = e^{m_1}/e^{m_2} = r_1/r_2$ and therefore $y = \frac{1}{1+\frac{1}{r_1/r_2}}$.
3. Note further the connection with the cross-ratio of projective geometry (see [16]).

Given four points on a line, say as in Figure 21 the cross ratio is $\frac{AC}{BC}/\frac{AD}{BD}$.

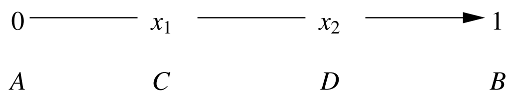
Namely

$$\frac{x_1}{1 - x_1}/\frac{x_2}{1 - x_2} = r_1/r_2 = r$$

So the attacks of x_2 on x_1 in the vector space of the Abelian group are cross-ratio attacks coordinate wise. See [3] for details.

Going back to (2), we get

$$y = \text{ new } x_1 \text{ (being the result of the attack of } x_2 \text{ on } x_1)$$
$$= \frac{1}{1+\frac{1}{r}} = \frac{1}{1+\frac{1}{\text{cross ratio of } x_1 \text{ and } x_2}}$$



The cross ratio is $\frac{AC}{BC} / \frac{AD}{BD}$

Fig. 21.

Remark 14. The correspondence between m and x of Remark 13 creates a bridge between argumentation networks and Abelian groups over the real numbers $0 \leqslant r \leqslant +\infty$.

We saw the correspondence of attacks to real numbers in the cube. We ask is this significant? Do we have real numbers in argumentation already? The answer is yes, we have plenty and especially in Gabbay's equational approach which we shall now briefly describe:

Let (S, R) be an argumentation network, not necessarily acyclic. Consider Gabbay's equational approach to argumentation (see [15]) where for each $x \in S$ and $\{y | yRx\}$ an equation over $[0, 1]$ (unit real interval) is associated as follows:

$$x = \prod_{yRx} (1 - y).$$

This system of equations has at least one solution by Brouwer fixed point theorem. Let $\mathbf{h} : S \mapsto [0, 1]$ be such a solution. Let $\lambda_{\mathbf{h}}$ be the extension (Caminada labelling) defined by

- $x = \text{in}$, if $\mathbf{h}(x) = 1$
- $x = \text{out}$, if $\mathbf{h}(x) = 0$
- $x = \text{undecided}$, if $0 < \mathbf{h}(x) < 1$.

Then the following holds:

1. $\lambda_{\mathbf{h}}$ is a legitimate Caminada labelling and hence yields an extension.
2. All preferred extensions can be obtained as $\lambda_{\mathbf{h}}$ for some $\mathbf{h}$.
3. Not all extensions can be obtained as $\lambda_{\mathbf{h}}$

Now consider an $\mathbf{h}(x)$ and consider

$$e^m = \frac{\mathbf{h}(x)}{1 - \mathbf{h}(x)}$$

Consider an Abelian group over the real numbers with a single generator g. Then we have a correspondence between x and g^m through e^m.

Remark 15. We also add that in formal argumentation real numbers are used also as weights of arguments and also as strength of arguments and as probabilities, etc. We are confident that as a by-product of our investigations we can give such uses of real numbers a meaning in Abelian groups.