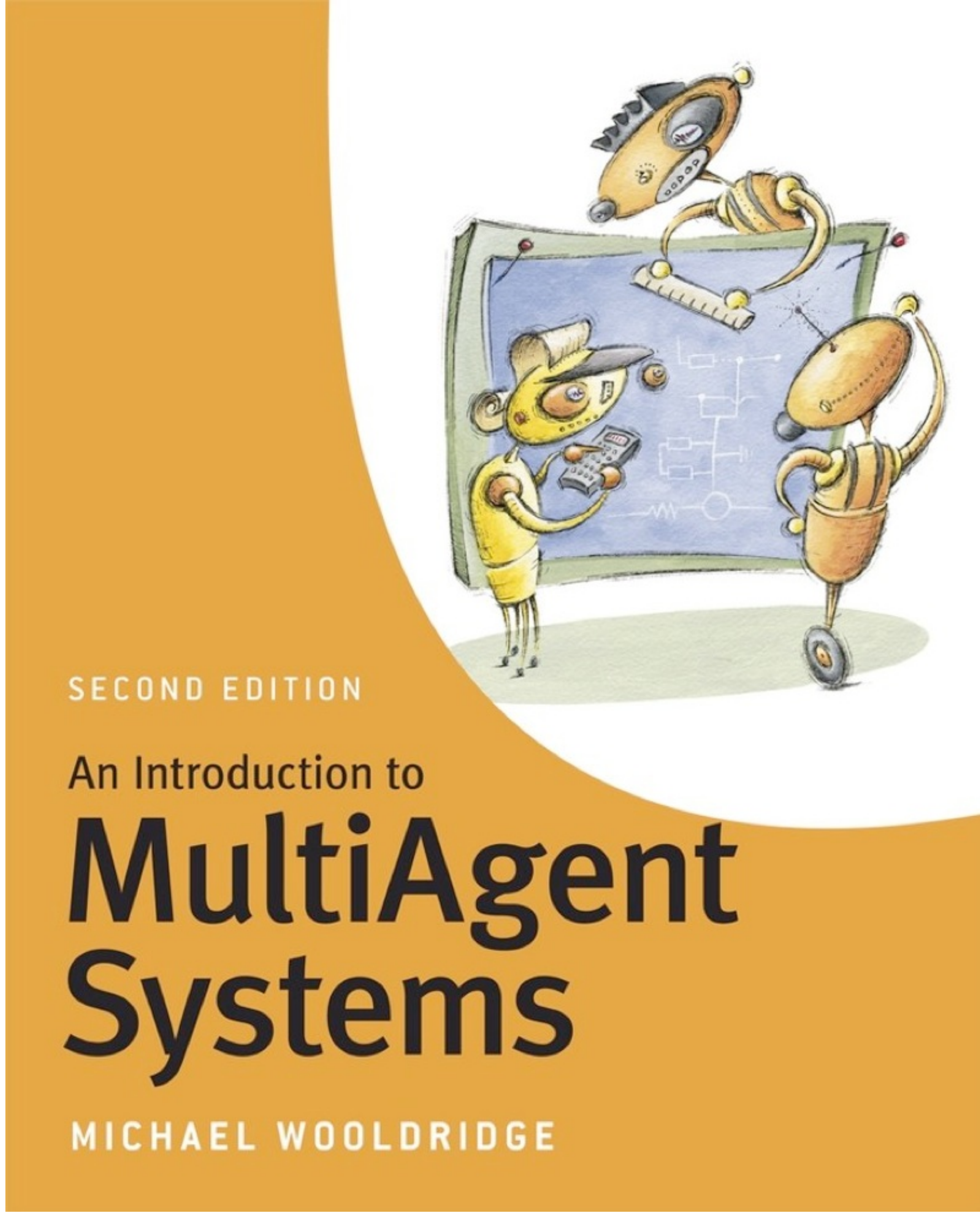
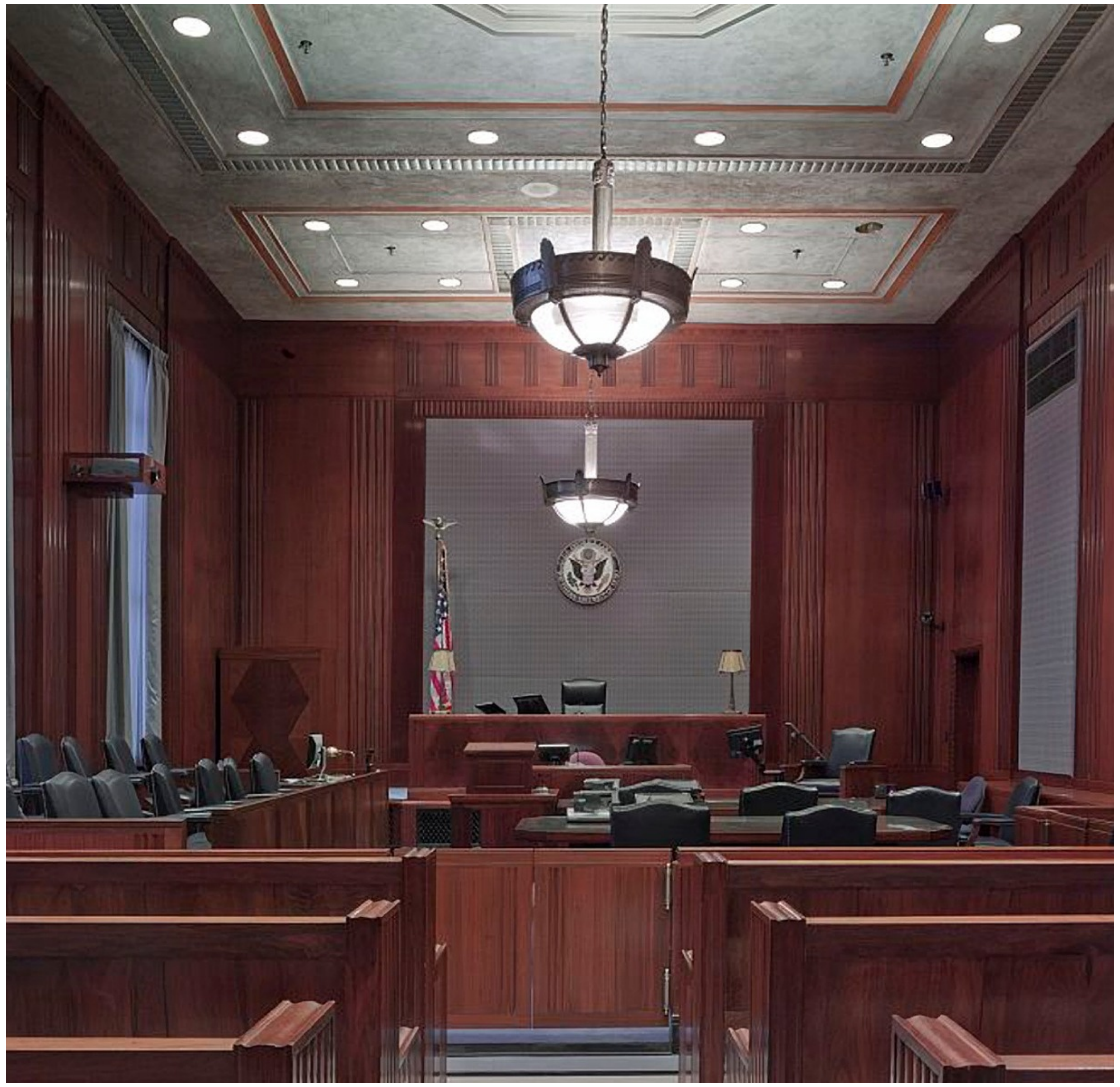
COMP310
Multi-Agent Systems
Chapter 16 - Argumentation

Dr Terry R. Payne
Department of Computer Science
UNIVERSITY OF LIVERPOOL



Overview

- How do agents ***agree on what to believe***?
  - In a court of law, barristers present a rationally justifiable position based on the arguments put forward.
    - If all of the evidence and arguments were consistent, there would be no disagreement
    - But often, the ***positions are contradictory or inconsistent***
  - We need principled techniques for dealing with inconsistency

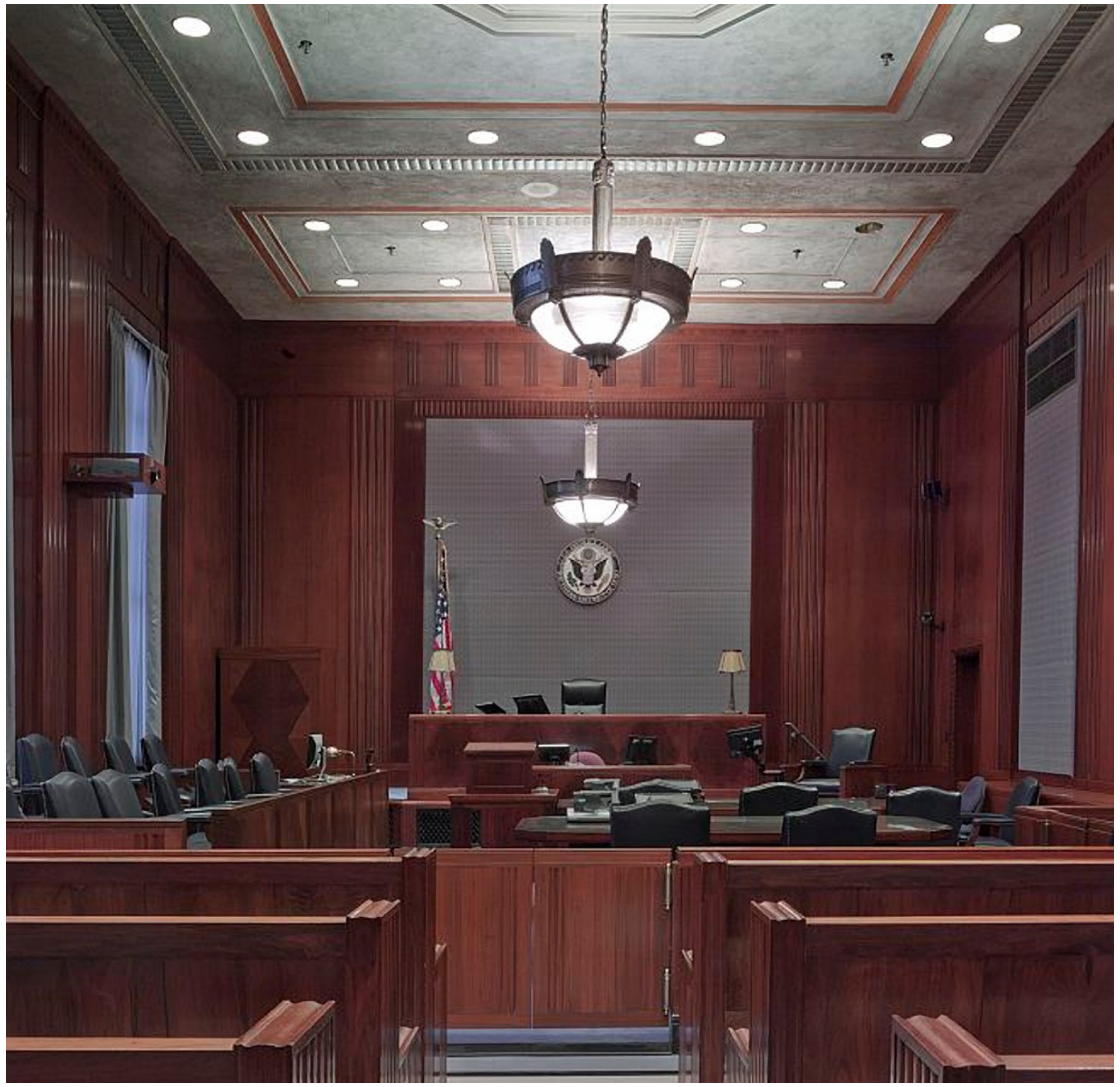


- Argumentation involves dealing with inconsistencies with beliefs of multiple agents
  - Sometimes they are obvious
    - I believe p; but you believe $\neg p$
  - Other times they are implicit
    - I believe p, and $p \longrightarrow q$. However, you believe $\neg q$

Argumentation

- Argumentation provides principled techniques for ***resolving inconsistency***.
  - Or at least, sensible rules for deciding what to believe in the face of inconsistency.

- If we are presented with p and $\neg p$ it is not clear what we should believe.
  - There can be many ***rational positions***, so which is the best?
    - If I believe p and you believe $\neg p$ then $\varnothing$ is a rational position
    - Or we could just accept one and discard the other (i.e. $\{p\}$ or $\{\neg p\}$)



Types of Argument

- Argumentation involves putting forward arguments for and against propositions
  - together with justifications for these arguments
- Michael Gilbert suggested that there are four modes of argument in human argumentation (see opposite)
  - Typically, law courts prohibit emotional and visceral modes of argumentation
  - But in other contexts (e.g. in families) emotional arguments may be permissible
- We focus here on two approaches to automated argumentation
  - ***Abstract argumentation***, which examines how arguments co-exist
  - ***Deductive argumentation***, which exploits logical reasoning

Michael Gilbert (1994) identified 4 modes of argument

- ***Logical*** mode — akin to a proof, and is deductive in nature.
  - *"If you accept that A and that A implies B, then you must accept that B".*
- ***Emotional*** mode — appeals to feelings and attitudes.
  - *"How would you feel if it happened to you?"*
- ***Visceral*** mode — physical and social aspect of human reasoning; e.g. stamping one's feet to indicate strength of feeling.
  - *"Cretin!"*
- ***Kisceral*** mode – appeals to the mystical or religious
  - *"This is against Christian teaching!"*

Abstract Argumentation

- An abstract argument system
  - A collection of arguments together with a relation "$\longrightarrow$" saying what attacks what.
    - Systems like this are called Dung-style (or Dungian) after their inventor.
    - Arguments are presented as ***abstract symbols***
    - The meaning of an argument is irrelevant
- If accepting one argument q means rejecting another argument p, we say that:
    - q attacks argument p
    - q is a counterexample of p; or
    - q is an attacker of p.
  - This can be written as (q, p) or alternatively $q \longrightarrow p$
    - However, we are not actually concerned as to what p and q are.

If this seems too abstract, here are some arguments we'll be looking at.

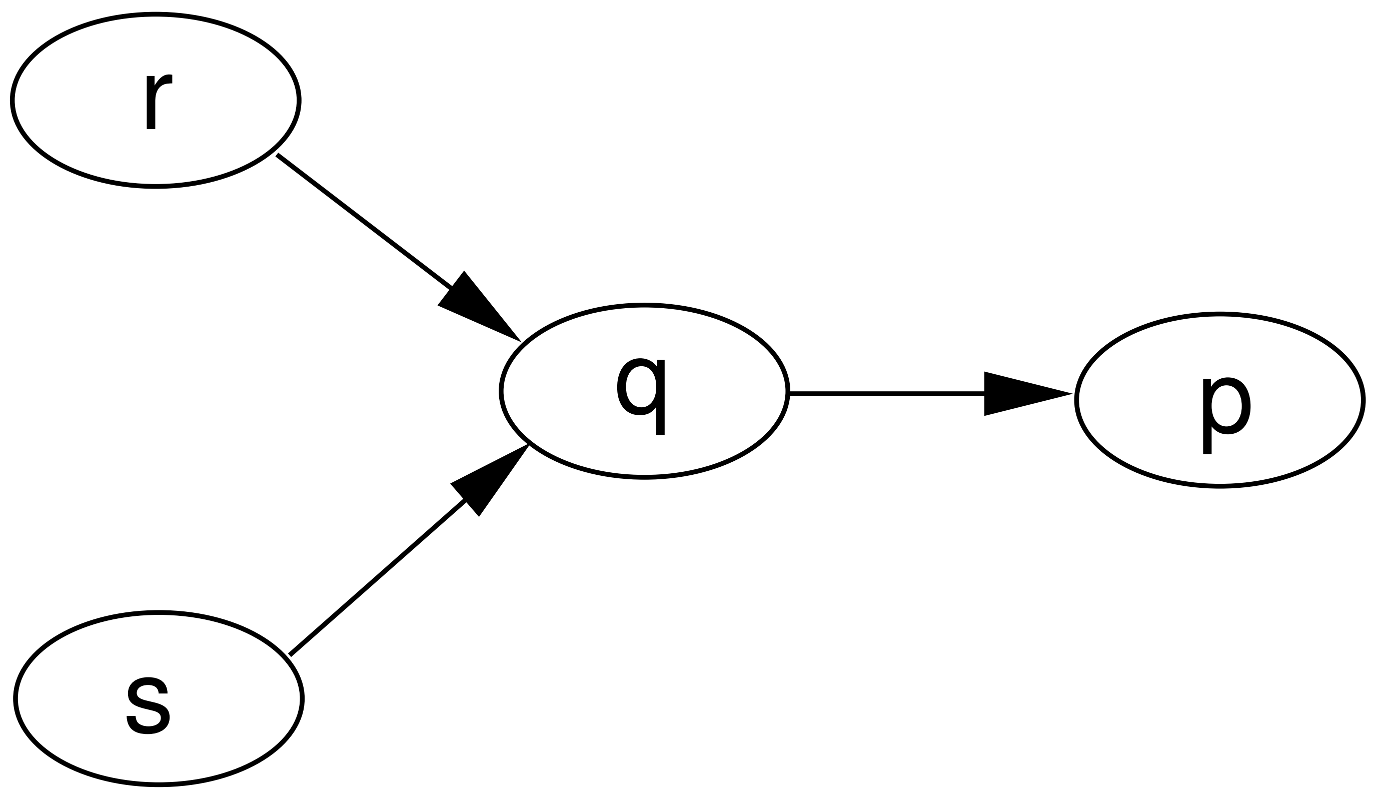
- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*

Dung's Argumentation System

- A set of Dung-style arguments is represented as a tuple $\langle \Sigma, \rhd \rangle$:
  - Σ is a (possibly inconsistent) ***set of arguments***
  - $\rhd$ is a ***set of attacks*** between arguments in Σ
  - $(\varphi, \psi) \in \rhd$ denotes the relationship: φ attacks ψ

- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*

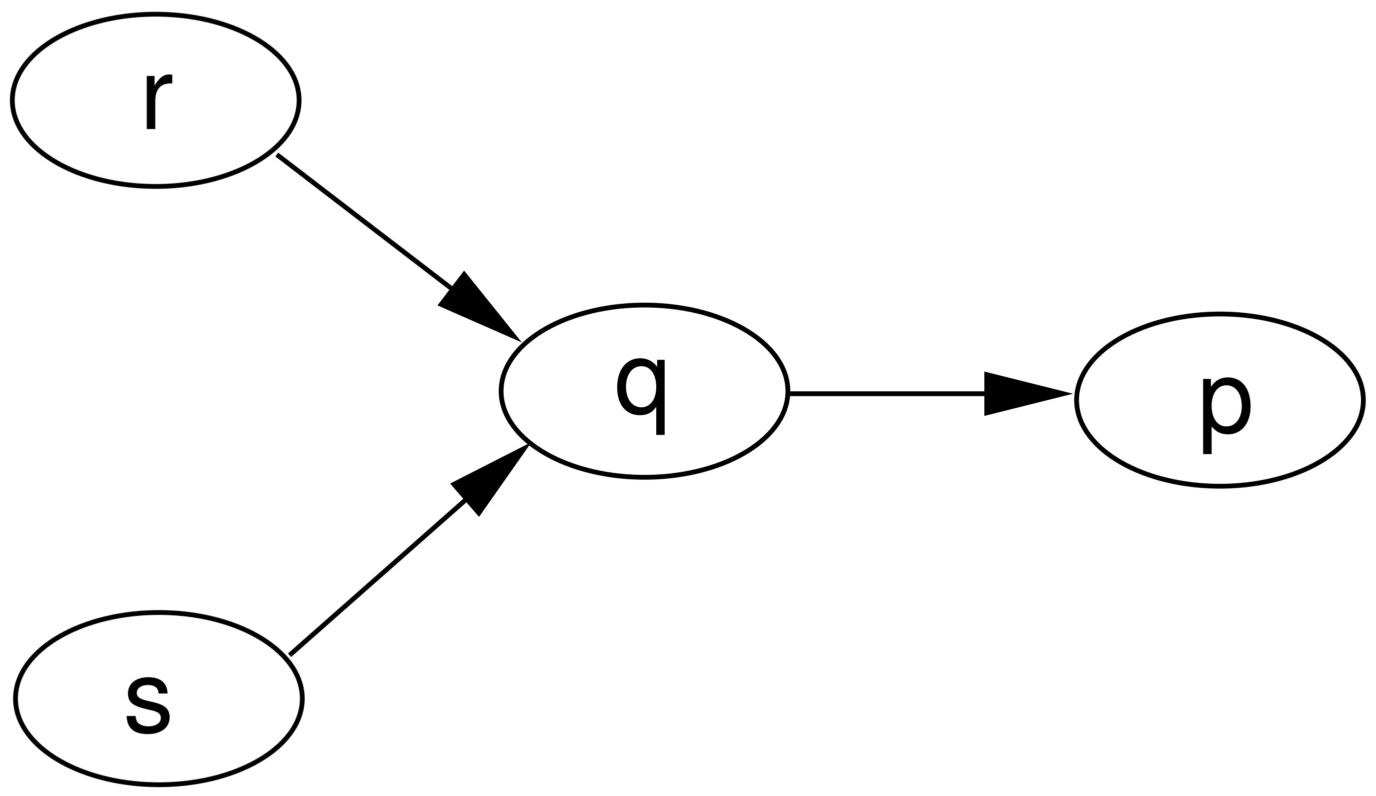
- For example: $\langle \{p, q, r, s\}, \{(r, q), (s, q), (q, p)\} \rangle$
  - There are four arguments, *p*, *q*, *r*, *s* (see opposite)
  - There are three attacks:
    - *r* attacks *q*
    - *s* attacks *q*
    - *q* attacks *p*
  - The question is, given this, what should we believe?



Conflict Free Positions

- A ***position*** S ⊆ Σ is a set of arguments
  - A position can be inconsistent - it is just a selection of arguments
- A position S is ***conflict free*** if no member of S attacks another member of S.
  - If an argument a is attacked by another a', then it is ***defended*** by a'' if a'' attacks a'
  - The position is Internally consistent
- The conflict-free sets in the previous system are:
  - ∅, {*p*}, {*q*}, {*r*}, {*s*}, {*r*, *s*}, {*p*, *r*}, {*p*, *s*}, {*r*, *s*, *p*}
  - Thus *p* is defended by *r* and *s*

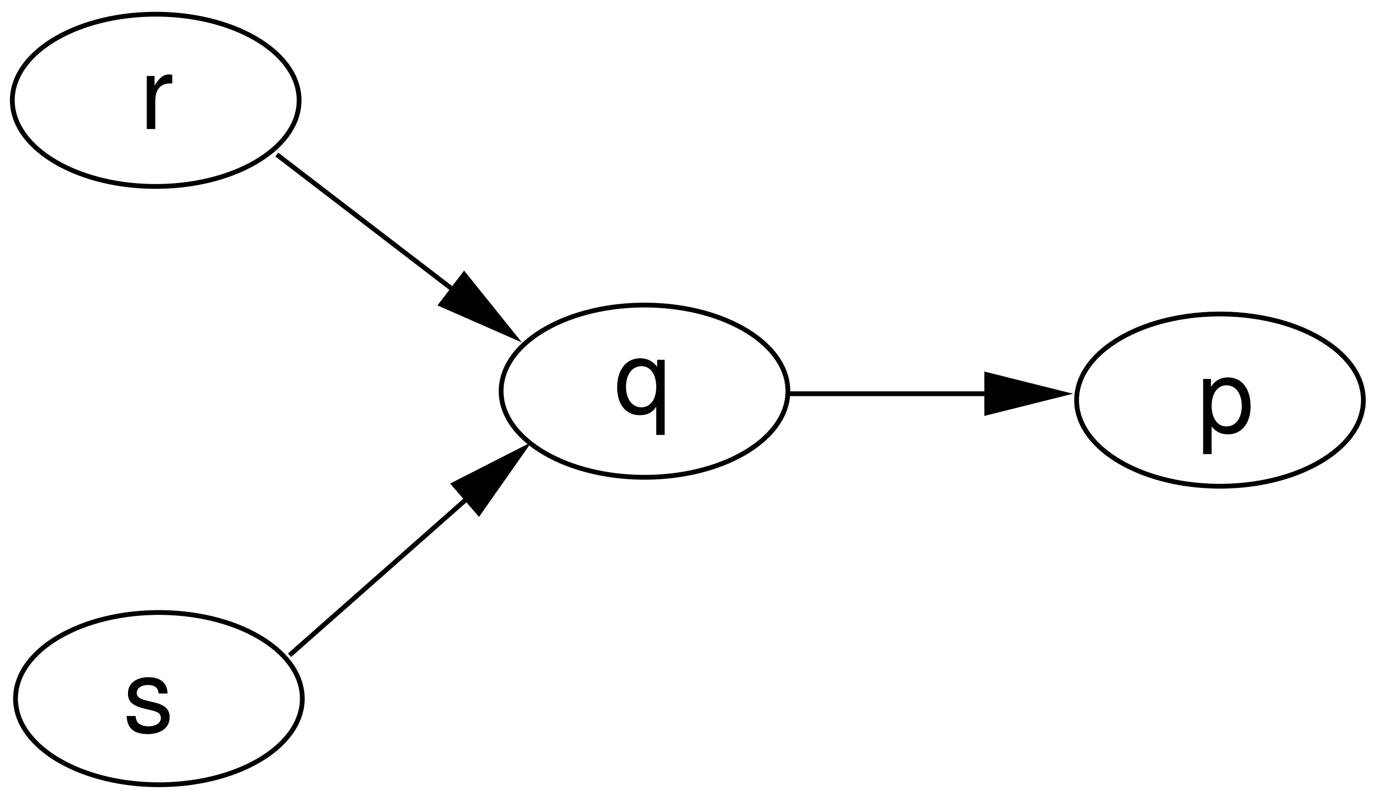
- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*



Mutually Defensive Positions

- A position S is ***mutually defensive*** if every element of S that is attacked ***is defended by some element*** of S.
  - Self-defence is allowed
- These positions are mutually defensive:
  - $\varnothing$, $\{r\}$, $\{s\}$, $\{r, s\}$, $\{p, r\}$, $\{p, s\}$, $\{r, s, p\}$
  - The position $\{p, r\}$ is defended, because if we have the case that another argument q is added to the position $\{p, r\}$, then although q attacks r, p defends r as it attacks q
- Note that $\{p\}$, $\{q\}$ are not mutually defensive
  - The position $\{p\}$ is not defended if another argument (e.g. q) is added to it

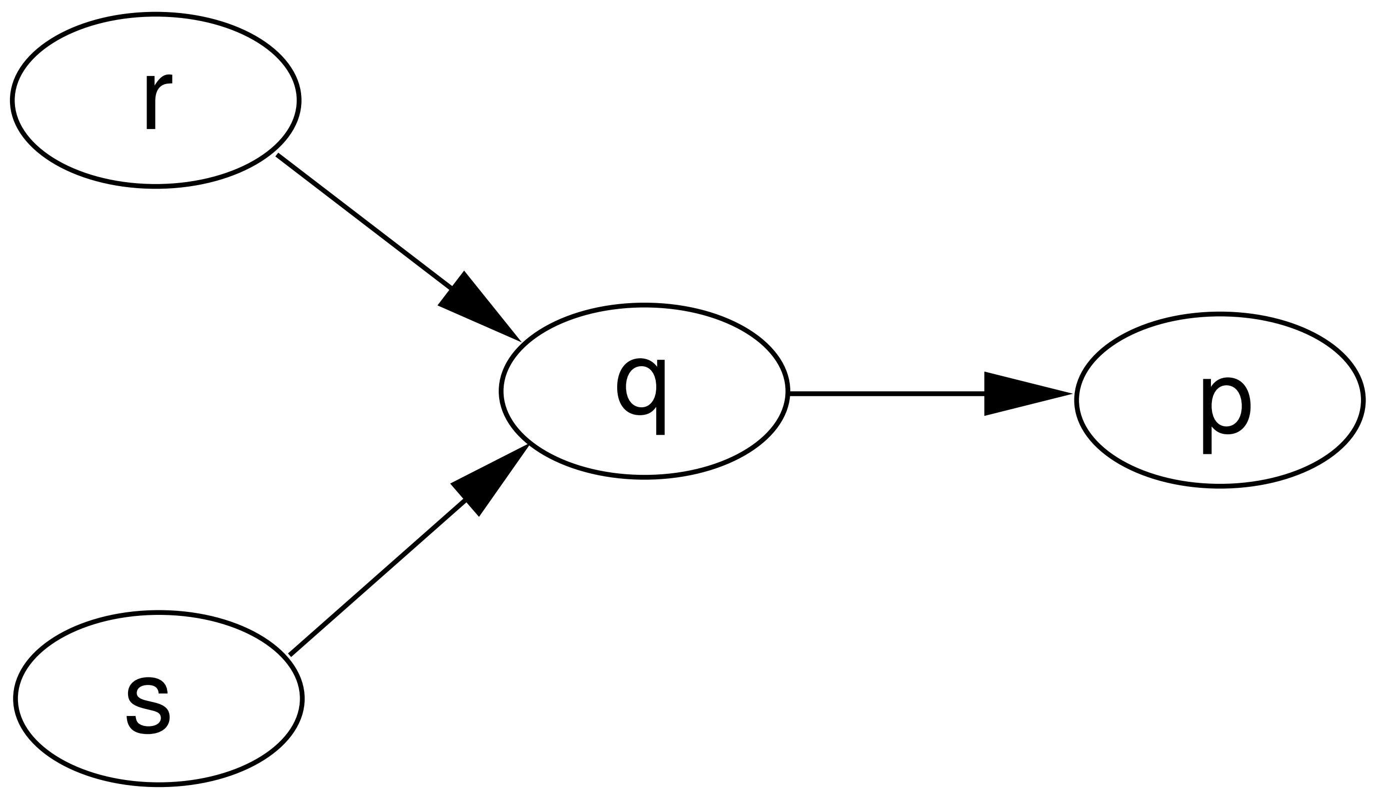
- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*



Admissible Positions

- A position that is conflict free and mutually defensive is admissible.
  - Recall that a position S is ***conflict free*** if no member of S attacks another member of S.
  - Also recall that a position S is ***mutually defensive*** if every element of S that is attacked ***is defended by some element*** of S.
- All of the following positions are admissible:
  - ∅, {*r*}, {*s*}, {*r*, *s*}, {*p*, *r*}, {*p*, *s*}, {*r*, *s*, *p*}
- Admissibility is a minimal notion of a ***reasonable position***:
  - It is internally consistent and defends itself against all attackers
  - It is a coherent, defendable position

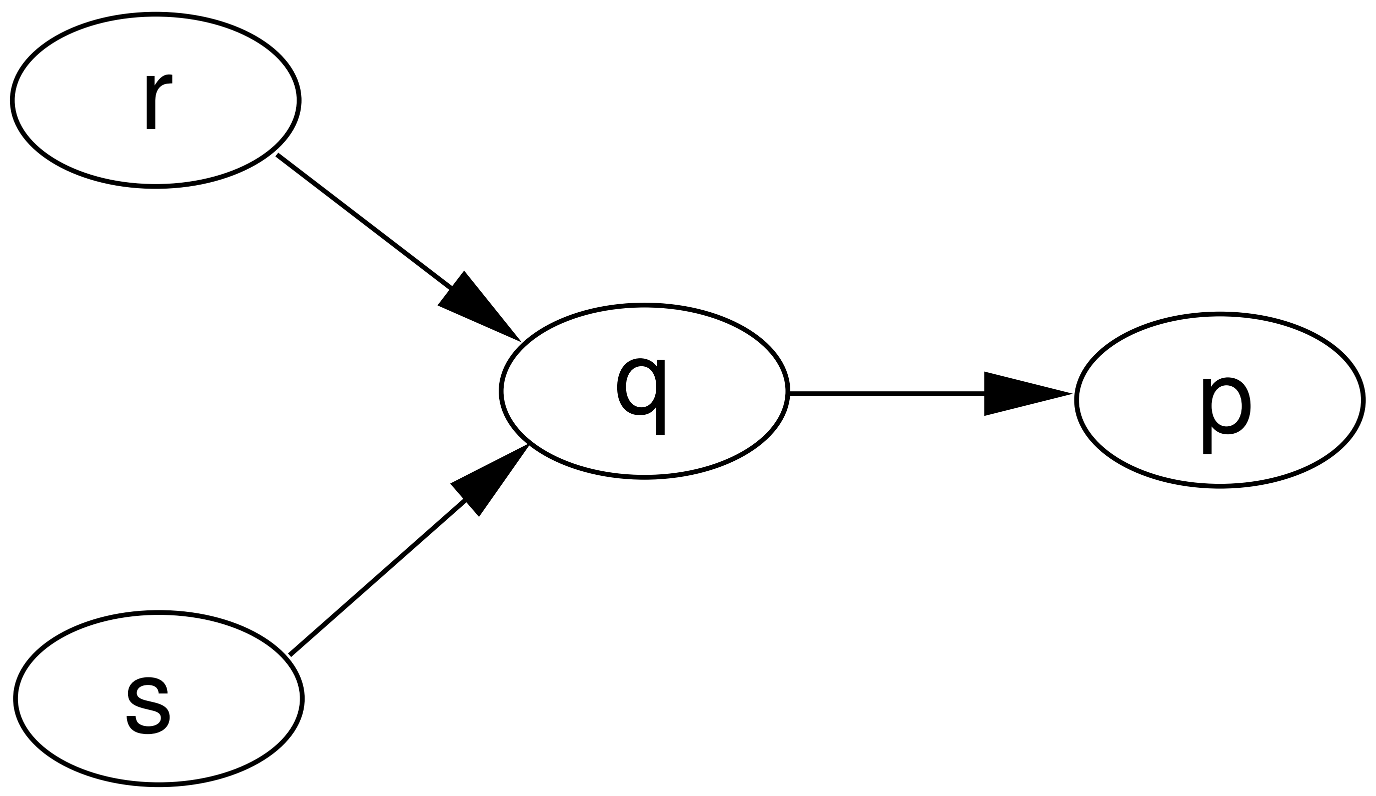
- ***p*** : *Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** : *Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** : *Since today is a holiday, I don't have to go to work.*
- ***s*** : *Since I took the day off, I don't have to go to work.*



Preferred Extension

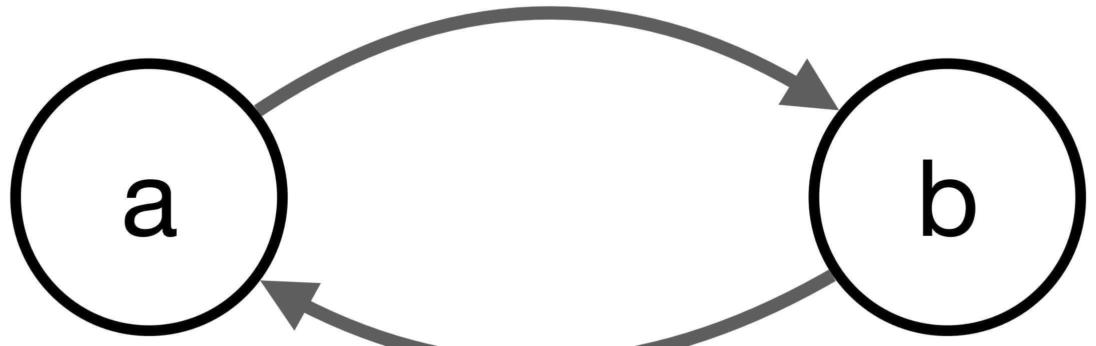
- A ***preferred extension*** is a maximal admissible set.
  - Adding another argument will make it inadmissible.
- A position S is a ***preferred extension*** if S is admissible and no superset of S is admissible.
  - Thus, in our example, ∅ is not a preferred extension, because {*p*} is admissible.
  - Similarly, {*p*, *r*, *s*} is admissible because adding *q* would make it inadmissible.
- A set of arguments ***always*** has a preferred extension
  - The empty set ∅ is always an admissible position.
  - If there are no other admissible positions, then it will be the maximal admissible set.

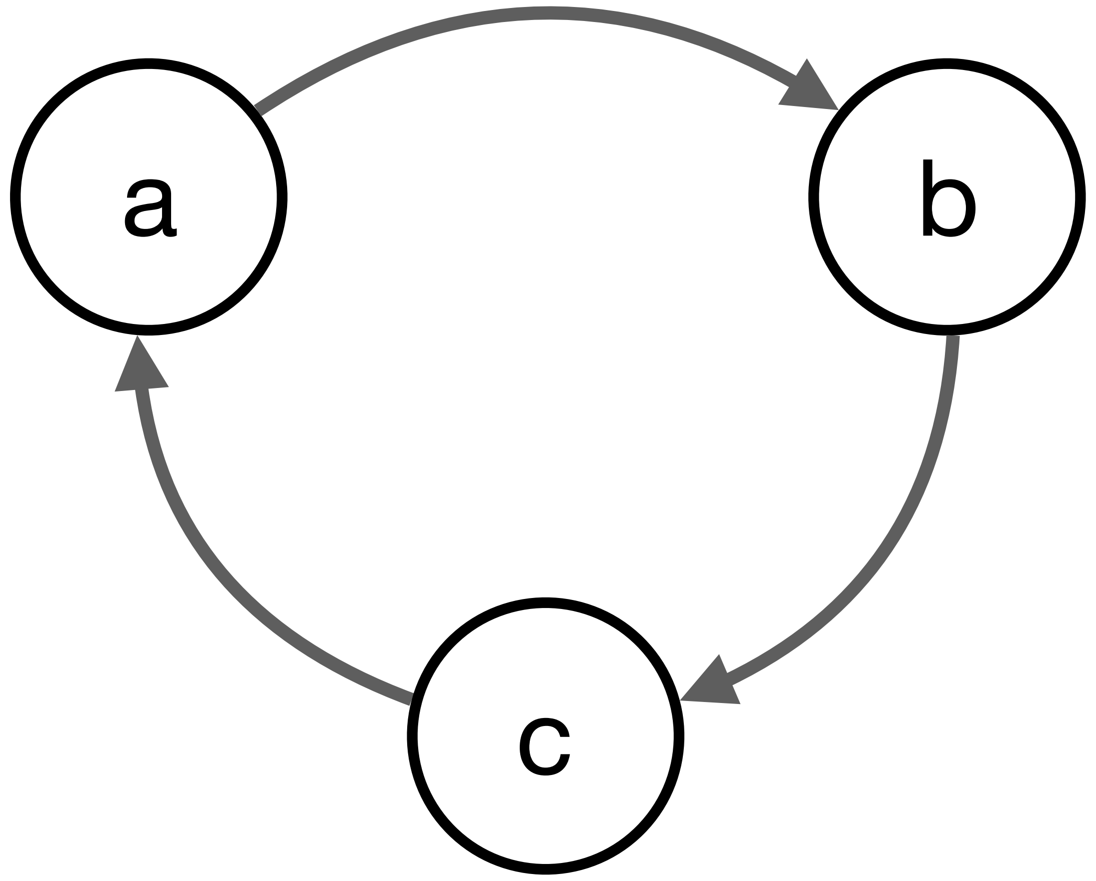
- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*

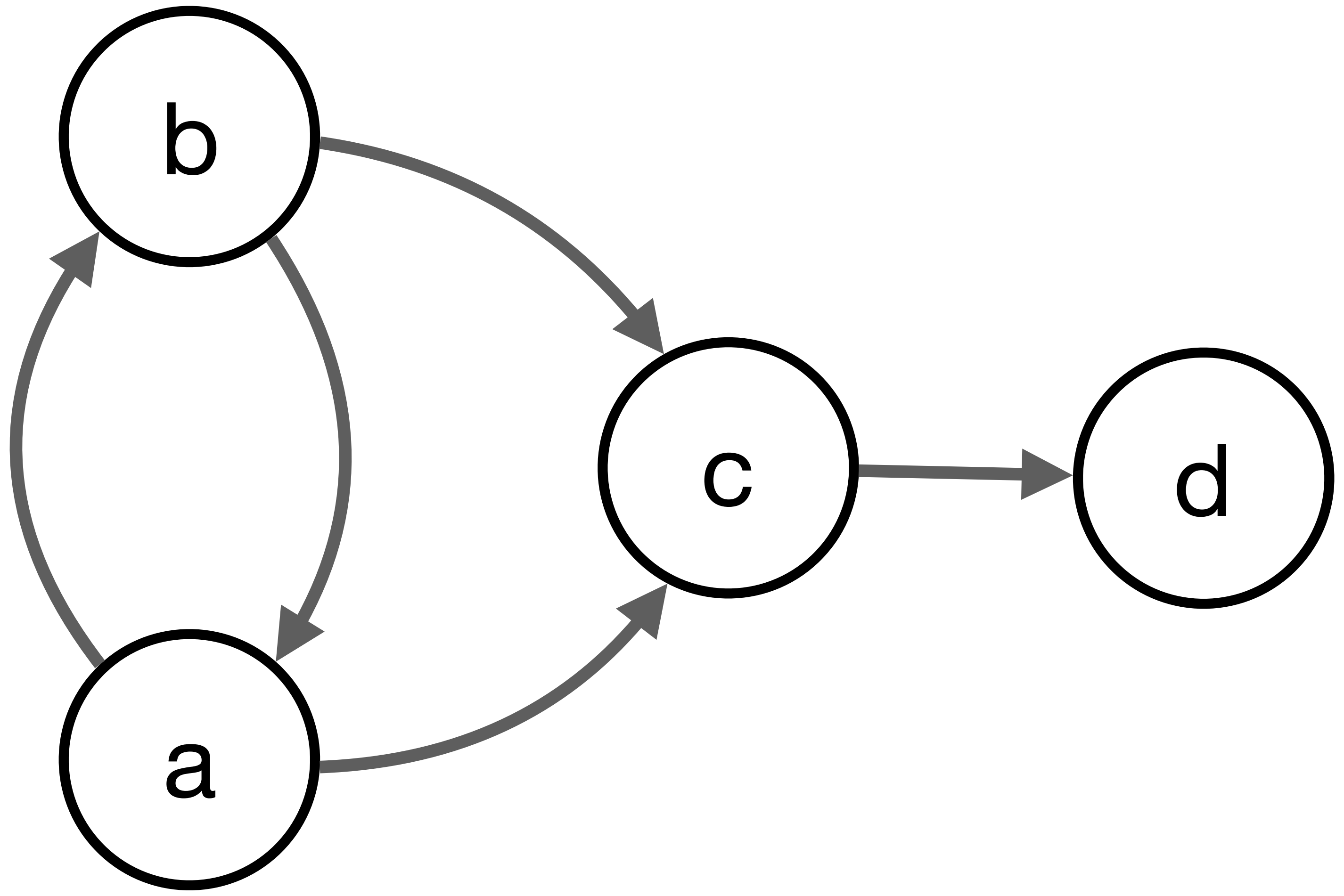


Preferred Extension

- The following examples are pathological cases







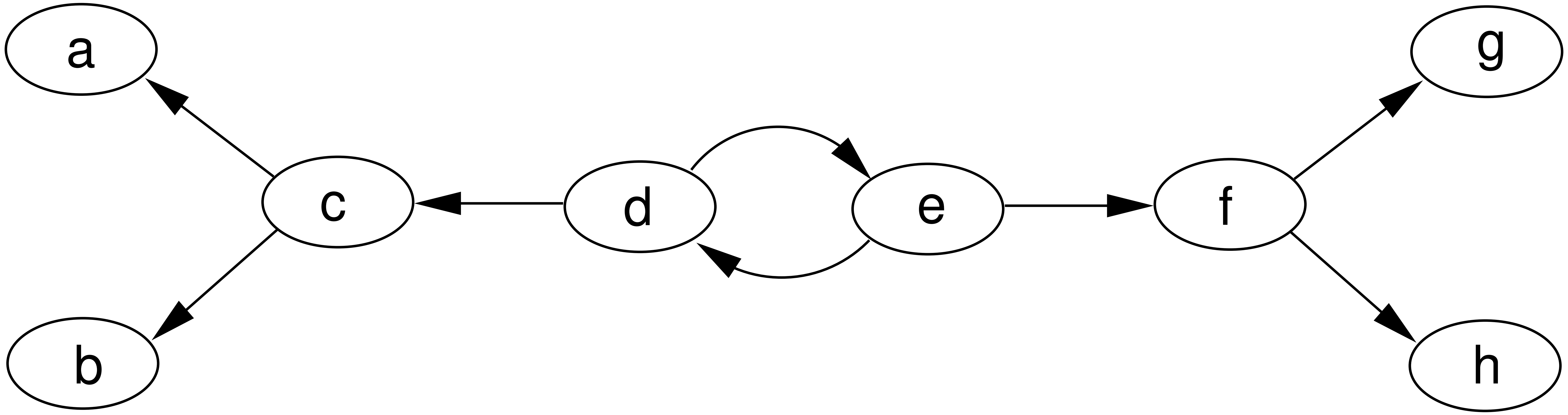
These two arguments are mutually attacking. As either could attack the other, there are two preferred extensions: {a} and {b}

With an odd number of arguments attacking in a cyclic pattern, there can be no consistent state. Thus, the preferred extension is ∅.

In this case, a and b are mutually attacking, and thus there will be at least two preferred extensions. As they both attack c, d is defended. Therefore, we have the two extensions: {a,d} and {b,d}

Preferred Extension

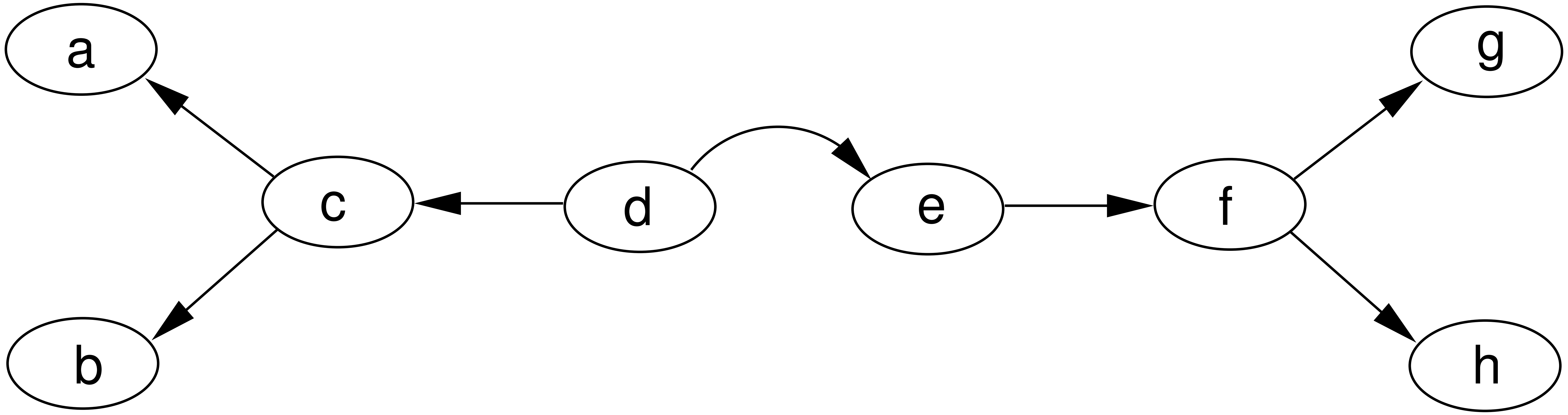
- With a larger set of arguments it is exponentially harder to find the preferred extension.
  - n arguments have 2^n possible positions.



- The set of arguments above has two preferred extensions: {*a*, *b*, *d*, *f*} and {*c*, *e*, *g*, *h*}
  - Note that *d* and *e* mutually attack each other.
  - Therefore we have two maximal admissible sets, depending on whether *d* attacks *e*, or *e* attacks *d*

Preferred Extension

- In contrast:

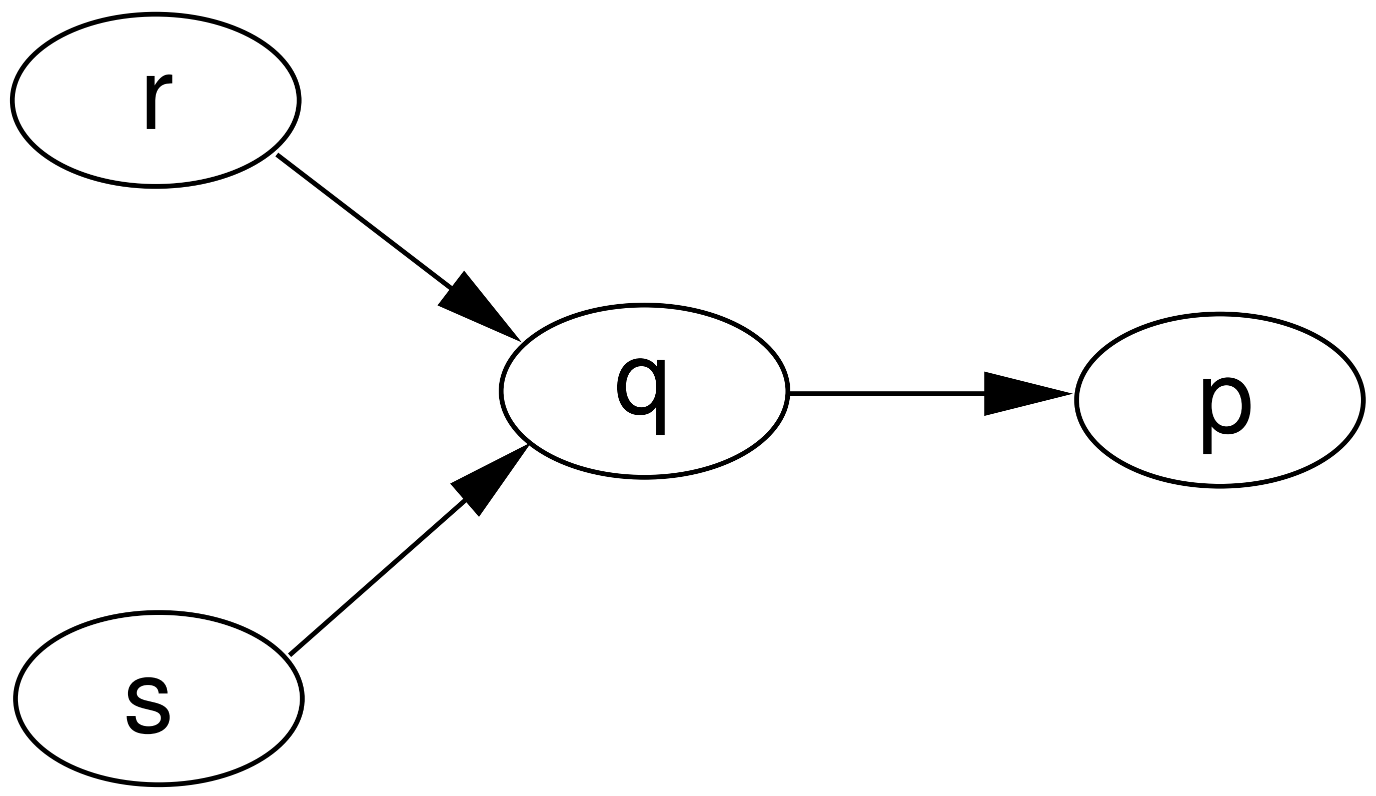


- The set of arguments above has only one preferred extension: {*a*, *b*, *d*, *f*}
  - Both *c* and *e* are now attacked by *d* and neither are defended
  - Therefore neither can be within an admissible set

Credulous and sceptical acceptance

- To improve on preferred extensions we can define:
  - An argument is ***sceptically*** accepted if it is a member of ***every*** preferred extension; and
  - An argument is ***credulously*** accepted if it is a member of ***at least one*** preferred extension
- Clearly anything that is sceptically accepted is also credulously accepted.
  - In our original example:
    - *p*, *r* and *s* are all sceptically accepted
    - *q* is neither sceptically or credulously accepted

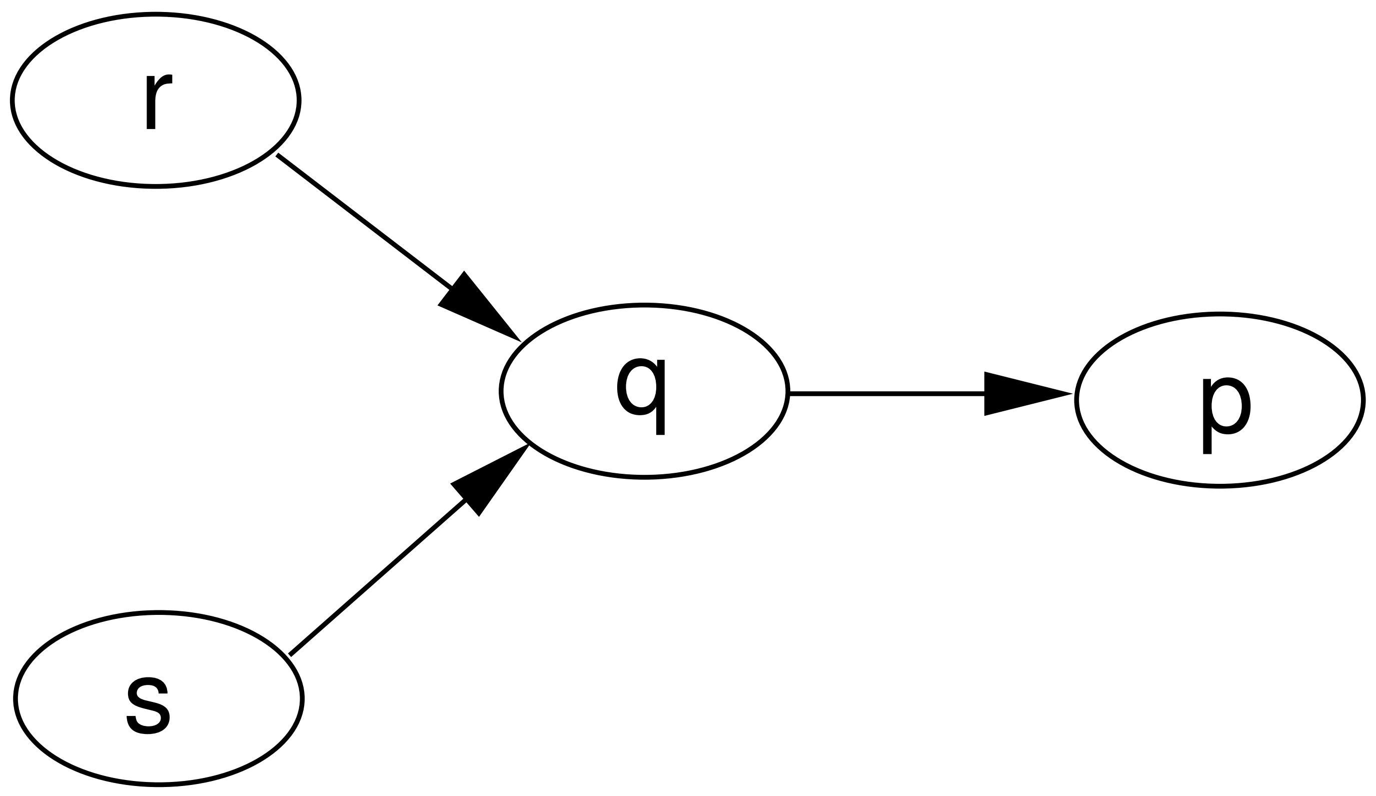
- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*



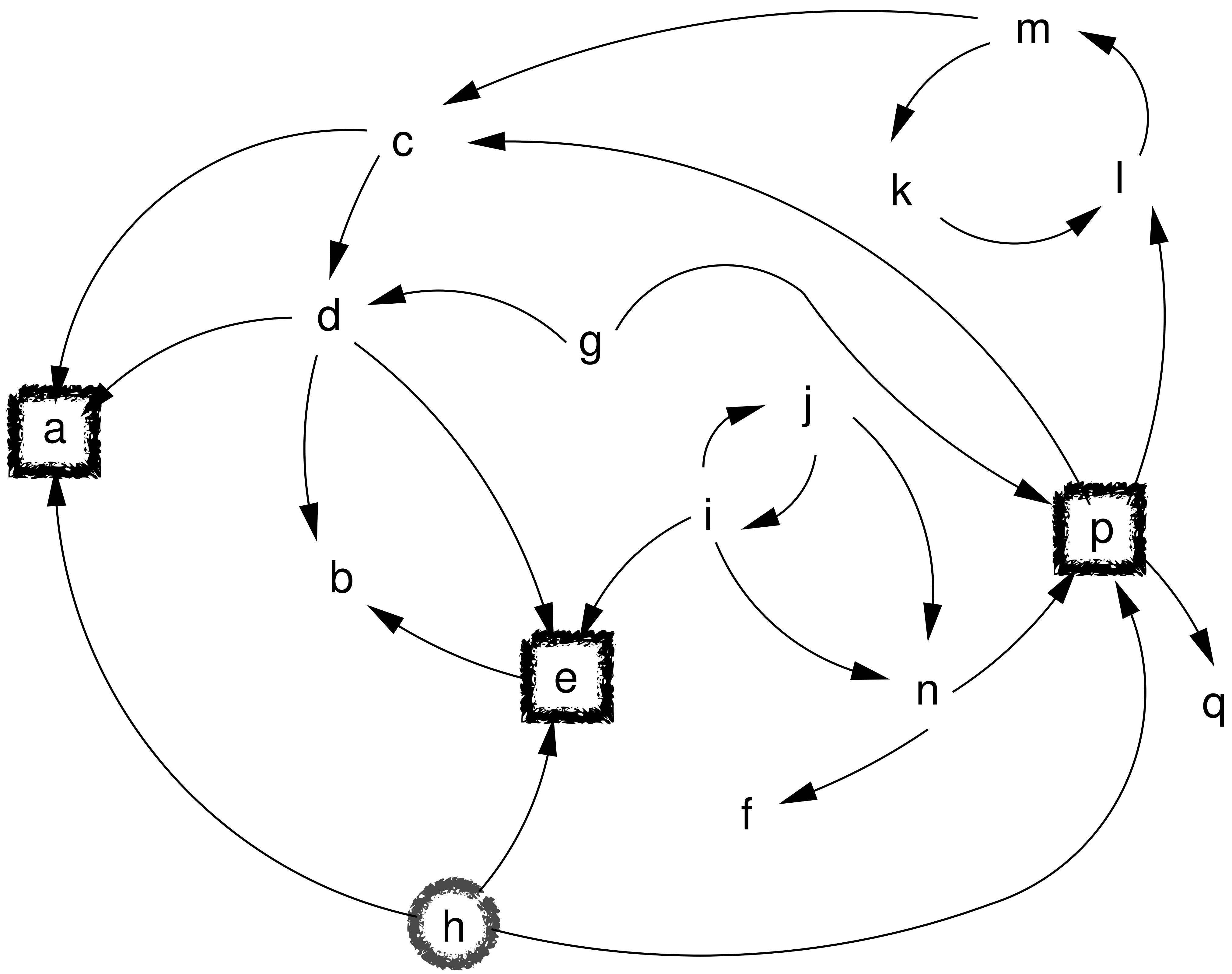
Grounded Extension

- A ***grounded extension*** is the least questionable set.
  - Accept only the arguments that one cannot avoid to accept
  - Reject only the arguments that one cannot avoid to reject
  - Abstain as much as possible.
- This gives rise to the most skeptical (or least committed) semantics
  - Arguments are ***guaranteed to be acceptable if they aren't attacked***.
    - There is no reason to doubt them - ***they are IN***
  - Arguments ***attacked by those that are in*** are therefore unacceptable
    - ***They are OUT*** — delete them from the graph.
  - Continue until the graph doesn't change.
- The grounded extension is the set of IN arguments
  - The grounded extension for our example is {*r*, *s*, *p*}

- ***p*** *: Since the weather today is sunny, I'm going to go out on my bike.*
- ***q*** *: Since today is a weekday and I have to go to work, I can't go out on my bike.*
- ***r*** *: Since today is a holiday, I don't have to go to work.*
- ***s*** *: Since I took the day off, I don't have to go to work.*

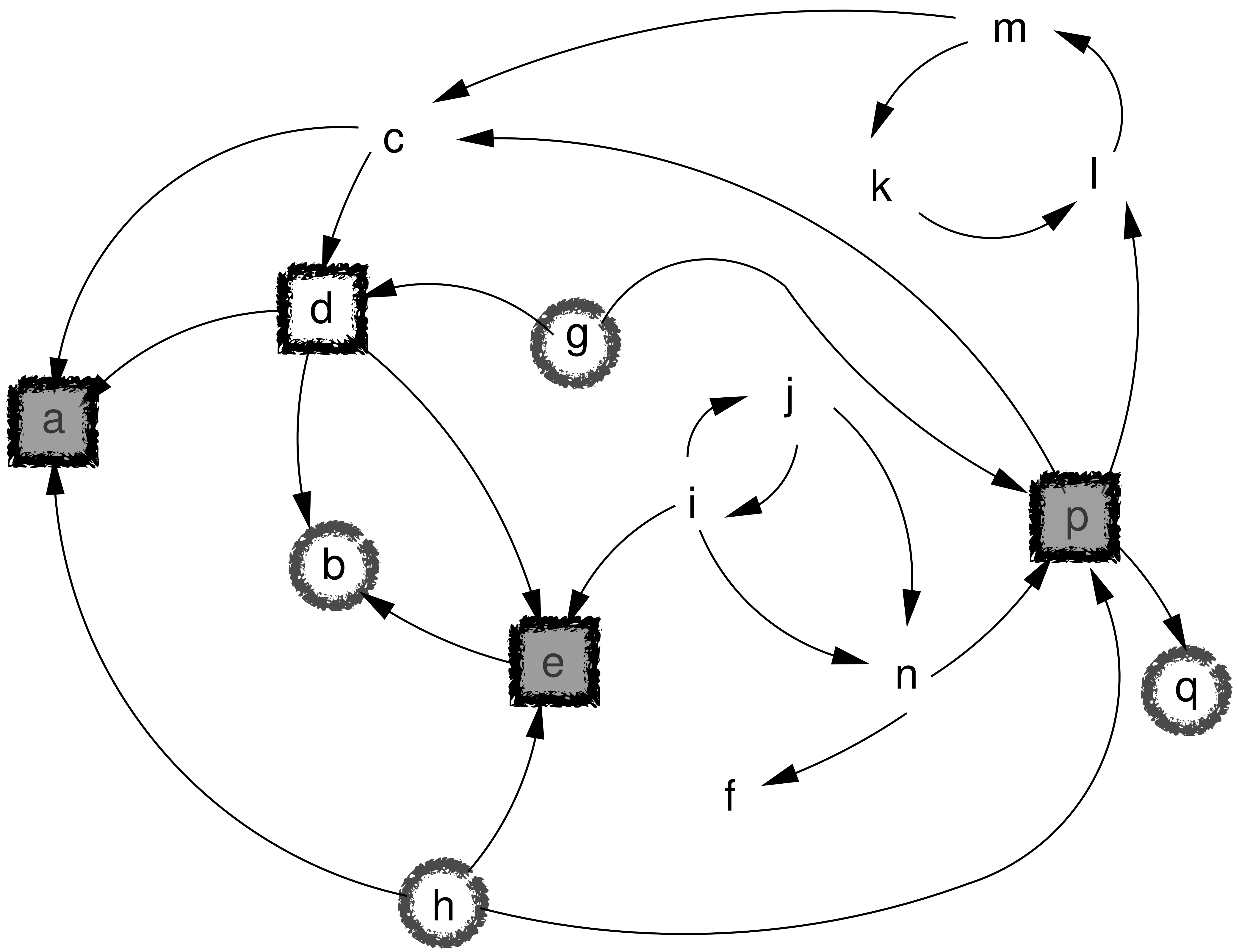


Grounded Extensions



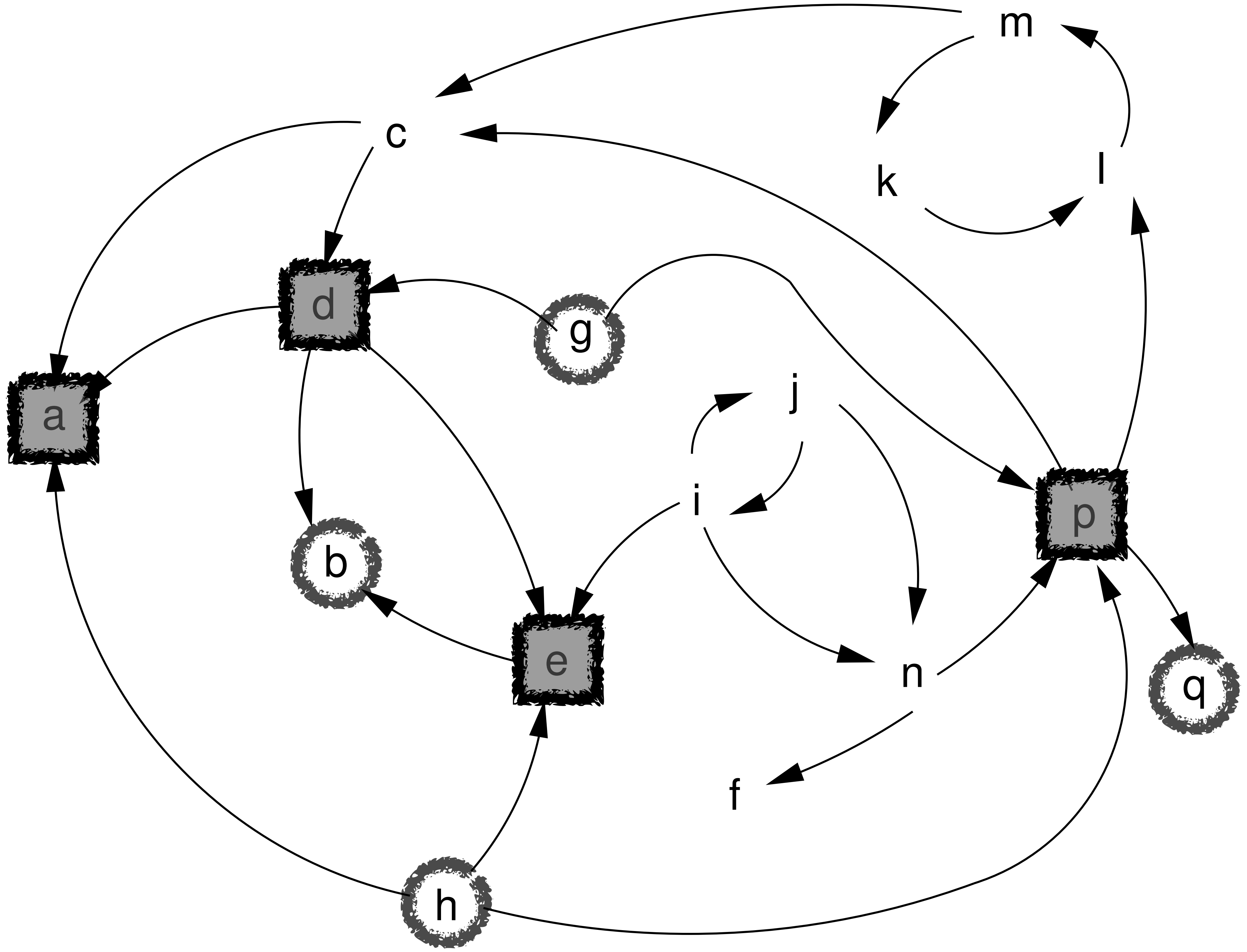
- Consider computing the grounded extension of the graph opposite.
  - We can say that:
    - h is not attacked, so ***h is IN***.
      - h is IN and attacks a, so ***a is OUT***.
      - h is IN and attacks p, so ***p is OUT***.
      - h is IN and attacks e, so ***e is OUT***.
    - …

Grounded Extensions



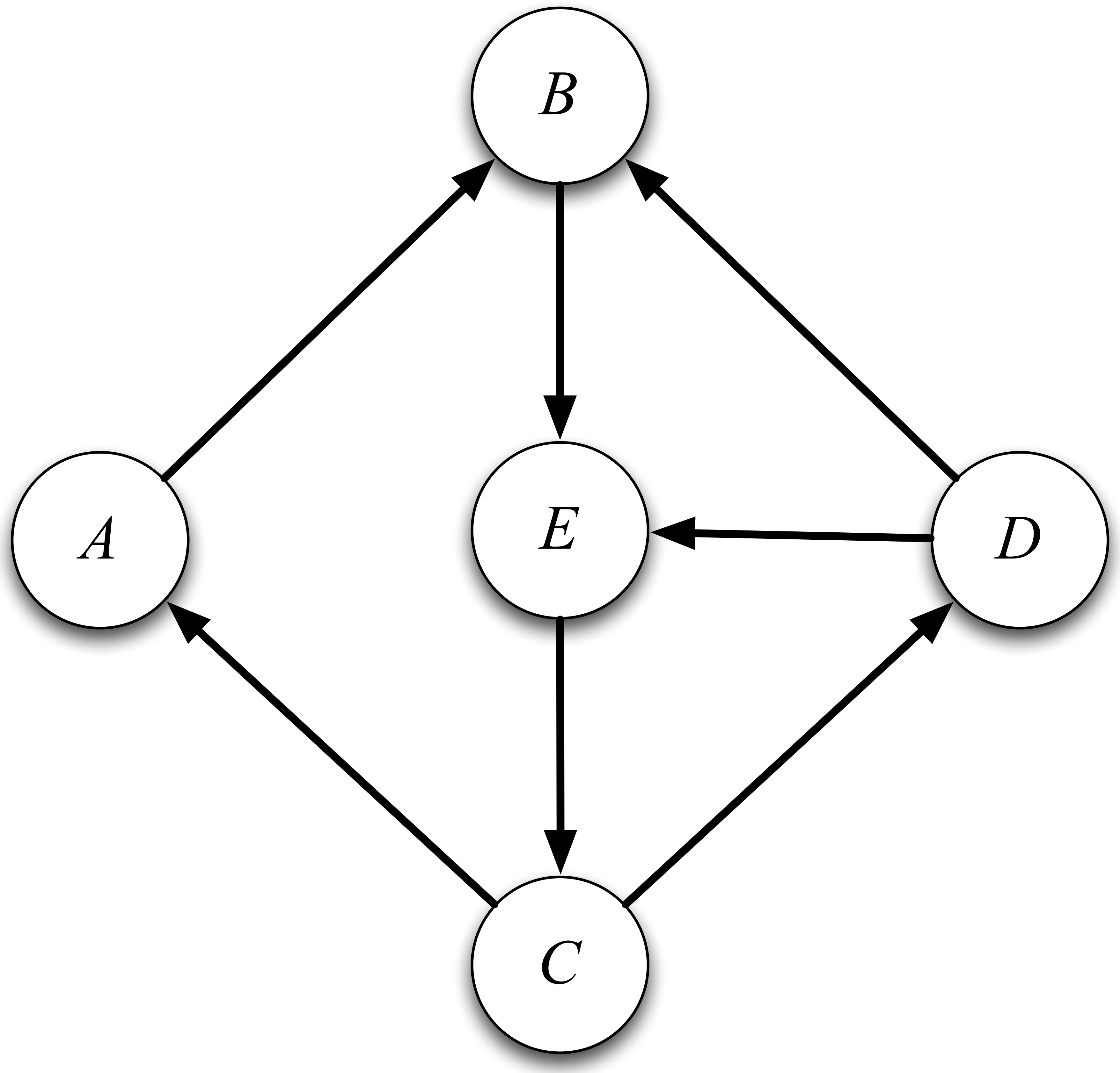
- Consider computing the grounded extension of the graph opposite.
  - We can say that:
    - h is not attacked, so ***h is IN***.
      - h is IN and attacks a, so ***a is OUT***.
      - h is IN and attacks p, so ***p is OUT***.
      - h is IN and attacks e, so ***e is OUT***.
    - p is OUT and is the only attacker of q so ***q is IN***.
    - g is not attacked, so ***g is IN***.
      - g is IN and attacks d, so ***d is OUT***.
      - g is IN and attacks p (which is also attacked by h) so ***p is OUT***.
    - B is no longer attacked, and so ***b is IN***.

Grounded Extensions



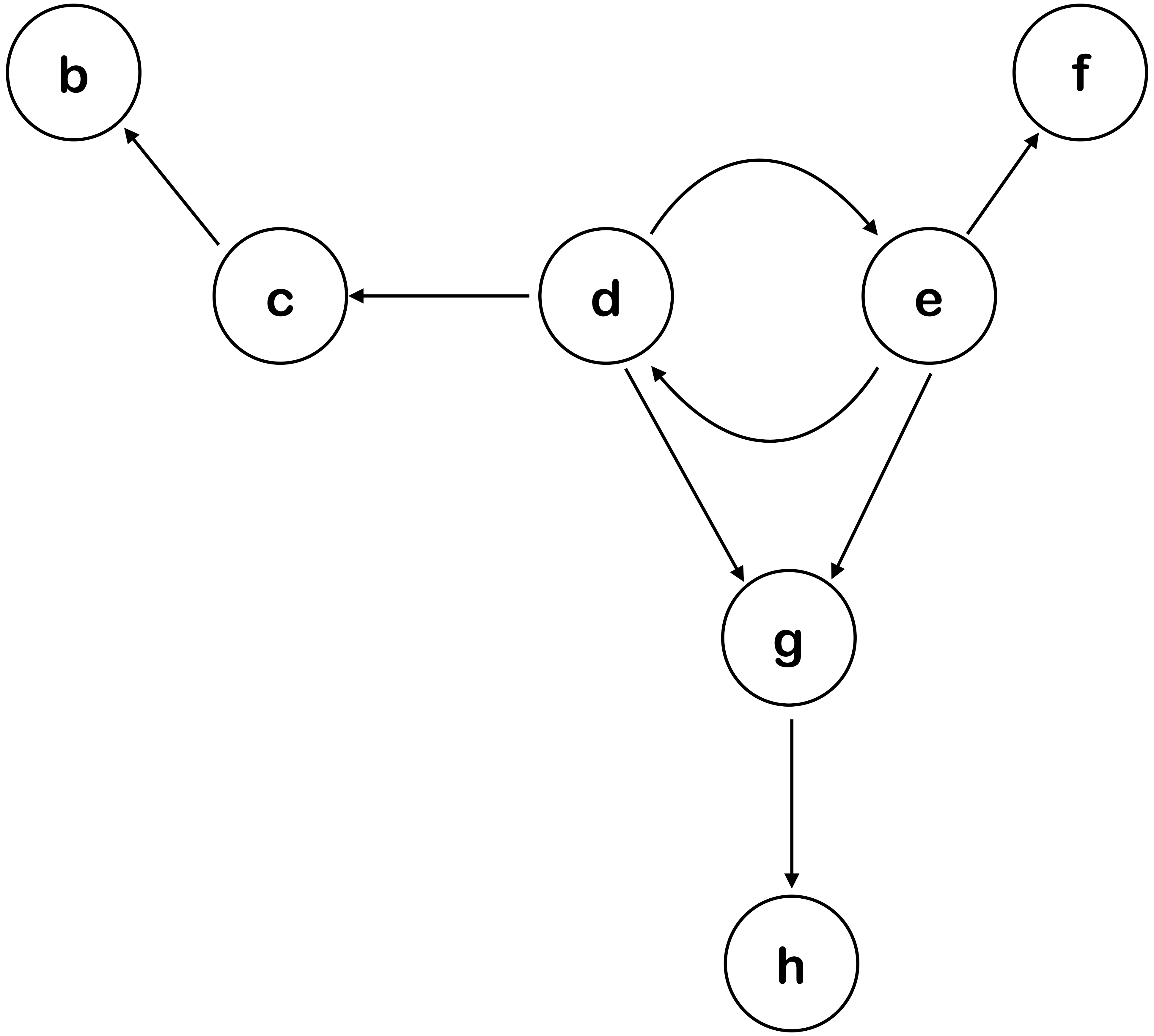
- Consider computing the grounded extension of the graph opposite.
  - We can say that:
    - h is not attacked, so ***h is IN***.
      - …
    - p is OUT and is the only attacker of q so ***q is IN***.
    - g is not attacked, so ***g is IN***.
      - …
    - B is no longer attacked, and so ***b is IN***
  - We can't say anything about:
    - m, k, l as they attack each other in a cycle
      - c as the status of m is not known
    - i, j as they mutually attack each other
    - n as the status of i or j is not definitively known
    - f as the status of n is unknown
  - The grounded extension is {b, g, h, q}

Full Example #1



- Conflict Free
  - ∅, {A, D}, {A, E}, {B, C}
    - These are the only positions that exist with no attack relations
- Mutually Defensive
  - ∅, {B, C}
    - {A,D} is not mutually defensive, because neither are defended from C
    - {A,E} is not mutually defensive, because A does not defend E from an attack by D
- Admissible:
  - ∅, {B,C}
    - These are the only positions that are both conflict free and mutually defensive
- Preferred Extensions:
  - {B,C}
- Credulously & Sceptically Accepted:
  - B, C
- Grounded Extension:
  - ∅
    - Every argument is attacked by at least one other argument, so it is not possible to determine any arguments that are IN (and consequently other arguments that are out)

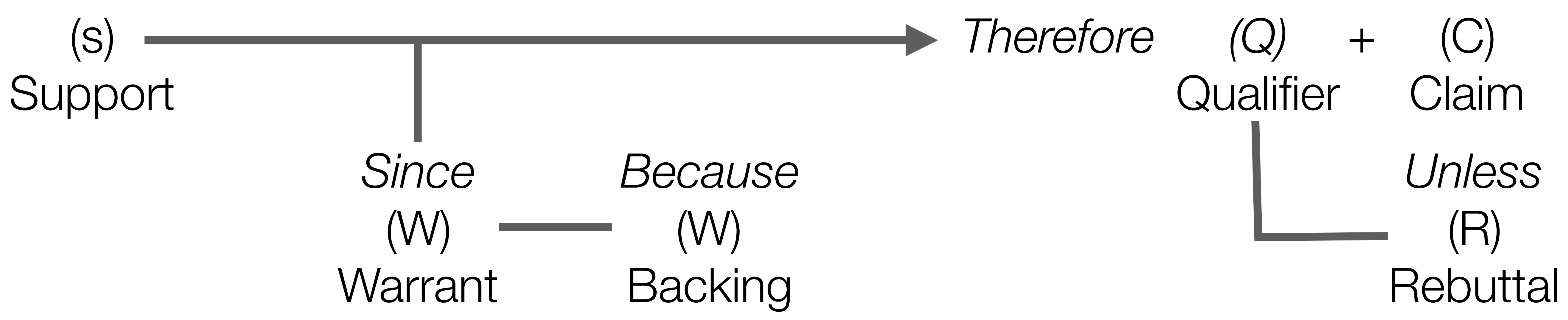
Full Example #2



- Admissible:
  - ∅, {b, d}, {c, e}, {e, h}, {d, f}, {d, h}, {b, d, f}, {b, d, h}, {c, e, h}, {d, f, h}, {b, d, f, h}
- Preferred Extensions:
  - {b, d, f, h}
  - {c, e, h}
- Credulously Accepted:
  - b, c, d, e, f
- Sceptically Accepted:
  - h
- Grounded Extension:
  - ∅

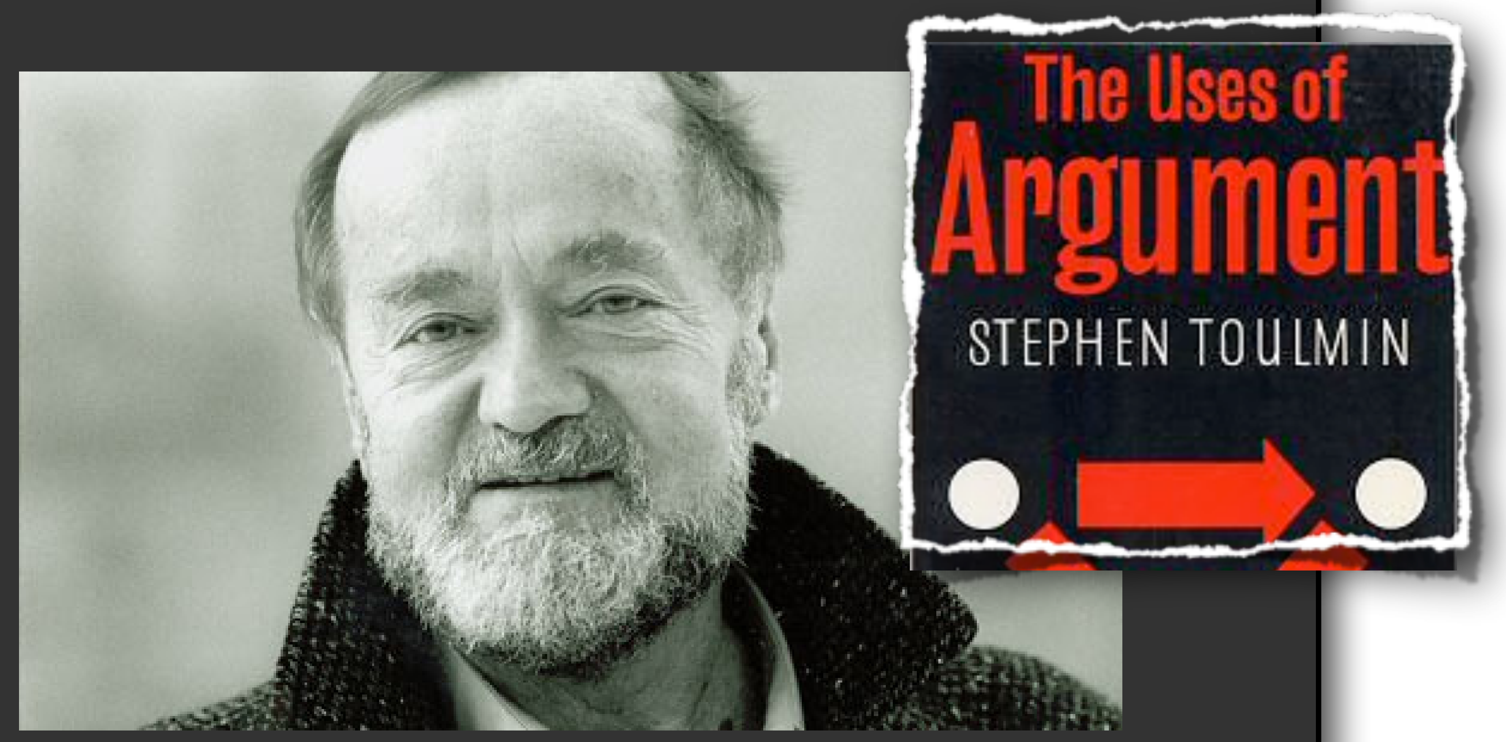
Deductive Argumentation

- Abstract argumentation models arguments as atomic, indivisible entities
  - However, arguments have a structure, which can be exploited when reasoning
- In deductive Argumentation, the arguments are modelled using logical formulae
  - Argumentation models ***defeasible reasoning***



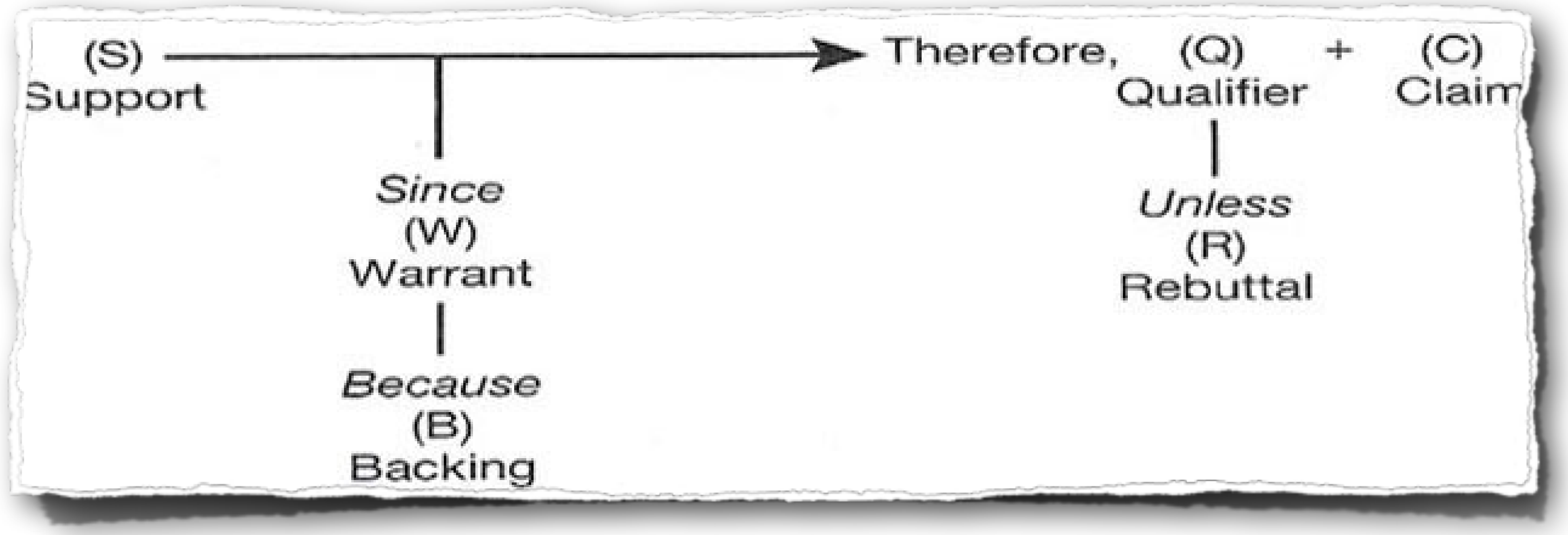
- Conclusions can be rebutted, premises (and warrants) can be challenged.

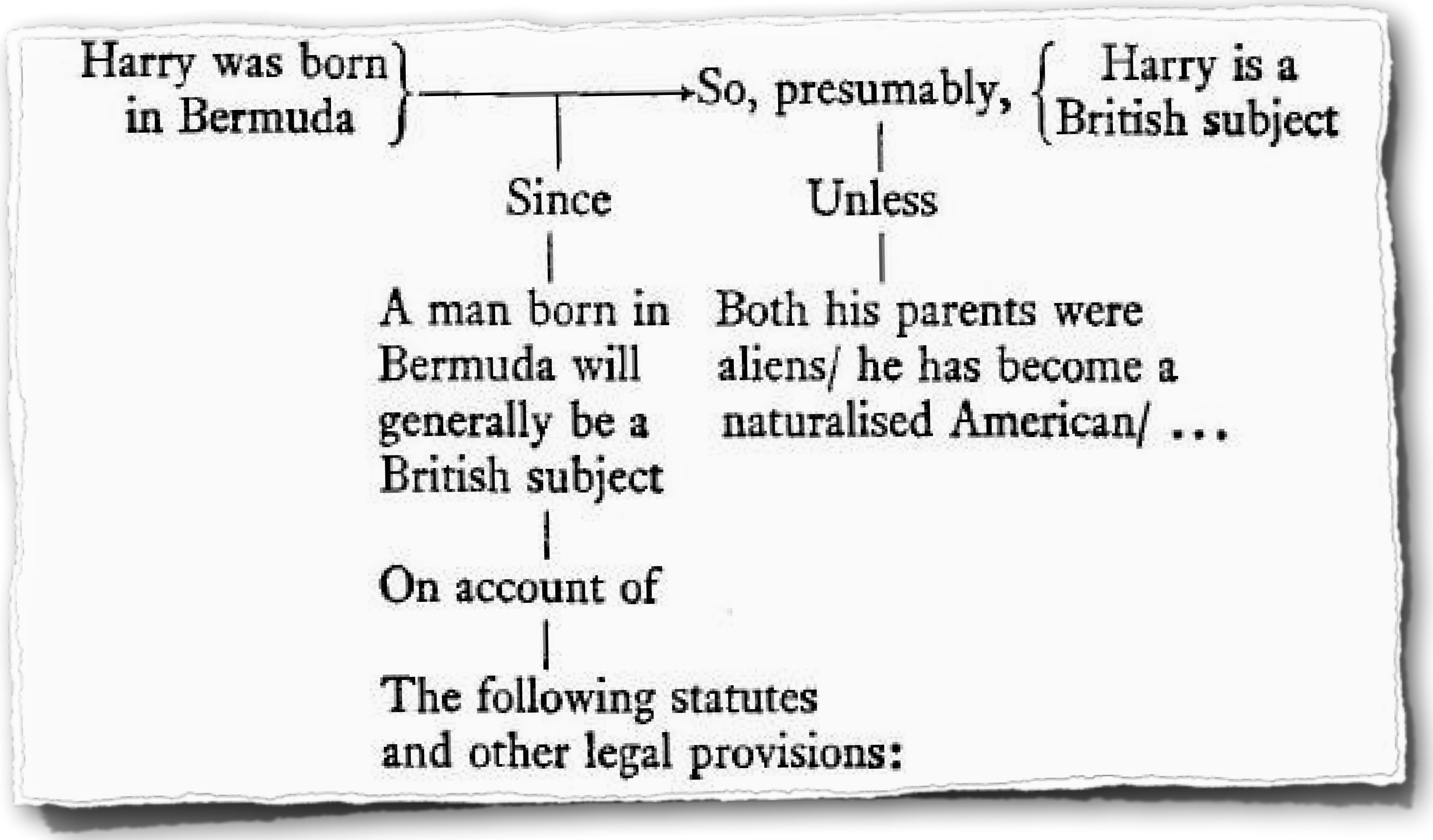
Stephen Toulmin



British philosopher who devoted his work to the analysis of moral reasoning. Throughout his writings, he sought to develop practical arguments which can be used effectively in evaluating the ethics behind moral issues. His works were later found useful in the field of rhetoric for analyzing rhetorical arguments.

Deductive Argumentation





- ***Claim (Conclusion)***
  - A conclusion whose merit must be established.
- ***Ground/Support (Fact, Evidence, Data)***
  - A fact one appeals to as a foundation for the claim.
- ***Warrant (Rule, Axiom)***
  - A statement authorising movement from the ground to the claim.
- ***Backing***
  - Credentials designed to certify the statement expressed in the warrant; backing must be introduced when the warrant itself is not convincing enough to the readers or the listeners.
- ***Rebuttal***
  - Statements recognising the restrictions which may legitimately be applied to the claim.
- ***Qualifier***
  - Words or phrases expressing the speaker's degree of force or certainty concerning the claim. Such words or phrases include "probably," "possible," "impossible," "certainly," etc

Deductive Argumentation

- The basic form of deductive arguments is $\Sigma \vdash (S, p)$
  - Σ is a (possibly inconsistent) set of logical formulae;
  - p is a sentence or proposition; i.e. a logical formula known as the ***conclusion***; and
  - S is the ***grounds*** or ***support***; i.e. a set of logical formulae such that:
    - $S \subseteq \Sigma$
    - $S \vdash p$ and
    - There is no $S' \subset S$ such that $S' \vdash p$
- Often we just write the argument as (S, p)

Example

Σ = {

human(*Socrates*)
human(*Heracles*)
father(*Heracles*, *Zeus*)
father(*Apollo*, *Zeus*)

divine(X) → ¬ *mortal*(X)
human(X) → *mortal*(X)
father(X, *Zeus*) → *divine*(X)
¬(*father*(X, *Zeus*) → *divine*(X)

}

Therefore, the following argument Arg_1 holds:

Arg_1= ({*human*(*Socrates*), *human*(X) → *mortal*(X)}, *mortal*(*Socrates*))

I.e.

S = {*human*(*Socrates*), *human*(X) → *mortal*(X)}
p = *mortal*(*Socrates*)

Deductive Argumentation

- Argumentation takes into account the relationship between arguments.
  - Let (S_1, p_1) and (S_2, p_2) be arguments from some database Σ
  - Then (S_1, p_1) can be attacked in one of two ways:
- Rebut
  - (S_2, p_2) rebuts (S_1, p_1) if $p_2 \equiv \neg p_1$.
  - i.e. the conclusions attack or contradict each other
- Undercut
  - (S_2, p_2) undercuts (S_1, p_1) if $p_2 \equiv \neg q_1$ for some $q_1 \in S_1$.
  - i.e. the conclusion p_2 attacks some formulae q_1 in the support for p_1

Example

Σ = {
human(*Heracles*)
father(*Heracles*, *Zeus*)
father(*Apollo*, *Zeus*)

divine(X) → ¬ *mortal*(X)
human(X) → *mortal*(X)
father(X, *Zeus*) → *divine*(X)
¬(*father*(X, *Zeus*) → *divine*(X)
}

Given the argument Arg_2:
Arg_2= ({*human*(*Heracles*), *human*(X) → *mortal*(X)}, *mortal*(*Heracles*))

The argument Arg_3 rebuts Arg_2:
Arg_3= ({*father*(*Heracles*, *Zeus*), *father*(X, *Zeus*) → *divine*(X), *divine*(X) → ¬ *mortal*(X)}, ¬ *mortal*(*Heracles*))

The argument Arg_4 undercuts Arg_3:
Arg_4= ({¬(*father*(X, *Zeus*) → *divine*(X))}, ¬(*father*(X, *Zeus*) → *divine*(X)))

Attack and Defeat

- Deductive argumentation connects to the abstract ideas we were just looking at.
  - A ***rebuttal*** or ***undercut*** between two arguments becomes ***the attack*** in a Dungian system.
- Note that a rebut is symmetrical
  - Causes problems with some kinds of extension.
- Once we have identified attacks, we can look at preferred extensions or grounded extensions to determine what arguments to accept.

Another Example

Argument x

Here is one deductive argument.

a denotes "*We recycle*"
b denotes "*We save resources*"
$a \rightarrow b$ denotes "*If we recycle, then we save resources*"

Formally we get: $(\{a, a \rightarrow b\}, b)$

Argument y

A second argument, that conflicts with the first:

c denotes "*Recycled products are not used*"
$a \wedge c \rightarrow \neg b$ denotes "*If we recycle and recycled products are not used then we don't save resources*"

Formally we get: $(\{a, c, a \wedge c \rightarrow \neg b\}, \neg b)$

Argument z

A third argument, that conflicts with the first:

d denotes "*We create more desirable recycled products*"
$d \rightarrow \neg c$ denotes "*If we create more desirable recycled products then recycled products are used*"

Formally we get: $(\{d, d \rightarrow \neg c\}, \neg c)$

x and y rebut each other.

z undercuts y

Different Dialogues

- With appropriate choice of language, can use argumentation to capture all of these kinds of dialogue.
  - ***Information seeking*** (Personal Ignorance)
    - Tell me if p is true.
  - ***Inquiry*** (General Ignorance)
    - Can we prove p?
  - ***Persuasion*** (Conflict of opinions)
    - You're wrong to think p is true.
  - ***Negotiation*** (Conflict of interest)
    - How do we divide the pie?
  - ***Deliberation*** (Need for Action)
    - Where shall we go for dinner?

Persuasion Dialogues

- We have two agents, P and C, each with ***some knowledge base***, Σ_P and Σ_C.
  - Each time one makes an assertion, it is considered to be an addition to its ***commitment store***, $CS(P)$ or $CS(C)$.
    - Thus: P can build arguments from $\Sigma_P \cup CS(C)$ …
    - … and C can use $\Sigma_C \cup \mathrm{CS}(P)$.
  - Commitment stores are information that the agent has made public.

- We assume that dialogues start with P making the first move.
  - The outcomes, then, are:
    - P generates an argument both classify as IN, or
    - C makes P argument OUT.

- Can use this for negotiation if the language allows you to express offers.

Persuasion Dialogues

- A typical persuasion dialogue would proceed as follows:
  - P has an acceptable argument (S, p), built from Σ_P, and wants C to accept p.
  - P asserts p.
  - C has an argument $(S', \neg p)$.
  - C asserts $\neg p$.
  - P cannot accept $\neg p$ and challenges it.
  - C responds by asserting S'.
  - P has an argument $(S'', \neg q)$ where $q \in S'$, and challenges q.
  - ...

Persuasion Dialogues

- This process eventually terminates when

$$\Sigma_P \cup CS(P) \cup CS(C)$$

and

$$\Sigma_C \cup CS(C) \cup CS(P)$$

  - eventually provide the same set of IN arguments and the agents agree.

- Clearly here we are looking at grounded extensions.

Summary

- This chapter has looked at argumentation as a means through which agents can reach agreement.
  - Argumentation allows for more complex interactions than the negotiation mechanisms we looked at last chapter.

- Argumentation can be used for a range of tasks that include negotiation.
  - Also allows for inquiry, persuasion, deliberation.

Class Reading (Chapter 16):

"An introduction to argumentation semantics", Pietro Baroni, Martin Caminada and Massimiliano Giacomin, The Knowledge Engineering Review, Volume 26 Issue 4, December 2011, pp 365-410

This paper reviews Dung's original notions of complete, grounded, preferred, and stable semantics, as well as subsequently proposed notions like semi-stable, ideal, stage, and CF2 semantics, considering both the extension-based and the labelling-based approaches with respect to their definitions.