Interacting Answer Sets

Chiaki Sakama
Wakayama University

Tran Cao Son
New Mexico State University

CLIMA-X, Hamburg, Sep. 2009

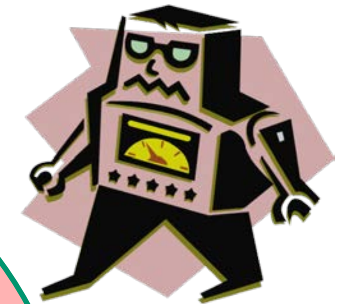
Background

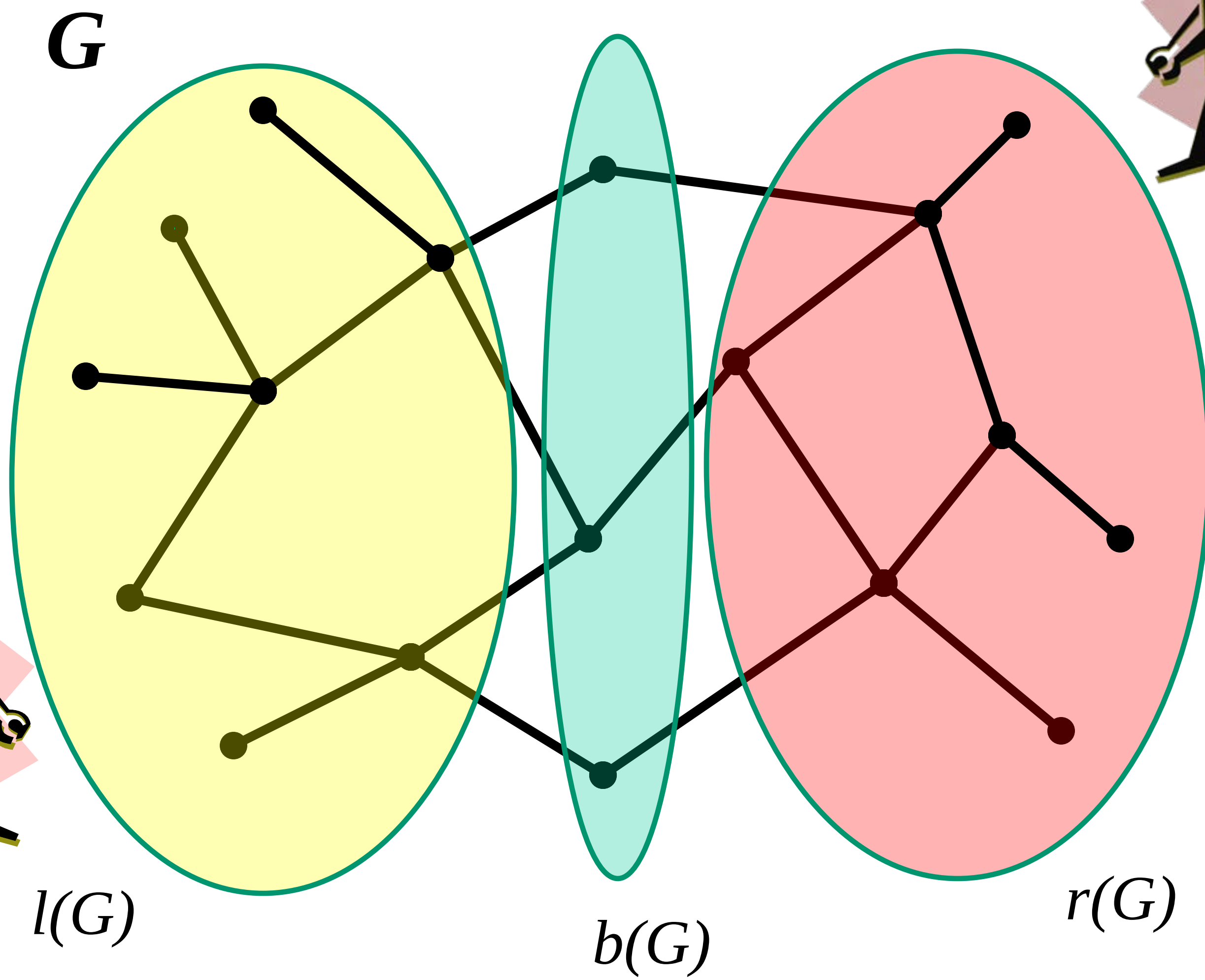
- In a multiagent society, agents interact with one another to pursue their goals or perform their tasks.
- The behavior of one agent is affected by other agents or constrained in a society she belongs to.
- Agents interact differently depending on situations: they work cooperatively to achieve a common goal, while behave competitively when goals are conflicting.

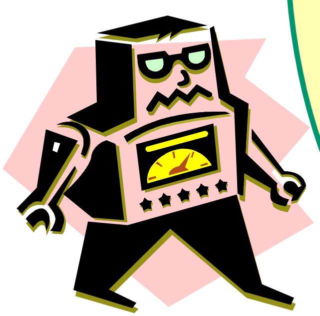
Example

- There is a graph G and two robots P1 and P2, trying to cooperatively solve the graph-coloring problem on G.
- They make a plan: P1 paints the left-half l(G) and P2 paints the right-half r(G).
- There are some nodes on the border b(G) and these nodes can be painted by each robot independently.

Example







Example

Controls over the behaviors of robots are requested.

- Every node in the graph G must be painted by either P1 or P2. (Norms)
- Every node on the border must have a unique color. (Cooperation)
- Every node in the left-half of G is painted by P1 but not by P2. A similar condition is imposed on nodes in the right-half of G. (Competition)
- If P1 is prior to P2 in deciding colors of nodes on the border, P2 must accept the decision of P1. (Subjection)

Contribution

- We formulate various types of social interactions such as cooperation, competition, norms, subjection.
- Those interactions are captured as the interactions among answer sets of logic programs.
- We provide a method for computing coordinated solutions using answer set programming.

Problem Setting

- An agent has a knowledge base represented by a logic program that consists of rules of the form:

  $L_1 ; \ldots ; L_l \leftarrow L_{l+1}, \ldots, L_m, \textit{\textbf{not}}\ L_{m+1}, \ldots, \textit{\textbf{not}}\ L_n$

  where L_i is a literal and ***not*** is negation as failure.
- The declarative semantics of a program is given by the answer set semantics. The set of all answer sets of a program P is written as AS(P).
- A society is a finite set of agents, and individual agents have their own respective programs over a common language and a shared ontology in a society.

Social Interactions

- Cooperation: an interaction among agents to work together to achieve a common goal.
- Competition: an interaction such that a satisfactory result for one agent implies unsatisfactory results for others.
- Norms: an interaction that directs an agent to meet expectation or obligations in a society.
- Subjection: an interaction that restricts behavior of one agent relative to another agent.

Cooperation

- Let P1 and P2 be two programs and $\Phi \subseteq Lit$, where Lit is the set of all ground literals. Two answer sets $S \in AS(P1)$ and $T \in AS(P2)$ cooperate on Φ if $S \cap \Phi = T \cap \Phi$.
- The above condition requires that two answer sets S and T must include the same elements from Φ.
- This type of interaction is useful to specify agreement or a common goal in a society.

Example

- John and Mary are planning to go to a restaurant. John prefers French and Mary prefers Italian, but they behave together anyway. John (P1) and Mary (P2) have programs such that

  P1: preferred ← french, french; italian ←,

  P2: preferred←italian, french; italian ←.

- P1 has two answer sets S1={french, preferred} and S2={italian}, while P2 has T1={italian, preferred} and T2={french}. Putting Φ={french, italian}, S1 and T2 cooperate on Φ, and S2 and T1 cooperate on Φ.

Accept, Adapt

<u>Prop.</u> If S and T cooperate on Φ, they cooperate on any Φ' such that $\Phi' \subseteq \Phi$. (monotonicity)

<u>Def.</u> $S \in AS(P1)$ accepts $T \in AS(P2)$ if $S \supseteq T$.
If S accepts T, T adapts to S.

<u>Prop.</u> $S \in AS(P1)$ accepts $T \in AS(P2)$ iff S and T cooperate on T. S adapts to T iff S and T cooperate on S.

Concession

When S cannot accept nor adapt to T, two agents might make a concession.

Def. For any pair of answer sets $S \in AS(P1)$ and $T \in AS(P2)$, $\Phi = S \cap T$ is called a concession between P1 and P2.

Prop. If a set Φ is a concession between P1 and P2, then there are $S \in AS(P1)$ and $T \in AS(P2)$ which cooperate on Φ.

Competition

- Let P1 and P2 be two programs and $\Psi \subseteq Lit$. Two answer sets $S \in AS(P1)$ and $T \in AS(P2)$ are competitive for Ψ if $S \cap T \cap \Psi = \{\}$.
- The above condition requires that two answer sets S and T do not share any element belonging to Ψ.
- This type of interaction is useful to specify a limited resource or an exclusive right in a society.

Example

- John and Mary share a car. John plans to go fishing if he can use the car, while Mary wants to go shopping if the car is available. John (P1) and Mary (P2) have programs such that

  P1: fishing ← use_car, use_car ; ¬use_car ←,

  P2: shopping ← use_car, use_car ; ¬use_car ←.

- P1 has two answer sets S1={fishing, use_car} and S2={¬use_car}, while P2 has T1={shopping, use_car} and T2={¬use_car}. Putting Ψ={use_car}, S1 and T2 are competitive for Ψ, and S2 and T1 are competitive for Ψ.

Benefit, Precedence

Prop. If S and T are competitive for Ψ, they are competitive for any Ψ' such that $\Psi' \subseteq \Psi$. (monotonicity)

Def. Suppose that $S \in AS(P1)$ and $T \in AS(P2)$ are competitive for Ψ. Then,

- S has benefit over T wrt Ψ if $S \cap \Psi \neq \{\}$.
- S has precedence over T wrt Ψ if $S \cap \Psi \supseteq T \cap \Psi$.

Prop. If S has precedence over T wrt Ψ, T cannot have benefit over S wrt Ψ.

Example

- There are two companies P1 and P2. P1 has a right to mine both oil and gas, while P2 has a right to mine either one of them. The situation is represented by answer sets of programs:
  AS(P1)={{oil, gas}} and AS(P2)={{oil},{gas}}.
- Then, {oil, gas} and {gas} are competitive for Ψ={oil}, while {oil, gas} and {oil} are not. In this case, {oil, gas} has precedence over {gas} wrt {oil}.
- This means that if two companies coordinate their answer sets to be cometitive for Ψ, there is no chance for P2 to mine oil.

Norms

- Let P1 and P2 be two programs and $\Theta \subseteq Lit$. Two answer sets $S \in AS(P1)$ and $T \in AS(P2)$ achieve norms for Θ if $(S \cup T) \cap \Theta = \Theta$.
- The above condition requires that two answer sets S and T jointly include every element in Θ.
- This type of interaction is useful to specify duty or task allocation in a society.

Example

- Mary plans a home party. She asks her friends, John and Susie, to buy wine, juice and water. John will visit a liquor shop and can buy wine or water or both. Susie will visit a grocery store and can buy juice or water or both. John (P1) and Susie (P2) have programs s.t.

  P1: $wine \,;\, \neg wine \leftarrow, \quad water \,;\, \neg water \leftarrow,$

  P2: $juice \,;\, \neg juice \leftarrow, \quad water \,;\, \neg water \leftarrow.$

- Each program has 4 answer sets representing buying items. Of which, the following 3 pairs achieve norms for Θ={wine,juice,water}: {wine, water} and {juice,water}, {wine, ¬water} and {juice,water}, and {wine, water} and {juice,¬water}.

Responsibility

Prop. If S and T achieve norms for Θ, they achieve any norms for any Θ' such that $\Theta' \subseteq \Theta$. (monotonicity)

Def. Let $S \in AS(P1)$, $T \in AS(P2)$ and $\Theta \subseteq Lit$. We say

- S is individually responsible for $\Theta \setminus T$;
- S has no responsibility if S is individually responsible for {}.
- S is less responsible than T if $\Theta \setminus T \subseteq \Theta \setminus S$.

Prop.

1. S and T achieve norms for Θ if either S or T contains individual responsible set.
2. If $S \subseteq T$ then S is less responsible than T.
3. If $T \supseteq \Theta$ then S has no responsibility.

Example

- An individual responsible set $\Theta \setminus T$ represent the least task or obligation for S to achieve norms. Undertaking individual responsibilities does not always achieve norms.
- S={wine, water} and T={juice, water} achieve norms for Θ={wine, juice, water}. Thus, S is responsible for $\Theta \setminus T$={wine} and T is responsible for $\Theta \setminus S$={juice}.
- If John only buys wine and Susie only buy juice, however, they might not achieve norms for Θ.
- To achieve norms, John or Susie has to voluntarily buy water.

Volunteer

Def. Let $S \in AS(P1)$, $T \in AS(P2)$ and $\Theta \subseteq Lit$. We say

- S and T volunteer for $S \cap T \cap \Theta$;
- For $S' \in AS(P1)$ and $T' \in AS(P2)$, (S,T) requires less voluntary actions than (S',T') if $(S \cap T \cap \Theta) \subseteq (S' \cap T' \cap \Theta)$.

By the definition, a voluntary action is required only if $S \cap T \neq \{\}$.

Prop. Let $\Theta \subseteq Lit$, $\{S,S'\} \subseteq AS(P1)$, and $\{T,T'\} \subseteq AS(P2)$ s.t. S and T (resp. S' and T') achieve norms for Θ. Then, (S,T) requires less voluntary actions than (S',T') iff S and T have more individual responsibility than S' and T'.

Commitment

An agent is expected to take a voluntary action in addition to his/her individual responsibility. To declare his/her action to another agent, an agent creates commitment.

Def. A commitment $C(P1,P2,Q)$ represents a pledge of an agent P1 to another agent to realize Q.

Prop. $S \in AS(P1)$ and $T \in AS(P2)$ achieve norms for Θ only if commitments $C(P1,P2,U)$ and $C(P2,P1,V)$ are made such that $U \subseteq S$, $V \subseteq T$, and $\Theta \subseteq U \cup V$.

Ex. In order for $S=\{wine,water\}$ and $T=\{juice,water\}$ to achieve norms for $\Theta=\{wine,juice,water\}$, it is requested to make commitments $C(P1,P2,\{wine\})$ and $C(P2,P1,\{juice,water\})$, for instance.

Subjection

- Let P1 and P2 be two programs and $\Lambda \subseteq Lit$. An answer set $S \in AS(P1)$ is subject to an answer set $T \in AS(P2)$ wrt Λ if $T \cap \Lambda \subseteq S \cap \Lambda$.
- The above condition represents that any element from Λ which is included in T must be included in S.
- This type of interaction is useful to specify priority or power relations in a society.

Example

- Bob and John are two kids in a family, and they have limited access to the Internet. Since Bob is older than John, any site which is limited to access by Bob is also limited to John, but not vice versa. For site1 and site2, John (P1) and Bob (P2) have programs s.t.

P1: acc_site1; ¬acc_site1 ← usr_John,
acc_site2; ¬acc_site2 ← usr_John,
usr_John ←.

P2: acc_site1; ¬acc_site1 ← usr_Bob,
acc_site2; ¬acc_site2 ← usr_Bob,
usr_Bob ←.

Example (cont.)

- Remind: $S \in AS(P1)$ is subject to $T \in AS(P2)$ wrt Λ if $T \cap \Lambda \subseteq S \cap \Lambda$. And Bob (P2) is older than John (P1).
- Each program has 4 answer sets representing accessible sites. Putting $\Lambda = \{\neg acc\_site1\}$, 12 pairs of answer sets, out of 16 combinations of those of P1 and P2, are in subjection relation wrt Λ.
- For instance, the following pairs are two solutions:
  $S1 = \{\neg acc\_site1, \neg acc\_site2, usr\_John\}$ is subject to $T1 = \{acc\_site1, acc\_site2, usr\_Bob\}$ wrt Λ; and
  $S2 = \{\neg acc\_site1, acc\_site2, usr\_John\}$ is subject to $T2 = \{\neg acc\_site1, \neg acc\_site2, usr\_Bob\}$ wrt Λ.

Properties

Prop. If S is subject to T wrt Λ, the subjection relation holds for any Λ' such that $\Lambda' \subseteq \Lambda$. (monotonicity)

Prop. If $S \supseteq T$, S is subject to T wrt any Λ.

If any information in $T \in AS(P2)$ should be included in $S \in AS(P1)$, it is achieved by putting $\Lambda = T$.

Prop. If S is subject to T wrt T, $S \supseteq T$.

Prop. For any Λ,

1. S and T cooperate on Λ iff S is subject to T wrt Λ and T is subject to S wrt Λ.
2. If S and T are competitive for Λ and S is subject to T wrt Λ, then S has precedence over T wrt Λ.

Thus, precedence is a special case of a subjection relation.

Coordination

Answer set interactions are combined into a single framework.

Def. For two programs P1 and P2, a tuple of sets of literals $\Omega=(\Phi,\Psi,\Theta,\Lambda)$ is called a coordination over P1 and P2. Each component X of Ω is denoted as Ω_x. $S \in AS(P1)$ and $T \in AS(P2)$ satisfy Ω_x if they satisfy the condition of X of the corresponding interaction.

Def. Let P1 and P2 be two programs and Ω a coordination over P1 and P2. Two answer sets $S \in AS(P1)$ and $T \in AS(P2)$ are compatible wrt Ω if S and T satisfy Ω_x for every $X \in \{\Phi,\Psi,\Theta,\Lambda\}$.

Extensions

- When no pair of answer sets $S \in AS(P1)$ and $T \in AS(P2)$ is compatible wrt Ω, priority relations are introduced over Ω_X for $X \in \{\Phi, \Psi, \Theta, \Lambda\}$. Then, the notion of compatibility under priority is introduced.
- When Φ, Ψ, Θ, or Λ is given as a set of rules, we can specify interactions that may change depending on different contexts.
- Interactions between 2 answer sets are generalized to those among more than two agents.
- The notion of interactions between answer sets is applied to interactions between programs.