1. Short GRL Overview & Analysis

GRL Elements (Intentional elements, IE's)

- Softgoal
- Goal
- Task
- Resource
- Belief

GRL Links

- Decomposition
  - *Means-end* links
  - AND/IOR/XOR decomposition
  - Relations between an IE and a set of IE
  - Decomposes(IE,{IE1,...,IEn},Type), Type = AND/IOR/XOR
- Contribution
  - Indicate *impact* of one IE's satisfaction on another IE's satisfaction.
  - Relation between any two IE
  - Valued:
    - Qualitative: make, help, some positive, non, some negative, hurt, break
    - Quantitative: domain [-100,100]
  - Contributes(IE1,IE2,Value)
- Dependency
  - Used between IE's of different actors.
  - Relation between any two IE
  - Depends(IE1,IE2)

GRL Components

- Actor
  - A grouping of intentional elements and links.

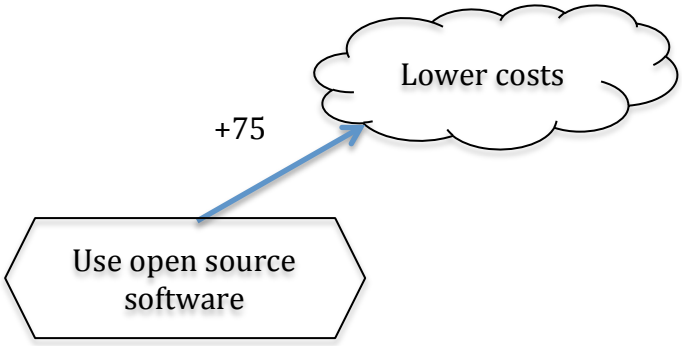
GRL Evaluation Algorithms

- Three types of evaluation algorithms:
  - *Qualitative:* Useful in early phase when quantative values are unknown
  - *Quantative*
  - *Hybrid*
- All algorithms have the following steps:
  1. Assign satisfaction values to subset of IE's (*GRL Strategy),* qualitative/quantitative
  2. Propagate values through links in the following order:
     a. Decompositions
     b. Contributions
     c. Dependencies
  3. Compute an actor's satisfaction level using an *importance* attribute

- Defines relative importance of IE over other IE's bound to an actor
- Qualitative: (H)igh, (M)edium, (L)ow, or None
- Quantative: [0,100]

2. GRL from an Argumentation Perspective

GRL models are very closely related to practical reasoning (as we discussed in RENext2015, ER2016). In fact, we can view A GRL diagram as a collection of practical reaoning arguments.



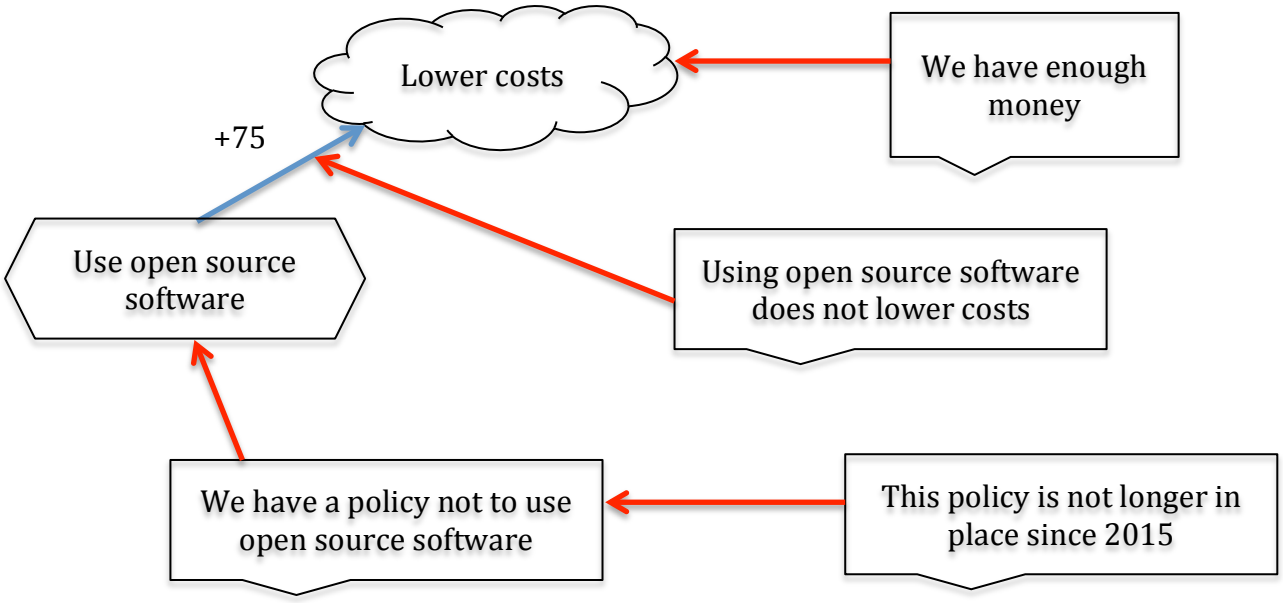
The practical reasoning argument in this case is something like:
Action "use open source software" should be performed, because performing this would promote goal "lower costs"

Such a practical argument may be attacked in a number of ways [Bench-Capon, 2003]:

1. Performing "use open source software" does not promote goal "lower costs".
2. Performing some other action that is mutually exclusive with *"use open source software"* also promotes the goal "lower costs", and this other action is more desirable.
3. Action "use open source software" cannot be performed.
4. Goal "use open source software" is not a worthy goal.

Attack (2) can possibly be handled by GRL by providing an alternative action with a higher contribution. However, it is not possible to state that two IE's are mutually exclusive **(RIGHT??).**

The following figure shows all attacks, including an extra argument attacking another argument to show that this is also possible.



Given this perspective, we can ask ourselves the following question:

When should the arguments be evaluated?

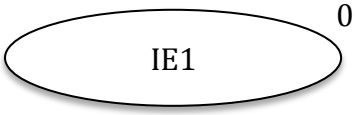
- Suggestions:
  - Arguments can be used to set the GRL Strategies, i.e. evaluation of the IE elements. So if an IE is attacked it may receive a negative evaluation. Different *argument extensions* may correspond to different GRL Strategies.
  - Another option is to remove elements from the model based on arguments. So in the example above, if the contributes-to relation is attacked, it should perhaps not be considered in the evaluation algorithms.

How to assign values to IE's based on argument extensions?

- Suggestion: As a first step, simply use binary values, i.e. +100 and -100 (quantative), or *satisfied* and *conflict* (qualitative)
- Next step: Some kind of value-based argumentation.
- Again, some elements of the GRL model may also be inactive due to arguments.

3. Examples of Arguments and GRL

Example 1: Basic Case



- IE1 is not supported or attacked by any argument, so it has evaluation 0.
- Discussion:
  - Does this make sense? If we treat IE1 itself as an argument (as we did in ER2016), then it should be IN, so it receives a positive evaluation.
  - However, intuitively giving it no evaluation seems to make more sense, because then it can be assigned a value by the evaluation algorithms.
  - If we decide that the value of IE1 is positive, then *every IE receives a non-zero initial evaluation value.* This seems very strong!

Example 2: Single argument that is IN

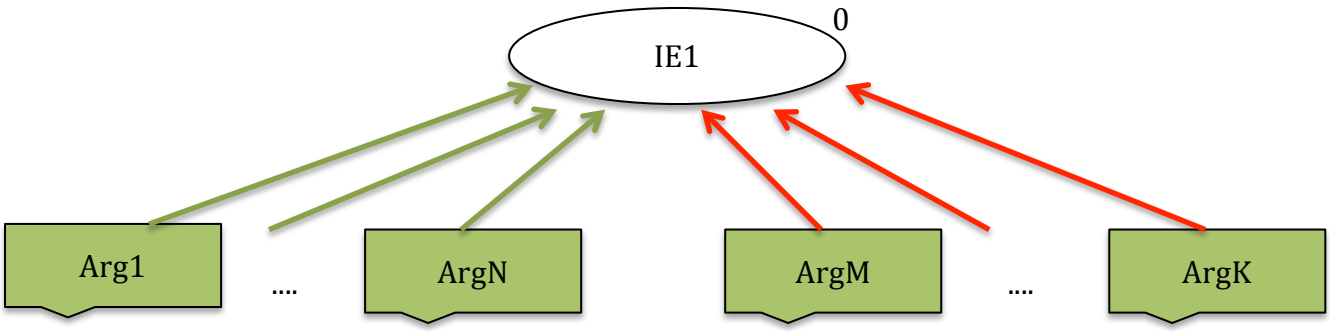
- IE1 is supported by argument 1, which is unattacked and thus IN, so IE1 receives a positive evaluation.
- IE2 is attacked by argument 2, which is unattacked and thus IN, so IE2 receives a negative evaluation.
- Discussion:
  - This makes sense to me.

Example 3: Single argument that is OUT

- IE1 is supported by argument 1, which is OUT, so IE1 receives a neutral evaluation.
- IE2 is attacked by argument 2, which is OUT, so IE2 receives a neutral evaluation.

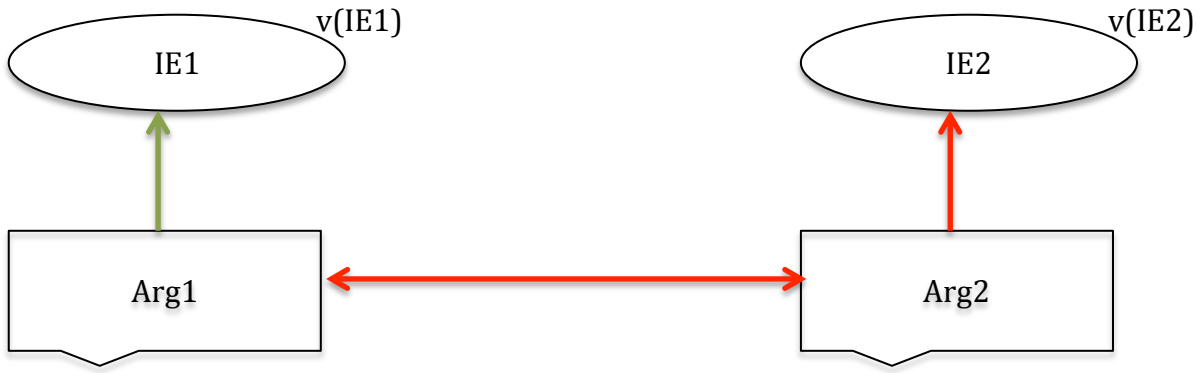
- Discussion:
  - This is related to example 1. If IE1 is supported by an argument, but this argument turns out to be invalid, what then?
  - Intuitively I'd say that there is nothing in favor of IE1 anymore, but also nothing against it. So therefore it receives 0.

Example 4: Multiple arguments for/against IE's



- There are N arguments for supporting IE (N>0) and K-M arguments against it (K-M>0).
- Since the position of IE1 is unclear, it receives a neutral evaluation
- Discussion:
  - I'm not sure what to do here.
  - We could also use some kind of counting, for instance, if there are more arguments supporting IE1 than attacking it, it receives a positive evaluation.
  - This would be a good place to combine it with valued-based argumentation.

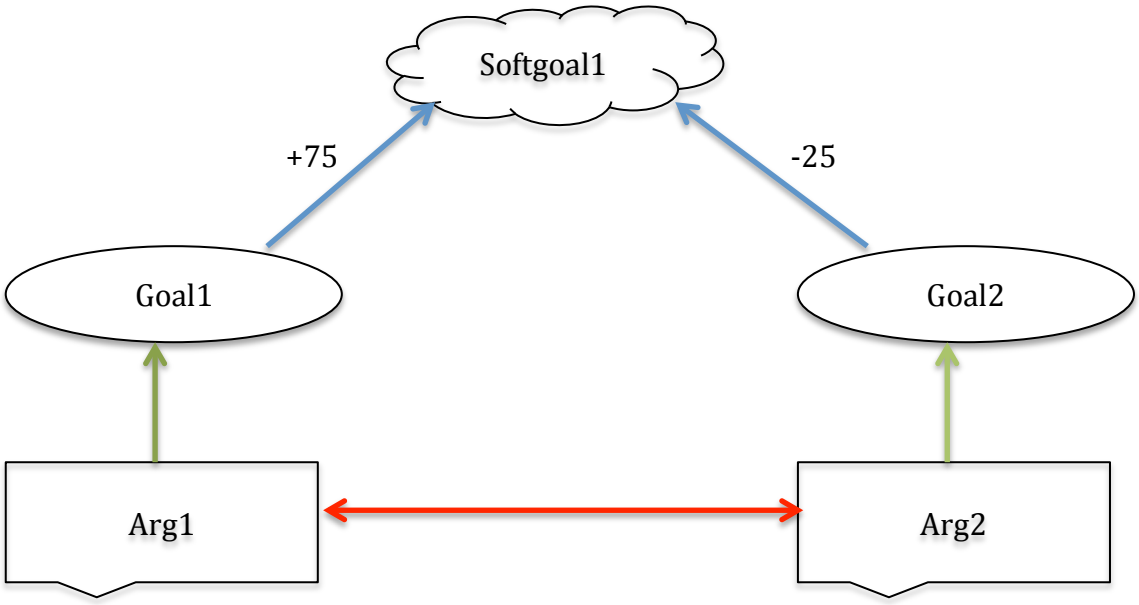
Example 5: Multiple extensions, multiple GRL strategies



- Arg1 supports IE1, and arg2 suports IE2. However, Arg1 and 2 are in conflict with each other.
- There are two extensions, corresponding to two GRL strategies:
  1. Arg1=IN, Arg2=OUT. Then v(IE1)=+100, v(IE2)=0
  2. Arg1=OUT, Arg2=IN. Then v(IE1)=0, v(IE2)=-100

- Discussion:
  - I like the idea of combining extension with strategies, it seems to fit naturally.
  - We can use different argumentation semantics, e.g., grounded, preferred, stable.

Example 6: Combination with GRL evaluation algorithms



- Goal1 contributes positively to softgoal1 (+75)
- Goal2 contributes negatively to softgoal1 (-25)
- Goal1 is supported by Arg1, and Goal2 by Arg2
- Arg1 and Arg2 are in conflict.
- Two extensions:
  - E1: Arg1=IN, Arg2=OUT
  - E2: Arg1=OUT, Arg2=IN
- So two GRL strategies:
  - S1: v(Goal1)=+100, v(Goal2)=0, v(Softgoal1)=0
  - S2: v(Goal1)=0, v(Goal1)=+100, v(Softgoal1)=0
- Applying the quantitative algorithm gives the two models below.