FINAL AS SUBMITTED TO ARGUMENT AND COMPUTATION APRIL 05, 2017, 1930 HOURS ISRAEL TIME

Introducing Abstract Argumentation with Many Lives

D. Gabbay, G. Rozenberg and Students of CS Ashkelon

Paper 587
First draft of short version of paper 583. 31 March 2017
Compiled April 5, 2017

Abstract

We propose to view argumentation networks (of the form (S, R)) as representing a survival game. The players are the elements of S and the relation R is the attack relation. The various traditional Dung semantics for subset of S can be viewed as defining extensions in the form of possible survival groups $E \subset S$. The survival sets E (which are the traditional extensions) are groups of players which are conflict free and able to protect themselves. So far we have a different point of view on exensions which is compatible with the traditional Dung formal mathematical machinery. However, given the survival point of view we can generalise the traditional argumentation networks in new directions:

1. The research problems:
   (a) We can add to each x in S a many lives value $M(x)$, meaning how many live attackers are needed to force x to be out (i.e. x to become dead).
   (b) We associate with each attack pair (y, x) in R a value $K(y, x)$, meaning how many lives are taken out of $M(x)$ should the attack of y on x be successful (i.e. y is alive).
   (c) We can now investigate semantics for such systems (S, R, M, K).
2. The research state of the art concerning this problem: The many lives idea is new.
3. The idea used in the paper to expand on the state of the art: We view extensions as survival sets.
4. The results of the paper are: Generalise the notions of attack, conflict free, admissible and extension.
5. Relevant outside the research community: We can model the legal and therapy reasoning and procedures dealing with sex offenders.

1 Orientation: The many lives idea

Our starting point is a network (S, R), where S is a non empty set (of arguments) and R is a binary relation on S. When $(x, y) \in R$ holds we say that x

(geometrically) attacks y. Dung [6] (see Section 3) introduced several concepts related to (S, R), among them the concept of:

D1. A subset E of S attacks a node $y \in S$ iff (for some $e \in E$ we have eRy).

D2. A subset E of S is conflict free iff (for no e_1, e_2 in E do we have e_1Re_2).

D3. A subset E of S protects a node $x \in S$ iff (for all y, if yRx then E attacks y).

D4. A subset E of S is admissible iff E is conflict free and it protects all its members.

D5. A subset E is a complete extension iff E is admissible and contains all nodes it protects.

The above concepts were defined by Dung using the geometrical single attack, between x (the attacker) and y (the target), namely $(x, y) \in R$.

Our generalisation to to the above is to change D1. We introduce a function $M(x)$, for $x \in S$, giving a natural number value > 0, for each x, and changing the definition D1 into the new DM1 below:

DM1. A subset E of S attacks a node $y \in S$ iff (for some $e_i \in E$ we have e_iRy, where $i = 1, ..., M(x)$ and where $i \neq j$ implies $e_i \neq e_j$, for all $0 \leqslant i, j \leqslant M(x)$).

The function $M(x)$ gives the many lives of x, meaning how many live attackers of x we need in order to kill x.

The change of DM to DM1 necessitates changes in the other DM clauses. In other words, we need to define new corresponding clauses DM2–DM5.

To give our readers an idea of the nature of such changes, we answer some questions:

Question 1: Is traditional Dung network a special case of our new networks?

Answer to question 1: Yes, because we can let $M(x) = 1$, for all x. However, we must be careful how to define DM2–DM5, so that they also conform to the special case.

Questions 2: What happens with the concept conflict freeness? Consider a single point e which attacks itself and has 2 lives, i.e. we have $S = \{e\}$. $R = \{(e, e)\}$ and $M(e) = 2$. e is not dead because it suffers only one attack and it takes 2 attacks to kill it.

- is $\{e\}$ conflict free?
- How many lives does e have (after the attack)?

Answer to Question 2: Let us move carefully here. We have that e geometrically attacks itself but cannot kill itself. So no matter how we look at it, e cannot be dead or undecided. It is alive. To overcome this lack of clarity, let us talk about "geometrical attack" and "successful attack" of a set E on a node x. Given a subset E, we can also talk about the old Dung concept E as being "geometrical conflict free" and introduce a new concept of E as being "at peace" or as "able to survive together".

So according to these new concepts $E = \{e\}$ is able to survive together with himself because it cannot kill himself. We can also reasonably say that e has one life left now after having attacked itself.

Question 3: What is a complete extension? The example in Question 2 creates a problem because we get a new network with e geometrically attacking itself, where e has one life (one life left). Why don't we carry on attacking?

Answer to question 3: We are therefore forced to say that the new concept of a complete extension of any one network is another network, which is properly "Dung style" identified. So the network $N = (\{e\}, \{(e,e)\}, M(e) = 2)$, has the single complete extension which is the (Dung) network $N' = (\{e\}, \{(e,e)\}, M(e) = 1)$ which has the traditional complete extension $N'' = \varnothing = \{e = \text{ undecided}\}$. We immediately ask: Is this concept compatible with the old Dung concept of extension?. The answer is yes, it is.

The old Dung concept of a complete extension E is a set but E can be can be viewed as another network because it is conflict free so it is a network with the empty attack relation. So the new concept contains the old concept. This is OK.

In familiar everyday life we have many examples of the many lives/tolerance/resilience function $M(x)$ of x. These include:

1. How may complaints against x can be tolerated/covered-up/ignored before action needs to be taken
2. How many applications/demonstrations/hints/pressure/repeated nuisance, can be tolerated before compliance/giving-in.
3. How many witnesses are needed legally to establish a fact in law
4. how many violations are sufficient to cross a legal threshold to the next legal level.

We now give a major example for the idea of many lives.

There have been many cases in the UK and Israel where public figures and celebrities were accused by several alleged victims that they have been sexually abused/offended by the public figure/celebrity. All these cases and accusations had a similar pattern.

Let x be the accused. First a y_1 would come forward with allegations against x. Naturally x would deny any wrong doing and dismiss y_1's accusations. Then more and more accusers come forward, say $y_2, y_3, \ldots, y_n$. At some point, say at accuser n, the public perception will change and action/response is taken.

$$\begin{tikzcd} y_1 : M(y_1) \arrow[dr, two heads] & , \ldots, & y_k : M(y_k), \ldots \arrow[dl, two heads] \\ & x : M(x) & \end{tikzcd}$$

Figure 1: General attack formation, where "$\twoheadrightarrow$" denotes attack.

It usually starts with increased activity in social networks and may end up in social pressure on the accused to resign or pressure on the police to investigate the complaints and press charges. The next scenarios can vary from case to case; they include:

1. The public figure x resigns and disappears from the news and that is the end of the story.

2. The police investigates and the accused might end up with a prison sentence

3. Any outcome between the outcomes (1) and (2) above.

2 Formal Discussion of the Many lives idea

This section presents the many lives idea more formally and gives formal definitions of the concepts involved.

Let us focus on the number n of mny lives and consider it as the resilience of x to attacks, or in other words, the many lives $M(x)$ which x has.

$M(x)$ = how many live attackers does it take to kill x.[1]

Figure 1 indicates this basic situation. Figure 1 is a general schematic description and Figure 2 is a particular case of it. Note that is this paper double headed arrows "$\twoheadrightarrow$" denote attacks.

In this figure we assume that each node z has a value $M(z)$ of number of lives and that $y_1, \ldots, y_k$ attack x. If we have that all $y_1, \ldots, y_k$ are alive then x would be dead if $M(x) \leqslant k$. In fact we can write a formula for the new value $M^*(x)$, which is obtained after the attack of $y_1, ..., y_k$ is carried out. The value is

$M^*(x) = M(x) - k$, if $k < M(x)$ and 0 otherwise.

In particular for mathematical reasons we are going to allow the M function to give values 0. This would force us to say that $M(z) = 0$ means that z is "dead" for any z.

[1] We use here the informal words "live", "dead" and "kill". We ask the reader to understand them intuitively in this motivating section. Formal definitions will be given later in the formal sections.

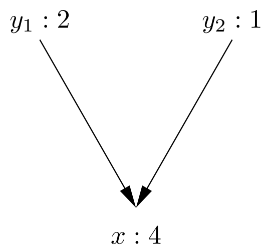

Figure 2:

So in Figure 2, the node x, which has 4 lives survives the attack of the nodes y_1 and y_2, but its number of lives is reduced from 4 to 2, because it withstood the attacks of 2 live attackers. Note that although the attacker y_1 has two lives in Figure 2, its attack on x reduces x's number of lives by 1 life only. The number of lives of y_1 indicates how many attacks can kill it, not how strongly y_1 can attack others.

We note the first two principles we are adopting here:

PP1: Every element x has a number $M(x)$ of lives (including possibly the value 0). To really kill x you need to kill it $M(x)$ times. In particular non-attacked elements retain all their many lives intact and have the capability of attacking other elements (reducing the target's number of lives) if their value is not 0.

PP2: Although an element y may have $M(y)$ lives, when attacking any x it can kill only one of x lives.[2]

Remark 2.1 *One of the students, name Shay, questioned the validity of Principle* **PP2**. *He pointed out that in real examples of complaints y against alleged sex offender x (namely $y \twoheadrightarrow x$), the strength of attack is not necessarily only 1 (i.e. killing only one of the lives of x).*

Two strong complaints can kill maybe 3 lives. It was agreed after some discussion to allow the annotation for y in the model to be of two numbers,

1. *$M(y)$ the number of lives which y has*
2. *$K(y)$, the strength of attack of y or in other words, how many lives does y take when attacking. The notation $K(y)$ assumes that the strength of attack of y is the same, no matter whom y attacks. If we want to make the strength of attack also depend on the target of y, we need to make K a function of the pairs (y, x) where y attacks x. We can write it as $K(y \twoheadrightarrow x)$.*

[2] The reader familiar with traditional Dung extensions and Caminada labelling (this shall be explained in Section 3) will note that we allow in Figure 1 to have all $M(y_i)$ as well as $M(x)$ all equal 0.

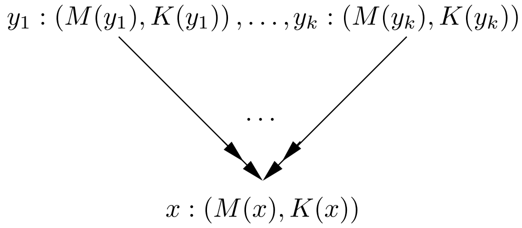

Figure 3:

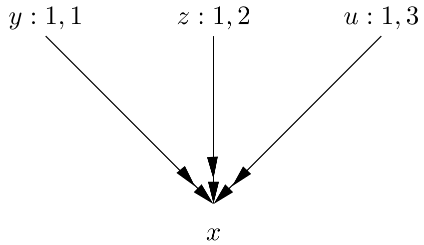

Figure 4:

So according to this model, Figure 1 will become Figure 3 and the new value $M^*(x)$ of x after the attack from all y_i would be

$$M^*(x) = M(x) \dot{-} \sum_{i=1}^{k} K(y_i).$$

Where the symbol "$\dot{-}$" is truncated substraction, namely.

$$\alpha \dot{-} \beta = \begin{cases} \alpha - \beta, & \text{if } \alpha \geqslant \beta \\ 0, & \text{if } \alpha \leqslant \beta \end{cases}$$

So for example in Figure 4, we have that y has attack strength 1, z has 2 and u has 3.

The number of lives of x is 7. So after the attack the new number of lives of x is $M^*(x)$.

$$M^*(x) = 7 - (1 + 2 + 3) = 7 - 6 = 1.$$

The students remarked that having one attack of strength 3 should be weaker than 3 attacks of strength 1. We must give extra bonus in recognition that there are more complaints (attacks) on x. We thus agree to deduct one more life if the number of attacks is more than 2.

Let k be a natural number. Define $\beta(k)$ (β for bonus) to be

$$\beta(k) = \begin{cases} 1, & \text{if } k > 2 \\ 0, & \text{otherwise} \end{cases}$$

The calculation for $M^*_\beta(x)$ of Figure 4 with bonus β is

$$M^*_\beta(x) = 7 - (1 + 2 + 3 + 1) = 0.$$

Thus with the bonus we get that x is dead.

We further remark that we have not addressed the question of a node x attacking more than one other node. For example x may attack node y and also node z. We associated the strength of attack to node x, so the attack of x on y will have the same strength as the attack on the node z. This is not true for all possible applications. In the sex offender case and in many other complaints contexts, the strength of the complaints of x against y may not be as solid and strong as the complaints on z. This means that the strength of attack needs to be associated not with the node x itself but with the attack arrows emanating from x, giving possibly different strengths to different arrows.

There are examples where the strength of attack is done by associating a number with x itself. In a survival game where the attack is done by shooting a gun, then x has a gun and x shoots always the same strength.

We shall discuss the variations in modelling the strength of attack in the formal section. We shall then put the strength of attack on the emanating arrows.

We can now, for the time being, formulate our new principle for the case of strength attached to nodes:

PP2 new: Given a network of the form (S, R, M, K) and a node x in S with k live attackers $y_1, \ldots, y_k$ of x with attack strength $K(y_1), \ldots, K(y_k)$[3] respectively, i.e. we have $M(y_i) = 0$ for $i = 1, .., k$ and m dead attackers $z_1, ..., z_m$, with $M(z_j) = 0$, for $j = 1, ..., m$, and given $M(x)$ as the number of lives of x, then the new number of lives $M^*(x)$ after the attack is given by the formula

$$M^*_\beta(x) = M(x) \dot{-} [\beta(k) + \sum_{i=1}^{k} K(y_i)] \qquad (*)$$

Definition 2.2 (Legitimate $MK\beta$ annotation for a network) *Let (S, R, M, K, β_m) be an annotated network as follows:*

1. *(S, R) is a network with $S \neq \varnothing$ and $R \subseteq S \times S$.*
2. *M is a function on S giving for each $x \in S$ a natural number in $\{0, 1, 2, 3, \ldots\}$ called the number of lives of x.*
3. *$K(x, y)$ is a function giving each attack $(y, x) \in R$ a natural number value in $\{1, 2, 3, \ldots\}$ called the strength of the attack.*
4. *β_m for m a natural number or ∞ is a function k we have $\beta_m(k) = 0$ if $k \leqslant m$ and $\beta_m(k) = 1$ if $k > m$. Let β_∞ be $\beta_\infty(k) = 0$ for all k.*
5. *Let $\delta(x)$ be Kronecker δ function, namely*

$$\delta(x) = \begin{cases} 0, & \text{if } x = 0 \\ 1, & \text{if } x \neq 0 \end{cases}$$

[3]If the strength of attack is associated with arrows we replace "$K(y)$" by "$K(y \twoheadrightarrow x)$".

6. *Let Attack(x), for $x \in S$ and subsets E of S be the set $\{y|yRx\}$. Let E be any subset of S and let Attack (E, x) be $\{y|y \in E \wedge yRx\}$.*

7. *Let $M^*(E, x)$ defined for $x \in S$ be the function derived from M, satisfying the implicit equation (*) for any subset E of S and any $x \in S$:*

$$M^*(E,x) = M(x) \dot{-} [\beta_m(\sum_{y \in E \wedge yRx} \delta(M^*(E,y))) + \sum_{y \in E \wedge yRx} (\delta(M^*(E,y))K(y,x))] \tag{$*$}$$

8. *Let $(S, R, M, K, \beta_\infty)$ be a system with $K(x, y) = 1$ for all (x, y) in R and $M(x) \leqslant 1$ for all x in S.*

We say that this system has a legitimate M labelling iff $M^* = M$.[4]

3 Background and concepts from abstract argumentation

This section presents, for the convenience of the reader, some basic concepts of what we called traditional argumentation theory. Such systems contain attacks only. We refer to such system as Dung Argumentation with Attack only (see [6]). One can also add support to the system and in this case we get systems of Argumentation with Attack and Support. We shall then explain in what way the systems developed for this paper depart from the traditional ones.

There are two traditional ways to present the semantics for the traditional Dung argumentation with attack, the traditional set theoretical approach and the Caminada labelling approach.[5] For the mapping connections between the two approaches, see [7]. Let us briefly quote the traditional set theoretic approach:

Definition 3.1

1. *We begin with a pair (S, R), where S is a nonempty set of points (arguments) and R is a binary relation on S (the "attack" relation).*

2. *Given (S, R), a subset E of S is said to be conflict free if for no x, y in E do we have xRy.*

[4]The conditions on (S, R) of item 8 makes it a traditional network with "in" and "out" annotation. If the network is acyclic a legitimate labelling exists. Note that we allow $M(x) = 0$ even for x which is not attacked (i.e. even when all attackers are non-existent or have M value 0). If we insist that $M(x) = 1$ in such cases then a legitimate M will yield the grounded stable extension in the acyclic case. Compare with Example 3.5, especially item 3 of Example 3.5.

[5]Actually there are more ways of calculating the extensions

3. The equational approach of Gabbay [9]

4. The algorithmic approach, see [1]

3. *E protects an element $a \in S$, if for every x such that xRa, there exists a $y \in E$ such that yRx holds.*

4. *E is admissible if E is conflict free and protects all of its elements.*

5. *E is a complete extension if E is admissible and contains every element which it protects.*

Various different semantics (types of extensions) can be defined by identifying different properties of E. For example we might define that E is a stable extension if E is a complete extension and for each $y \notin E$ there exists $x \in E$ such that xRy or the grounded extension as the unique minimal extension or a preferred extension, being a maximal (with respect to set inclusion) complete extension. The above properties give rise to corresponding semantics (stable semantics, grounded semantics and preferred semantics).

It can be proved that extensions satisfying items (1)- -(5) of Definition 3.1 do exist. The proof is set- theoretical using fixed points. It is easy to see how the above conditions on extensions E can be interpreted as defining a survival group. The members of the group do not attack one another and attack anyone who attacks one of them. The group also adds to itself all candidates it can protect. This is a group of nodes taking a maximal defensive position.

Remark 3.2 *Definition 3.1 uses geometrical properties (the "attack" arrow $\twoheadrightarrow$, to define survival concepts. Since later we are going to generalise the concept of one life to many lives, it is helpful already at this point to rewrite Definition 3.1 in survival terms.*

The clause numbers here correspond to the clause numbers in Definition 3.1

1. *Given (S, R), where S is a nonempty set of points and R is a binary relation on S, a subset E of S is said to attack a point x in S if for some y in E we have that yRx holds.*

2. *A subset E of S is said to be able to survive together, if for no subset Y of E and no point x in E do we have that Y attacks x.*

3. *E protects an element a in S if whenever a set X attacks a, then the set $X - Y$ does not attack a, where Y is the set*

$$Y = \{y | y \text{ in } X \text{ and } E \text{ attacks } y\}.$$

4. *E is admissible if E is able to survive together and E protects all of its elements.*

   Note for example that if we allow for many lives then if $a \twoheadrightarrow b$ and $b \twoheadrightarrow a$ and each of $\{a, b\}$ have two lives, then the set $\{a, b\}$ is able to survive together, because neither of its elements can kill the other.

5. *E is a complete extension if E is admissible and contains every element which it protects.*

We can also present the complete extensions of $A = (S, R)$, using the Caminada labelling approach, see [7].

Definition 3.3 *A Caminada labelling of S is a function* $\lambda : S \mapsto \{in,\ out,\ und\}$ *such that the following holds.*

(C1) $\lambda(x) = in$*, if for all* y *attacking* $x, \lambda(y) = out$.

(C2) $\lambda(x) = out$*, if for some* y *attacking* $x, \lambda(y) = in$.

(C3) $\lambda(x) = und$*, if for all* y *attacking* $x, \lambda(y) \neq in$*, and for some* z *attacking* $x, \lambda(z) = und$.

A consequence of (C1) in Definition 3.3 is that if x is not attacked at all, then $\lambda(x) = \text{in}$. Any Caminada labelling yields a complete extension and vice versa. Any {in, out} Caminada labelling (i.e. with no "und" value) yields a stable extension and vice versa. Set theoretic minimality or maximality conditions on extensions E correspond to the respective conditions on the "in" parts of the corresponding Caminada labellings, see [7].

Remark 3.4 *Note that both definitions 3.1 and Definition 3.3 are declarative. The proof of existence is set- theoretical. One can give an algorithm for computing such extensions, but with algorithms one needs to show that for a given* (S, R) *the parameters in the algorithm can be modified to yield all possible extensions.*

If we choose a very simple network , then we probably will not have to worry about which algorithm to choose as calculating the extensions because the different algorithms will probably be almost identical on our simple network.

Example 3.5 below is designed to compare the M function approach with the λ *function approach. We chose a very simple network with one point attacking itself, and the* M *and* λ *extensions are very easy to identify. Even for such a simple networks the conceptual differences can become apparent.*

Example 3.5 *It would be useful for us to give an example with a loop. This example will illustrate the difference between the Caminada* λ *function of Definition 3.3 and our many lives M function. Consider Figure 5. We do not yet have a general declarative definition of extensions in the case of the M function, nor do we have a general definition of a legitimate M labelling in the general case. So we use an algorithmic approach, and since this network is so simple, then whatever algorithm we choose for our general theory in a later section, will probably agree with what we are doing here.*

Note however, that item 8 of Definition 2.2, does give a declarative definition for the case of 1 life and one strength network and this can be compared with the Caminada λ.

We use a new principle, namely

- *Continue the execution (of whatever algorithm we are using) again and again until we get nothing new (it becomes stable).*

1. *First round starts with* a *having two lives, and executing the attack of* a *on itself will reduce* $a : 2$ *to* $a : 1$.

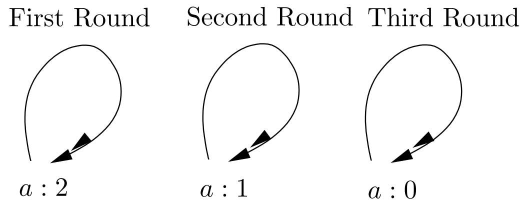

Figure 5:

2. *Second round starts with a having only one life, having survived the attack of the first round, and executing the second attack of a on itself, will turn* $a:1$ *to* $a:0$*. So the final survival picture is the graph of the third round with* $a:0$*.*

3. *Note that our many lives model with the* M *function will stop after the third round, quite content with the resulting value* $M(a)=0$*, as the final outcome of* $a:1$ *attacking itself. Traditional Dung or Caminada labelling rules cannot accept the graph of the third round. If the attackers of a (namely a itself), are all dead , then a must be alive. So Caminada forces us back to the graph of the second round and we will oscillate back and forth between the second and third rounds. This makes the value of a undecided.*

   Note that in our system there are no restrictions on the many lives annotation. So in the basic configuration of attacks described in Figure 1, the many lives function M *can give lives from the set* $\{0,1,2,...\}$ *without any restrictions, and the attack calculations use the principle* **PP2new***. In particular we can give all attackers of* x *a value 0 as well as give value 0 to* x *itself. This assignment of lives obeys principle* **PP2new***. In comparison, the Caminada function* λ *requires that if all attackers of* x *are assigned 0 lives, then* x *itself must be assigned value 1. So, for example, if we have the case of Figure 5, where a attacks itself, then a cannot be assigned any value. Thus* λ *needs to be a partial function, or equivalently, a many valued function with the additional value of "undecided".*

It is useful to introduce a familiar story as an example, the story of the party. To help us appreciate the story let us distinguish two types of attacks for a traditional network (S,R):

Definition 3.6 . *Let* (S,R) *be an argumentation network, We define two notions of attack using* R *as follows:*

- y **a**-*attacks* x *iff* yRx*. This is the traditional Dung attack notion.*
- y **d**-*attacks* x *iff* $(yRx \vee xRy)$*.*

Example 3.7 *We are planning a party and we have a set* S *which is the maximal set of all relatives friends, colleagues, etc. who can be invited to the party. The problem is that some of them do not get along/hate some others. So we* *have a relation R, where xRy (which we might denote by $x \twoheadrightarrow y$) means that if x is invited, y must not be invited. Let us also assume that for the sake of fairness, if any candidate has no people objecting to him, the candidate should be invited. For example if the party is a diplomatic event, then certainly all diplomats should be invited unless there is a problem. With this view the problem becomes an ecological kind of network (if you are not attacked you are alive). With this understanding we get here a traditional argumentation network with attack relation R. The complete extensions are possible groups of people we can invite.*

These are the traditional Dung extensions obtained by using the **a***-attack notion. If the party is a wedding, we can invite whom we please. So even if someone is not objected to, we can choose not to invite him. However if we want maximal sets of invitees we can invite any y which nobody objects to and who does not object to anyone already invited. This is the* **d** *notion of attack. So if we have $S = \{a, b, c\}$ with $a \twoheadrightarrow b \twoheadrightarrow c$, we can invite b and not invite a and c, using the* **d** *notion of attack. We cannot get this $\{b\}$ extension if we use the* **a** *notion of attack.*

Remark 3.8 *Let us summarise the comparison of the Caminada λ function (and hence the notion of the traditional Dung extension which is equivalent to it) with the many lives function $M(x)$:*
We can understand the Caminada labelling function $\lambda(x)$ a partially defined function M, giving values in $\{0, 1\}$ satisfying certain restrictions. If we write $M(x) =$ undefined when M is not defined on x, and write $M(x) =$ in, to mean $M(x) = 1$ and $M(x) =$ out to mean $M(x) = 0$, then the conditions (C1), (C2) and (C3) of Definition 3.3 become the restrictions on M.

This observation is of methodological importance. We are offering a new many lives system and we need to show how the traditional Dung system fits in as a special case. We have just shown that if we allow M to be partial function and put conditions on M in terms of R we can get the traditional Caminada Dung semantics as a special case.

4 Worked Examples

This section will give worked examples. Our purpose is to familiarise the reader with our options and to motivate the next formal section.

We give some detailed examples illustrating our concepts of many lives and the **a** and the **d** attack notions of Definition 3.6.

Example 4.1 *Consider the network in Figure 6*

Assume that each node z in this figure has lives $M(z) = 1$ and attack strength $K(y, z) = 1$ for any yRz. For such a case we have a declarative definition of a legitimate labelling as given in item 8 of Definition 2.2. Question 1 below tries to find all extensions. This can be compared with the Caminada λ labelling, and question 2 below addresses that option.

Question 1. *Take the view that the nodes are people to invite to a wedding and that the $u \twoheadrightarrow w$ attack arrow means that if we invite u we cannot invite w.*

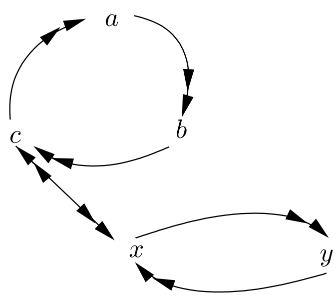

Figure 6:

List all the possible maximal sets of invitees to the wedding. (This corresponds to item 8 of Definition 2.2, using the **d** *attack notion).*

Question 2. *Assume the nodes are predator elements and that $u \twoheadrightarrow v$ means u is going to kill v. List all possible equilibrium solutions for this ecology. (This corresponds to Caminada λ functions of Definition 3.3, i.e. we use the* **a** *attack notion).*

Solution to Question 1. *Clearly any set E of invited guests to the wedding must be conflict free. We cannot have $e_1, e_2 \in E$ such that $e_1 \twoheadrightarrow e_2$. We also note that to compose a maximal list of people to invite we can start from any single candidate from among $\{a, b, c, x, y\}$ and systematically see how many more we can invite without conflict. This means that we can choose any element z in the network and be sure that there will be a maximal set of survivors containing it.*

To find all possible sets E which are maximal, let us start with each element and see who else can be safely invited.

Case a. *Invite a. Because a is invited we cannot invite b and we cannot invite c. If we want to invite more people in addition to inviting a, then since c is not invited we can invite x in which case we cannot invite y, or we can invite y in which case we cannot invite x.*

So for this case a, where we start by inviting a, we have two options for a full maximal set of invitees, namely, $E_1^1 = \{a, x\}, E_2^1 = \{a, y\}$.

Case b. *Invite b. So we cannot invite c and we cannot invite a.*

We can continue and invite more people, and invite either x and not y or invite y and not x. So in this case where we start by inviting b, we get two more options, $E_3^1 = \{b, x\}, E_4^1 = \{b, y\}$.

Case c. *Invite c. Then we cannot invite x, we cannot invite a and we cannot invite b. If we want to invite more people we invite y. We get that the maximal set to invite if we start with inviting c is $E_5^1 = \{c, y\}$.*

Case x. *Invite x. So we cannot invite y and we cannot invite c. We can now continue to invite either a and not b or b and not a. We thus get $E^1_6 = \{x, a\}, E^1_7 = \{x, b\}$.*

Case y. *Invite y. So we cannot invite x. We can continue and invite a and not b and not c or invite b and not c and not a or invite c and not a and not b. So our options are to invite $E^1_8 = \{y, a\}$, or $E^1_9 = \{y, b\}$, or $E^1_{10} = \{y, c\}$.*

Summary: our options for maximal conflict free sets of invited people to the wedding are

$$\begin{aligned}\{a, x\} &= E^1_1 = E^1_6\\ \{a, y\} &= E^1_2 = E^1_8\\ \{b, x\} &= E^1_3 = E^1_7\\ \{b, y\} &= E^1_4 = E^1_9\\ \{c, y\} &= E^1_5 = E^1_{10}\end{aligned}$$

Solution to Question 2. *Since we are dealing in this interpretation with ecological survival, then any element z which is not attacked by any live attacker y must be alive itself. This fact changes our options and the way we calculate the maximal sets of survivors. We cannot just choose any element x and be sure that there will be a maximal set of survivors containing it. Take for example an ecology of three items, a lion x, a goat y and grass z. We have that nobody attacks the lion, so he survives. The lion attacks the goat and the goat eats the grass. So the network is*

$$N = \{x, y, z\} \text{ with } x \twoheadrightarrow y \twoheadrightarrow z.$$

If we start with y, we cannot ensure it is alive. It is attacked by x and we cannot say so x is dead. To show x is dead we must show a live u which attacks x and there is no such a u. If our interpretation of N was the wedding interpretation, we could simply not invite x, but with the ecology interpretation we need a u to take x out. The ecology interpretation therefore means that any maximal set E of survivors must also satisfy the additional condition that for any y such that (i) it does not attack any $e \in E$ and also such that (ii) no $e \in E$ attacks y and further we have that (iii) E protects y, namely whenever a z attacks y then some $e \in E$ attacks z, then we must have also (iv) that $y \in E$.

Having clarified the difference between the ecology interpretation and the wedding interpretation, we can now go on to find the sets E of possible maximal survival groups. We go by cases element by element.

Case a. *Can a survive? If a is alive it will kill b. So b is dead. So can c be alive? If c were alive it would kill a in contradiction with the assumption of the case we are working on now namely that we want a alive. So c cannot be alive. So since we are dealing with the ecology interpretation, for c not to be alive we must find some live element which attacks c. Who can kill c? Answer: x must be alive. Is it possible that x is alive? In that case x kills y. So we check — can we say that x is alive and y is dead? Yes we can and indeed we get a stable solution: $E^2_1 = \{a, x\}$.*

Case b. *Can b be alive? In this case c must be dead. But then if c is dead all attackers of a are dead and so in our ecology interpretation b must be dead being attacked by a. So we have a contradiction. a must be alive and will kill b. So b cannot be alive.*

Case c. *Can c be alive? No, because then a is dead and b is alive and would attack c. So in this case also we get a contradiction.*

Case x. *Can x be alive? Yes. x alive means that we must have that c and y are dead, and so we also have that a is alive and b is dead. This is another stable consistent solution $E_1^2 = \{x, a\}$.*

Case y. *Can y be alive? Yes. y can be alive and x is dead. We now look at the cycle $H = \{c \twoheadrightarrow a \twoheadrightarrow b \twoheadrightarrow c\}$. We ask, can we add that c is alive and if we cannot, can we add that c is dead? Let us check each option.*

If x is dead, then we ask can c be alive? No, because then a is dead, b is alive and will attack c.

O.K. we now ask, can c be dead? No, c cannot be dead, because if c were dead then b would be alive (it would have no attackers) and so a would be dead. But we also have in our case that x is dead. So all attackers of c are dead so c must be alive. We get a contradiction.

So we can have $E_3^2 = \{y\}$.

Note that the analysis for **Case y** *shows that we have a problem here because we have three possibilities for a node z:*

- *z is alive*
- *z is dead (attacked by someone alive)*
- *z is unknown*

So in **Case y** *the value of the nodes $\{a, b, c\}$ is unknown/undecided. So it is more accurate to write the solution for* **Case y** *as $\{y =$ live, $x =$ dead, and $a = b = c =$ unknown/undecided$\}$. So to summarise the result for all cases , we have:*

$\{x, a =$ *alive, and all others are dead*$\}$

$\{y$ *is alive, x is dead and a, b, c unknown/undecided*$\}$.

Example 4.2 *This example makes a point about handling loops. Consider the network in Figure 7*

Let us calculate the extensions according to the invitation to a wedding point of view. Let us give each point x lives $M(x) = 1$ and strength of attack $K(x) = 1$.

We follow a case analysis as we did in solution to Question 1 in Example 4.1.

We shall give fewer details for each step since we already have some experience with this sort of calculation.

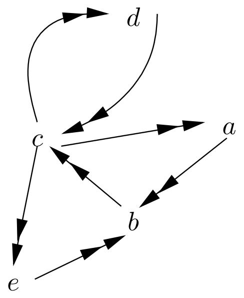

Figure 7:

Case a. *Let $a = 1$. Hence $b = 0$ and $c = 0$. Hence we can add also $d = 1$ and $e = 1$. We get*

$$E_1^1 = \{a = 1, b = 0, c = 0, d = e = 1\}.$$

Case b. *Let $b = 1$. Hence $a = 0, b = 0, e = 0$. We can add $d = 1$. We get*

$$E_2^1 = \{b = d = 1, a = b = e = 0\}.$$

Case c. *Let $c = 1$. Hence $d = 0, a = 0, b = 0, e = 0$. We get*

$$E_3^1 = \{c = 1, a = b = d = e = 0\}.$$

Case d. *Let $d = 1$. Hence $c = 0$. We can add more nodes as invitees. We have three options:*

1. *Add $a = 1$. Get $b = 0$ and we can also add $e = 1$.*
2. *Add $b = 1$. Get $a = 0, e = 0$.*
3. *Add $e = 1$. Get $b = 0$, and we can also add $a = 1$.*

We thus get the possible sets:

$$\begin{array}{l} E_4^1 = \{d = 1, c - 0, a = 1, b = 0, e = 1\} \\ E_5^1 = \{d = 1, c = 0, b = 1, a = 0, e = 0\} \\ E_6^1 = \{d = 1, c = 0, e = 1, b = 0, a = 1\}. \end{array}$$

Case e. *Let $e = 1$. Then $b = 0$ and $c = 0$. We also add $a = d = 1$. We get*

$$E_7^1 = \{a = d = e = 1, b = c = 0\}.$$

We now view Figure 7 as an ecological figure. Let us guess that node c is a key node. This is an inspirational guess that might work. There are three possible cases, c can be in, $c = 1$, or c can be out, $c = 0$ or c can be undecided.

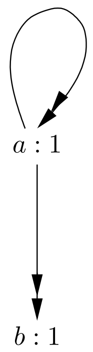

Figure 8:

Case $c = 1$**.** *In this case also* $d = 0, e = 0$ *and* $a = 0$*. We consider* b*. All of its attackers are 0 and so* b *must be 1 but it attacks* c *so how can* c *be 1?*

We get a contradiction and so the case $c = 1$ *is not possible.*

Case $c = 0$**.** *For* c *to be 0, it must be attacked by a live attacker. There are two such candidates,* b *and* d*.*

It cannot be that $b = 1$*, because our case is that* $c = 0$ *and so* $a = 1$ *since it has no attackers, and* a *attacks* b*. So our only option is that* $a = 1, b = 0$ *and* $d = 1$*.* e *can also be* $e = 1$ *since it is not attacked. Is this a consistent option? Yes, it is. We get*

$$E_1^2 = \{d = 1, c = 0, a = 1, b = 0, e = 1\}.$$

Case $c =$ **undecided.** *We note that if* z *is undecided then it cannot have an attacker which is 1. Also if an undecided* z *is the only attacker of some* x*,* x *is also undecided. So we get that also* $d = a = b = e =$ *undecided. So we have also*

$$E_2^2 = \{\textit{all undecided}\}.$$

Example 4.3 *Let us sharpen our view of Example 3.5. Consider Figure 8*

1. *The top part of this figure, namely the loop containing a attacking itself, is none other than the second round of Figure 5. We have already discussed the difference between our approach and the Caminada approach for this loop.*

   Our approach will let a attack itself and move from the figure of the second round to the figure of the third round, namely the figure with $a = 0$*. Thus the loop resolves to* $a = 0$*.*

   In the case of the Caminada λ *function, the loop will resolve to* $\lambda(a) =$ *undecided. Therefore if we adopt the view that we resolve the top loop first before we attack the second layer, namely before we attack* b*, we get in the* M *case* $b = 1$ *and in the* λ *case* $\lambda(b) =$ *und.*

2. *What happens if we do not decide to deal with the top loop first?*

   In this case, for the M approach, the only way to start is to let a attack both itself and b and we end up with $a = 0, b = 0$.

   For the λ case we get a different result computationally. The first round will give the same result as the M case, namely $\lambda(a) = \lambda(b) = 0$. But this is not an acceptable function, so we need to make it $\lambda(a) = \lambda(b) = 1$, since neither a nor b have any live attackers after the first round. We attack again (another round) and we get that $\lambda(a) = \lambda(b) = 0$ and we oscillate between these two rounds, and so we must say that $\lambda(a) = \lambda(b) =$ und.

3. *Let us now calculate the extensions as we did in previous examples, for instance as we did in Example 4.1.*

   Question 1. *Take the view that the nodes of Figure 8 are people to invite, i.e. we use the* **d** *attack notion.*

   Question 2. *Assume the nodes are predator elements, i.e. we use the* **a** *attack notion.*

   Solution to Question 1.

   Case a. *Invite a (a is in). Clearly the set $\{a\}$ is not conflict free so a cannot be invited. So $a = 0$. Who else can we invite? We can invite b. So we get $a = 0, b = 1$.*

   Case b. *Invite b. So we cannot invite a. So the solution again is $a = 0, b = 1$.*

   Solution to Question 2.

   Case a. *Can a survive? The answer is no, because it attacks itself. Can a be dead? The answer is no, because with a dead, a has no living attackers and so a must be alive. Thus a must be undecided.*

   What about b? Can b be alive, for thata must be dead. Can b be dead? For that to hold a must be alive. Since a is undecided, b is also undecided.

   Case b. *Can b be alive? For that, a must be dead, but we have argued that a must be undecided. Similarly b cannot be dead. So the only option is $b =$ und. If $b =$ und, a must also be undecided. So the only solution is $a = b =$ und.*

4. *We note that the argument in (3) is calculating options for the declarative definition and (1) and (2) are algorithmic.*

5 Formal set theoretic semantics

Our starting point is the notion of attack derived from item 7 of Definition 2.2.

Given an $MK\beta$ network as in Definition 2.2, with a set S of nodes and a relation R on S, let us define the notion of a subset E attacking a node x, Notation $\alpha_{\mathbf{a}}(E, x)$, as follows:

($\sharp$) $\alpha_{\mathbf{a}}(E, x)$ holds iff by definition $M^*(E, x) = 0$, where $M^*(E, x)$ is as defined in item 7 of Definition 2.2.

It is very important to note that for any E, E' and x we have:

- E attacks x and E is a subset of E' then E' attacks x.
- E does not attack x and E' is a subset of E then E' does not attack x.

We now continue by borrowing text from Section 7.3 of our paper [1].

Definition 5.1

1. *We say that E is* at peace *iff for no Y, a in E do we have $\alpha_{\mathbf{a}}(Y, a)$ holds ("at peace" means "able to survive together" where the attack does not kill, compare with Definition 5.6).*
2. *E protects x if for every Y such that $\alpha_{\mathbf{a}}(Y, x)$ holds we have $Y - \{y | y \in Y$ and $\alpha_{\mathbf{a}}(E, y)\}$ does not $\alpha_{\mathbf{a}}$ attack x.*

Lemma 5.2 *If E is at peace and protects its elements and E protects x then $E \cup \{x\{$ is at peace and protects its elements.*

Proof. Assume not at peace, get a contradiction.

Let $Y \subseteq E \cup \{x\}, z \in E \cup \{x\}$ be such Y **a**-attacks x.

Case 1. $x \notin Y, x \neq z$ contradicts E at peace.

Case 2. $x \notin Y, z = x$. We have Y **a**-attacks x. Since E **a**-protects x, E must **a**-attack some elements $y_1, \ldots, y_k$ such that $Y = \{y_j\}$ does not attack x. Since Y does attack x, there must be at least one y_1 s.t. E **a**-attacks y_1.

Case 3. $Y_o \cup \{x\}$ attacks z and $z \neq x$. Since $z \in E$, E **a**-attacks elements of $Y_o \cup \{x\}$. E cannot attack any elements from Y_0 so E attacks x but this is now case 1, which is impossible.

Case 4. $x \in Y, z = x$. so we have $Y_o \cup \{x\}$ attacks x. Since E protects x, E attacks $Y_0 \cup \{x\}$ but E cannot attack any of its elements. ■

Lemma 5.3 *If E admissible and protects x then $E \cup \{x\}$ protects itself because E protects all elements of $E \cup \{x\}$ so $E \cup \{x\}$ does this as well because of the monotonicity condition.*

Lemma 5.4 *There exists an admissible set $E \subseteq S$ s.t. E = all elements it protects.*

Proof. Start with $\varnothing$. It protects its elements and is at peace. Suppose $\varnothing$ protects x then $\{x\}$ protects x and is at peace.

Continue to increase the set using Lemma 5.3, until we reach a maximal st. This is the set E we need. ■

Example 5.5 *Start with $\emptyset$. It is at peace and* **p***-protects its elements. Suppose the empty set can also* **p***-attack. Suppose it protects x. Then x cannot* **a***-attack itself. If x* **a***-attacks x then $\varnothing$ must* **a***-attack x. So $\varnothing$ cannot protect x. Therefore x cannot attack x.*

Definition 5.6 *Let (S, R, M, K, β) be the network defined in Definition 2.2, and assume that we have the notion of* **a***- attack to go with it. Using the notion of* **a** *= we can identify the family of sets E which are admissible and are equal to the set of all the elements E protects. We can now use the notion of* **a** *attack to update the number of lives of each element x in E. Let x be any element x in E such that E* **d** *attacks x. Let the new annotation of x be $M * (E, x)$ of item 7 of Definition 2.2. If x is not* **a** *attacked by E, leave its annotation unchanged.*

Let M_E be the new annotation on E. We refer to the system (E, R restricted to E, M_E restricted to E) together with the **a** *respective attacks restricted to E, as an E complete extension of the original system.*

6 Comparison with the literature

We compare with two related papers.

(1). Comparison with the universal distortion paper [1]. This paper deals with thinking distortions of sex offenders in particular and of general thinking distortions in general. Part of this paper is the observation that the idea of many lives can be used in argumentation. A simple model is given in the paper and some semantics is described. The full analysis and study of many lives is postponed to the present paper.

(2). Comparison with graded acceptability of arguments paper [8]. This paper proposes a framework with a view of distinguishing between nodes that are out because of, say, two successful attacks, as opposed to nodes that are out because of, say, one successful attack. So for example, in Figure 9

which described a traditional network, d is more "dead" than c because d is attacked by two living attackers while c is attacked by only one.

The authors are trying to bring this difference out by defining a predicate $d_n^m(X)$, where X is a set of nodes (intended to be an admissible set) and $d_n^m(X)$ is the set which X protects. We use the definition for the case $d_2^1(X)$, because this is sufficient to bring out the differences with our paper and our notion of

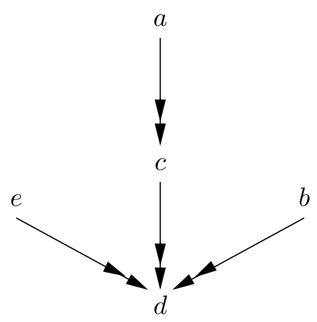

Figure 9:

many lives. SO X is a set of nodes and $d_2^1(X)$ defines the set of points which X protects.

We now quote and rewrite Definition 5 of [8] for the case $m = 1, n = 2$

$$d_2^1(X) = \{x | \neg \exists_{\geqslant 1} y([y \twoheadrightarrow x \wedge \neg \exists_{\geqslant 2} z(z \rightarrow y \wedge z \in X)]\}$$

We can rewrite the above as the following:

$$d_2^1(X) = \{x | \forall y[(y \twoheadrightarrow x) \rightarrow \exists_{\geqslant 2} z(z \twoheadrightarrow y \wedge z \in X)]\}.$$

We can again rewrite as the final version $\sharp$:

$$d_2^1(X) = \{x | \forall y[(y \twoheadrightarrow x) \rightarrow \exists z, z_2(z_1 \neq z_2 \wedge (z_1 \twoheadrightarrow y) \wedge (z_2 \twoheadrightarrow y) \wedge z_1 \in X \wedge z_2 \in X)]\}. \qquad (\sharp)$$

This formula says that x is protected by X iff every attacker y of x, $(y \twoheadrightarrow x)$ is itself attacked by two different members of X.

The above formula $d_2^1(X)$, which describes how X can protect a node x, looks very related to our two lives concept. However, it is not the same as a 2 lives. To see this, consider Figure 9. Let us apply $d_2^1(\varnothing)$ to a. This will determine whether a is alive or not. Substituting a for x in the rewritten formula, we find that $d_2^1(\varnothing)$ holds for a because a has no attackers. Thus a and similarly b and e are alive.

Let us now consider the node c. c attacks d. In order for d to be defended, c, being the attacker of d, must be attacked by two live attackers. Such attackers are not available in the figure.

However, c is being attacked by a and to get c dead it is enough to have one live attacker of c which cannot be defended.

The important point here is that we cannot assign a simple number of lives to c. For c the attacker of d, c has number of lives 2. For c the victim being attacked, the number of lives is 1. This is why d_2^1 has two indices "1" and "2". We can, however, assign two numbers to c, one for it being attacked as a victim and one for being attacked as an attacker.

We stop here. We shall study [8] critically elsewhere, see [10] and Section 7.3 of [1].

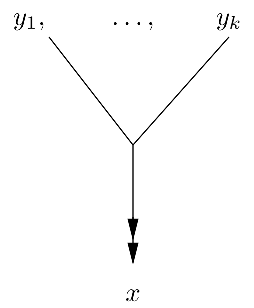

Figure 10:

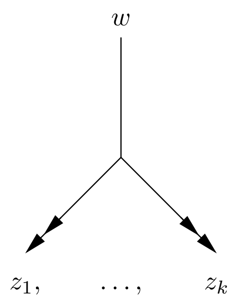

Figure 11:

(3). Comparison with joint attacks [17] and [4, Chapter 7]. The idea of joint attacks introduced in [17] and also studied in [4, Chapter 7] is explained in Figure 10. In this figure the set $Y = \{y_1, \ldots, y_k\}$ jointly attacks the node x. The meaning is that only when all $\{y_i\}$ are live (in) do we have that x is dead (out). Nielsen and Parsons in [17] use a set to point relation $\mathbb{R}$ for such an attack. So they consider networks of the form $(S, \mathbb{R}$ where $\mathbb{R} \subseteq 2^S \times S$. In Figure 10 we have $(Y, x) \in \mathbb{R}$. The notation of Figure 10 is used by [4, Chapter 7], who also allows for disjunctive attacks of the form of Figure 11.

This means that if w is alive (in), then one of $Z = \{z_1, \ldots, z_k\}$ must be out. We can also have conjunctive–disjunctive attacks of the form of Figure 12

This means that if all of $\{y_i\}$ are in then one of $\{z_j\}$ must be out. See [18]. This can be written as a relation between sets Y and Z.

The connection with m lives is explained in Figures 13 and 14.

In Figure 13, a_1, a_2 and a_3 attack b. b has 2 lives and so this figure can be translated into conjunctive attacks as shown in Figure 14 with $\{a_1, a_2\}\mathbb{R}b, \{a_1, a_3\}\mathbb{R}b$ and $\{a_3, a_2\}\mathbb{R}b$.

The translation works in this case, but what do we do with the case of b having 4 lives? What do we write? There are not enough nodes to make any distinctions.

We also cannot translate and get a network using joint attacks only and get the many lives extension with b reduced to 1 live (namely Figure 14 with b with one life only).

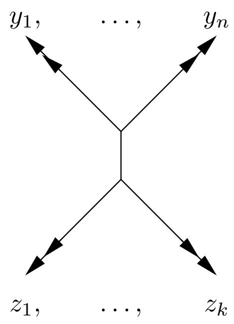

Figure 12:

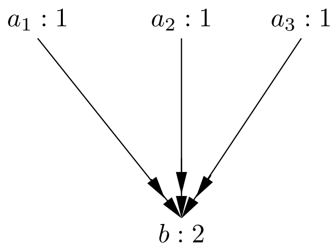

Figure 13:

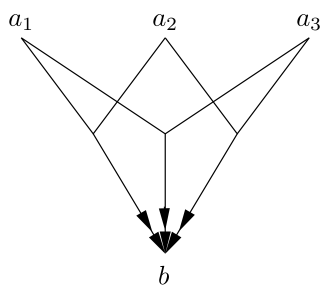

Figure 14:

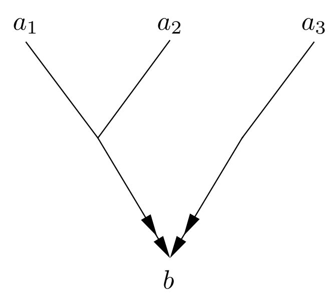

Figure 15:

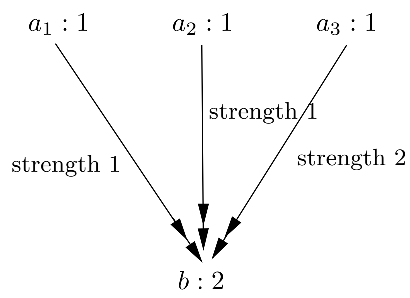

Figure 16:

Let us now ask about the other direction. Can the many lives model simulate joint attacks?

Consider Figure 15. This is a figure with two types of joint attacks.

For the attack of a_3 to succeed, we need b to have one life. For a_1 or a_2 alone not to succeed we need b to have two lives. The problem is that the joint attacks can be mixed, with different joint attacks having a different number of attackers. We can perhaps compensate by adding strength of attack to $a_3 \twoheadrightarrow b$ and get Figure 16

This may work in this case but the reader can see that the two ideas, joint attacks and many lives are different intuitions.

Consider Figure 17.

The natural translation into joint attacks is a network $(S, \mathbb{R})$ with $S = \{a, b, c\}$ and $\mathbb{R} = ((\{a, b\}, c), (\{a, c\}, b), (\{b, c\}, a)$, i.e. $\{a, b\} \twoheadrightarrow c, \{a, c\} \twoheadrightarrow b$ and $\{b, c\} \twoheadrightarrow a$.

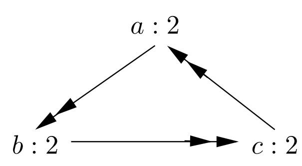

Figure 17:

$$a \twoheadrightarrow b \twoheadrightarrow c$$

Figure 18:

This has three extensions $\{a, b\}, \{b, c\}, \{a, c\}$ in the joint attacks system. However in the many lives system, Figure 17 natural extension is the 3 cycle with strength 1 to each of its elements.

(4). Comparison with abstract dialectical framework (ADF) [14]–[16]. ADF is more expressive. It can express the joint attacks, for example the condition on b of Figure 15 can be written as

$$b \leftrightarrow (\neg a_1 \vee \neg a_2) \wedge \neg a_3.$$

However, it cannot deal with Figure 17. ADF simply does not give the kind of semantics required.

As for the other direction, can the many lives approach simulate ADF or fragments of it? The answer is that the many lives model is monotonic, namely

- E attacks x and $E \subseteq E'$ then E' attacks x.

ADF does not have this restriction. We can write in ADF an acceptance condition which is not monotonic, for example:

- E attacks x if the number of elements in E is even.

Adding montonicity to ADF will not help us deal with Figure 17.

(5). Comparison with papers with the idea of graduality, e.g. [11]–[13]. These papers and many others like them want to pay attention to the number of attackers on x and the number of attackers on attackers, etc. Paying attention to such distinctions allows us to say that some nodes are "more in" than other nodes. For example in Figure 18, a is "more in" than c. This is a different theme but we can use many lives as another instrument to measure this feature. This is best explained by an example. Consider Figure 18. If we give all nodes one life we get:
One life: $a = 1, b = 0, c = 1$
Two lives: $a = 1, b = 1, c = 1$
Three lives: $a = 3, b = 2, c = 2$.

So if the network has n nodes, go in sequence up to n lives and see what you get. The differences will show in the sequence. Like the difference between a and c in the sequence for Figure 18.

Acknowledgements

We thank Pietro Baroni, Matthias Thimm, Leon van der Torre and Bart Verheij for most valuable comments.

References

[1] D. Gabbay, G. Rozenberg and L. Rivlin. Reasoning under the influence of universal distortion. To appear *Ifcolog Journal of Logic and Their Applications.*

[2] D. Gabbay and Gadi Rozenberg. Reasoning schemes, expert opinion and critical questions: Sex offenders case study. To appear in *Ifcolog Journal of Logic and their Applications*, 2017.

[3] P. Baroni, M. Giacomin, G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *J. Artificial Intelligence*, 168(1):162–210, 2005.

[4] D. Gabbay. Meta-logical Investigations in Argumentation Networks. College Publications, 2013, 770pp.

[5] H. Prakken and G. Vreeswijk. Logics for Defeasible Argumentation, in Handbook of philosophical logic vol 4, D Gabbay and F Guenthner, editors, Springer 2002, Pages 219-318

[6] P M Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games Artificial Intelligence Volume 77, Issue 2, September 1995, Pages 321-357

[7] M. Caminada and D. Gabbay. A logical account of formal argumentation. Studia Logica, 93(2-3): 109-145, 2009.

[8] D. Grossi and S. Modgil. On the graded acceptability of arguments. In *Proceedings of IJCAI'15*, to appear 2015.

[9] D. M. Gabbay Equational approach to argumentation networks *Argument and Computation* 3 (2-3):87 - 142 (2012)

[10] D. Gabbay *et al.*, 2017, Abstract Generic Argumentation (AGA).

[11] C. Cayrol and M.-Ch. Lagasquie-Schiex. Graduality in argumentation. *Journal of Artificial Intelligence Research*, 23:245–297, 2005.

[12] S. Egilmez, J. Martins, and J. Leite. Extending social abstract argumentation with votes on attacks. In *Proc. 2nd Int. Workshop on Theory and Applications of Formal Argumentation*, pages 16–31, 2013.

[13] C. Cayrol and M.-C. Lagasquie-Schiex. Gradual acceptability in argumentation systems. In *Proc 3rd CMN (International workshop on computational models of natural argument)*, Acapulco, Mexique, pages 55–58, 2003.

[14] Gerhard Brewka and Stefan Woltran. Abstract dialectical frameworks. In *Proc. KR'10*, pages 102–111. AAAI Press, 2010.

[15] G. Brewka, S. Ellmauthaler, H. Strass, J. P. Wallner, and S. Woltran. Abstract dialectical frameworks revisited. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence* (pp. 803–809). AAAI Press. (2013, August).

[16] S. Polberg. Extension based semantics of abstract dialectical frameworks. In STAIRS 2014: *Proceedings of the 7th European Starting AI Researcher Symposium* (Vol. 264, p. 240). IOS Press. (2014, August).

[17] Nielsen, SH and Parsons, S 2006, A generalization of Dung's Abstract Framework for Argumentation: Arguing with Sets of Attacking Arguments. in N Maudet, S Parsons and I Rahwan (eds), *Argumentation in Multi-Agent Systems (ArgMAS)*, 2006, pp 54–73

[18] D Gabbay and M Gabbay. Disjunctive attacks in argumentation networks, *Logic journal of IGPL* Volume 24, Issue 2, Pp. 186–218. `http://jigpal.oxfordjournals.org/content/early/2015/09/10/jigpal.jzv032.full.pdf`