Introduction to the Workshop on Logics for Resource-Bounded Agents

Natasha Alechina and Brian Logan

ESSLLI 2015, Barcelona

Topics of the workshop

- logics for modelling resource-bounded reasoners
  - epistemic logics for modelling resource-bounded reasoners
  - logics for modelling bounded memory, forgetting etc.
- logics for reasoning about resources

Timetable and brief introduction to the talks

Tuesday: Nils Bulling, *Verifying Resource-Bounded Agents*
Stéphane Demri, *Reversal-Bounded Counter Machines*

Wednesday: Fernando Velázquez-Quesada, *Forgetting Propositional Formulas*
Sophia Knight, *A Strategic Epistemic Logic for Bounded Memory Agents*

Thursday: Lasha Abzianidze, *A Logic of Belief with the Complexity Measure*
Igor Sedlár, *Substructural Epistemic Logics*

Friday: Dario Della Monica, *Model Checking Coalitional Games in Shortage Resource Scenarios*
Valentin Goranko, *Resource Bounded Reasoning in Concurrent Multi-Agent Systems*

Outline

- logics for modelling resource bounded reasoners
  - logical omniscience
  - Step Logic
  - Algorithmic Knowledge
  - Justification Logic
  - Dynamic Syntactic Epistemic Logic
- logics for reasoning about resources
  - RB-ATL
  - RB$\pm$ATL
- open problems

Logics for modelling resource-bounded reasoners

- this will be familiar to people who attended Fernando's course last week
- often, in this approach knowledge and beliefs are modelled syntactically rather than using possible worlds semantics
- we will give a brief survey of this area
- the talks by Fernando Velázquez-Quesada, Sophia Knight, Lasha Abzianidze, and Igor Sedlár belong to this area

Logics for reasoning about resources

- another area of the workshop is reasoning about actions that cost resources
- at least from our point of view, the two areas are very connected
- we started investigating syntactic epistemic logics where actions of deriving a formula and communicating had explicit costs, and storing formulas cost memory
- we then generalised it to Coalition Logic (CL) and Alternating Time Temporal Logic (ATL) where action have costs (RB-CL, RB-ATL, RB$\pm$ATL)

Logics for reasoning about resources

- resource quantities are numerical, and in addition to states we get vectors of numbers (resource amounts) updated by transitions
- this is why model-checking of such systems is related to decision problems for counter machines and vector addition systems with state
- the talks by Nils Bulling, Stéphane Demri, Hoang Nga Nguyen, Dario Della Monica and Valentin Goranko belong to this area

Epistemic logic: logical omniscience

- epistemic logic studies belief and knowledge modalities
- it usually interprets ‘agent knows (believes) that ϕ’ as ‘ϕ is true in all knowledge (belief)-accessible possible worlds’
- clearly, tautologies are all true in all accessible worlds, so the agent believes all tautologies
- also, if the agent believes ϕ, and ψ is a logical consequence of ϕ, then ψ is true in all ϕ-worlds, so the agent believes ψ as well
- so the agent believes all logical theorems and can derive infinitely many consequences infinitely fast (logical omniscience problem)

Logical omniscience: is this a problem?

- Hintikka 1975: philosophical problem (human reasoners)
- however, idealised reasoners *can be* considered logically omniscient (capable of arbitrary correct inferences)
- after all, not many people complain that epistemic logic does not account for logical mistakes

When logical omniscience is a problem

- logical modelling and verfication of AI agents
- if we ascribe beliefs to the agent incorrectly (for example assume that it believes arbitrary logical consequences of its beliefs when it does not) then we may model its behaviour incorrectly
- so if we ascribe to the agent an ability to reason in logic, then:
  - either the agent should really be able to reason (and exactly to the extent that the logic predicts)
  - or, its internal belief language and belief tests in its action selection should be so trivial that it does not matter

Solutions to the logical omniscience problem

- impossible worlds (beliefs still closed under logical consequence but in a weaker logic)
- neighbourhood semantics (beliefs are closed under logical equivalence: if the agent believes one tautology, it believes them all)
- explicit knowledge defined using awareness (syntactic notion - ‘awareness set’ is an arbitrary set of formulas)
- algorithmic knowledge, syntactic knowledge/beliefs (beliefs are tokens to be manipulated rather than propositions corresponding to sets of possible worlds)

Step logic

- Elgot-Drapkin & Perlis 1990
- the idea is to represent stages in agent's reasoning (corresponding to time points):

$$\frac{i : \ A, \ A \rightarrow B}{i+1 : \ B}$$

- if at time i the agent knows A and $A \rightarrow B$, then at time $i+1$ the agent will know B

Algorithmic Knowledge

- Halpern, Moses, and Vardi 1994: agents' explicit knowledge is given by an algorithm they use to answer queries
- Pucella 2006: deductive algorithmic knowledge
- explicit knowledge of agents comes from a logical theory expressed by a deductive system consisting of deduction rules
- agents' explicit knowledge is closed with respect to this set of rules (similar to Konolige 1986)

Logical omniscience as a complexity problem

- Artemov, Kuznets, Krupski since 2006, inspired by Justification Logic
- a proposition can be feasibly knowable if it is provable in polynomial time
- to be more precise:
  - a system weakly avoids logical omniscience, if for every provable $K\ A$, A has a polynomial size proof
  - a system strongly avoids logical omniscience, if there is a polynomial algorithm such that which for every provable $K\ A$, produces a proof of A

Consider an agent reasoning in S4

- $K\ (A \to B) \to (K\ A \to K\ B)$
- $K\ A \to A$
- $K\ A \to K\ K\ B$
- Necessitation: if A is an axiom, $\vdash_{S4} K\ A$

Feasible knowledge in $S4_\bullet$ (Artemov et al)

- $[k_1](A \to B) \to ([k_2]A \to [k_1 \cdot k_2]B)$
- $[k]A \to A$
- $[k]A \to [!k][k]B$
- if A is an axiom, $\vdash_{S4_\bullet} [\bullet]A$
- $S4_\bullet$ (with $[k]$ read as knowledge operator) weakly avoids logical omniscience)
- justification logic ($\bullet$ replaced by axiom names) strongly avoids logical omniscience

Dynamic syntactic epistemic logic

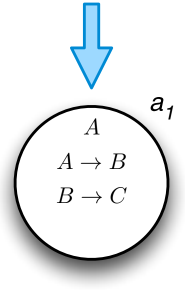
- Ho Ngoc Duc 1997: 'ϕ is true after some train of thought of agent *i*'
- adds a generic operator $\langle F_i \rangle$, for each agent *i*, to the language
- $\langle F_i \rangle K_i \phi$ means that agent *i* can get to know the formula ϕ some time in the future
- Duc presents a formal logical system $DES4_n$ for this language, intended to be a dynamic version of $S4_n$
- $DES4_n$ describes agents who do not necessarily know any consequences of their knowledge now, but can get to know any such consequence in the future
- a sound and complete semantics for $DES4_n$ is given in Ågotnes & Alechina 2006

More work on epistemic logics without omniscience

- Alechina & Logan 2001 (modal version of step logic)
- Ågotnes 2004 (PhD thesis on syntactic knowledge, knowing inference rules)
- Jago 2006 (PhD thesis on resource-bounded reasoning)
- Velázquez Quesada 2011 (PhD thesis on dynamics of information)
- ...

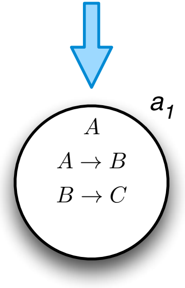
The basic idea of dynamic syntactic epistemic logic

Agent a_1's epistemic state

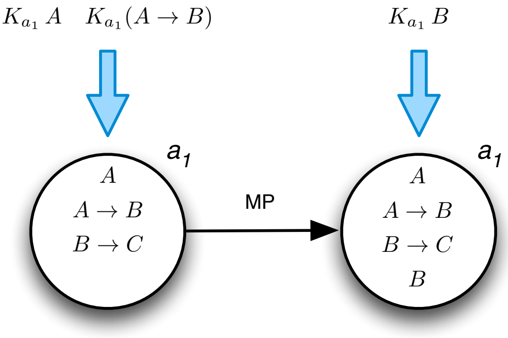


The basic idea of dynamic syntactic epistemic logic

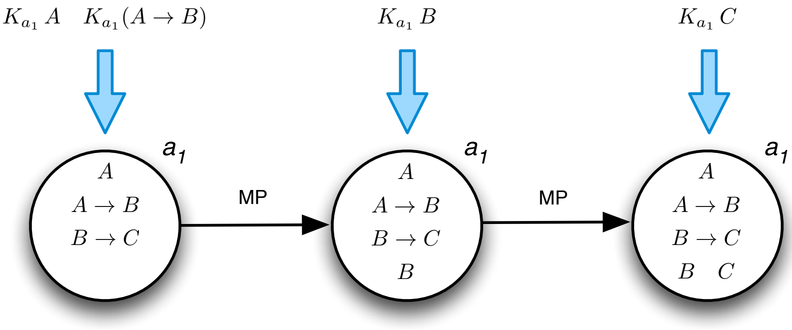
$K_{a_1} A \quad K_{a_1}(A \to B)$



Suppose the agent only knows Modus Ponens



Eventually it can derive all consequences by MP



Resources required for reasoning

- so far, we only looked at the number of steps/proof length
- what about memory required for reasoning?
- what about communication (in a multi-agent setting)?

Resource Logics

- variants of Alternating-Time Temporal Logic (ATL) where transitions have costs (or rewards) and the syntax can express resource requirements of a strategy, e.g.:

  agents A can enforce outcome φ if they have at most b_1 units of resource r_1 and b_2 units of resource r_2

- various flavours of resource logics exist: RBCL, RB-ATL, RB$\pm$ATL (Alechina et al.), RAL (Bulling & Farwer), PRB-ATL (Della Monica et al.), QATL* (Bulling & Goranko)

Verification Using Resource Logic

- one of the main problems in resource logics is model-checking
- model-checking problem: given a structure, a state in the structure and a formula, does the state satisfy the formula?
- using model-checking, we can verify resource requirements of a multi-agent system (specify the system as a model, and write a formula expressing a system objective)

Model-checking for Resource Logics

- for most resource logics the model-checking problem is undecidable: in particular, various flavours of RAL, and QATL*
- here, we present two resource logics with decidable model-checking problems:
  - RB-ATL which allows only consumption of resources
  - RB$\pm$ATL which allows unbounded production of resources

RB-ATL: syntax

- $Agt = \{a_1, \ldots, a_n\}$ a set of n agents
- $Res = \{res_1, \ldots, res_r\}$ a set of r resources,
- Π a set of propositions
- $B = \mathbb{N}^r_\infty$ a set of resource bounds, where $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$

RB-ATL: syntax

Formulas of RB-ATL are defined by the following syntax

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \psi \mid \langle\langle A^b \rangle\rangle \bigcirc \varphi \mid \langle\langle A^b \rangle\rangle \varphi \mathcal{U} \psi \mid \langle\langle A^b \rangle\rangle \Box \varphi$$

where $p \in \Pi$ is a proposition, $A \subseteq Agt$, and $b \in B$ is a resource bound.

RB-ATL: meaning of formulas

- $\langle\langle A^b \rangle\rangle \bigcirc \psi$ means that a coalition A can ensure that the next state satisfies φ under resource bound b
- $\langle\langle A^b \rangle\rangle \psi_1 \, \mathcal{U} \, \psi_2$ means that A has a strategy to enforce ψ while maintaining the truth of φ, and the cost of this strategy is at most b
- $\langle\langle A^b \rangle\rangle \Box \psi$ means that A has a strategy to make sure that φ is always true, and the cost of this strategy is at most b

Resource-bounded concurrent game structure

A RB-CGS is a tuple $M = (Agt, Res, S, \Pi, \pi, Act, d, c, \delta)$ where:

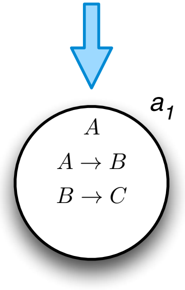
- *Agt* is a non-empty set of n agents, *Res* is a non-empty set of r resources and S is a non-empty set of states;
- Π is a finite set of propositional variables and $\pi : \Pi \to \wp(S)$ is a truth assignment
- *Act* is a non-empty set of actions which includes *idle*, and $d : S \times Agt \to \wp(Act) \setminus \{\emptyset\}$ is a function which assigns to each $s \in S$ a non-empty set of actions available to each agent $a \in Agt$
- $c : S \times Agt \times Act \to \mathbb{Z}^r$ (the integer in position i indicates consumption of resource res_i by the action a)
- $\delta : (s, \sigma) \mapsto S$ for every $s \in S$ and joint action $\sigma \in D(s)$ gives the state resulting from executing σ in s.

Additional assumptions and notation

- for every $s \in S$ and $a \in Agt$, $idle \in d(s, a)$
- $c(s, a, idle) = \bar{0}$ for all $s \in S$ and $a \in Agt$ where $\bar{0} = 0^r$
- we denote joint actions by all agents in Agt available at s by $D(s) = d(s, a_1) \times \cdots \times d(s, a_n)$
- for a coalition A, $D_A(s)$ is the set of all joint actions by agents in A
- $out(s, \sigma) = \{s' \in S \mid \exists \sigma' \in D(s) : \sigma = \sigma'_A \wedge s' = \delta(s, \sigma')\}$
- $cost(s, \sigma) = \sum_{a \in A} c(s, a, \sigma_a)$

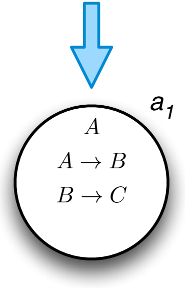
Example: dynamic syntactic epistemic logic in RB-ATL

Agent a_1's epistemic state



Example: dynamic syntactic epistemic logic in RB-ATL

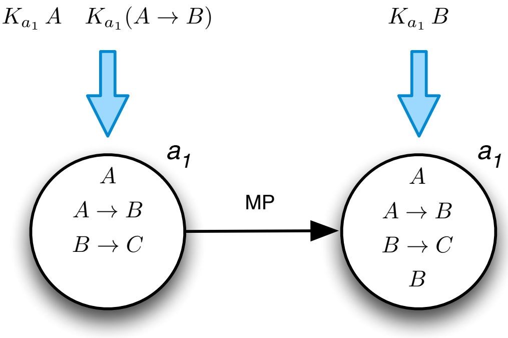
$K_{a_1} A \quad K_{a_1}(A \rightarrow B)$



Example dynamic syntactic epistemic logic in RB-ATL

Application of MP is an action that costs 1 unit of time and 1 unit of memory

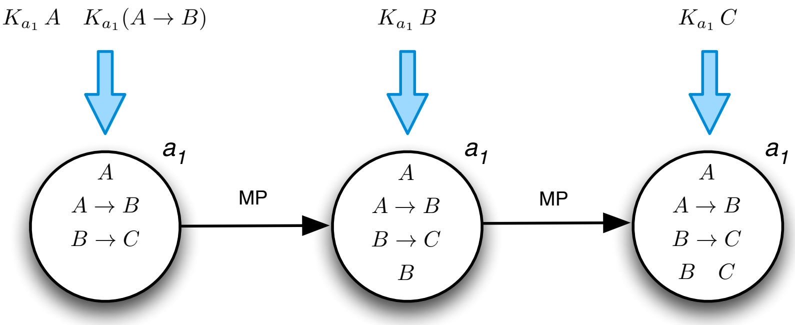
$$\langle\langle \{a_1\}^{time:1, memory:1} \rangle\rangle \bigcirc K_{a_1} B$$



Example: dynamic syntactic epistemic logic in RB-ATL

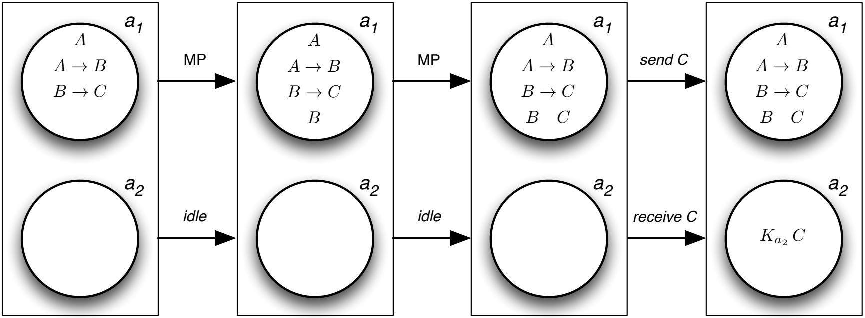
$$\langle\langle \{a_1\}^{time:1, memory:1} \rangle\rangle \bigcirc K_{a_1} B$$

$$\langle\langle \{a_1\}^{time:2, memory:2} \rangle\rangle \top \mathcal{U} K_{a_1} C$$



Example: extending to multi-agent case

$$\langle\!\langle \{a_1, a_2\}^{time:3, memory:3, energy:1} \rangle\!\rangle \top \, \mathcal{U} \, K_{a_2} K_{a_1} C$$



Strategies and their costs

- a *strategy for a coalition* $A \subseteq Agt$ is a mapping $F_A : S^+ \rightarrow Act$ such that, for every $\lambda s \in S^+$, $F_A(\lambda s) \in D_A(s)$
- a computation $\lambda \in S^\omega$ is consistent with a strategy F_A iff, for all $i \geq 0$, $\lambda[i+1] \in out(\lambda[i], F_A(\lambda[0,i]))$
- $out(s, F_A)$ the set of all consistent computations λ of F_A that start from s
- given a bound $b \in B$, a computation $\lambda \in out(s, F_A)$ is b-consistent with F_A iff, for every $i \geq 0$, $\sum_{j=0}^{i} cost(\lambda[j], F_A(\lambda[0,j])) \leq b$
- F_A is a b-strategy if all $\lambda \in out(s, F_A)$ are b-consistent

Truth definition

- $M, s \models \langle\langle A^b \rangle\rangle \bigcirc \phi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$: $M, \lambda[1] \models \phi$
- $M, s \models \langle\langle A^b \rangle\rangle \phi \mathcal{U} \psi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$, $\exists i \geq 0$: $M, \lambda[i] \models \psi$ and $M, \lambda[j] \models \phi$ for all $j \in \{0, \ldots, i-1\}$
- $M, s \models \langle\langle A^b \rangle\rangle \Box \phi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$ and $i \geq 0$: $M, \lambda[i] \models \phi$

Model-checking RB-ATL

The model-checking problem for RB-ATL is the question whether, for a given RB-CGS structure M, a state s in M and an RB-ATL formula ϕ, $M, s \models \phi$.

Theorem (Alechina, Logan, Nguyen, Rakib 2010):

The model-checking problem for RB-ATL is decidable

Model-checking algorithm for RB-ATL

function RB-ATL-LABEL(M, ϕ)
    for $\phi' \in Sub^+(\phi)$ **do**
        case $\phi' = p,\ \neg\psi,\ \psi_1 \wedge \psi_2$
            standard, see [Alur et al. 2002]
        case $\phi' = \langle\langle A^b \rangle\rangle \bigcirc \psi$
            $[\phi']_M \leftarrow Pre(A, [\psi]_M, b)$
        case $\phi' = \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$
            $[\phi']_M \leftarrow$ UNTIL-STRATEGY($M, \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$)
        case $\phi' = \langle\langle A^b \rangle\rangle \Box \psi$
            $[\phi']_M \leftarrow$ BOX-STRATEGY($M, \langle\langle A^b \rangle\rangle \Box \psi$)
    return $[\phi]_M$

$Sub^{+}(\phi_0)$

$Sub^{+}(\phi_0)$ includes all subformulas of ϕ_0, $Sub(\phi_0)$, and in addition:

- if $\langle\langle A^{b}\rangle\rangle\Box\psi \in Sub(\phi_0)$, then $\langle\langle A^{b'}\rangle\rangle\Box\psi \in Sub^{+}(\phi_0)$ for all $b' < b$
- if $\langle\langle A^{b}\rangle\rangle\psi_1 \mathcal{U} \psi_2 \in Sub(\phi_0)$, then $\langle\langle A^{b'}\rangle\rangle\psi_1 \mathcal{U} \psi_2 \in Sub^{+}(\phi_0)$ for all $b' < b$

$Sub^{+}(\phi_0)$ is partially ordered in increasing order of complexity and of resource bounds (e.g., if $b' \leq b$, $\langle\langle A^{b'}\rangle\rangle\Box\psi$ precedes $\langle\langle A^{b}\rangle\rangle\Box\psi$)

$Pre(A, \rho, b)$

$Pre(A, \rho, b)$ is a function which takes a coalition A, a set $\rho \subseteq S$ and a bound b, and returns the set of states s in which A has a joint action σ_A with $cost(s, \sigma_A) \leq b$ such that $out(s, \sigma_A) \subseteq \rho$

UNTIL-STRATEGY (RB-ATL)

function UNTIL-STRATEGY($M, \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$)

case $\phi' = \langle\langle A^{\bar{0}} \rangle\rangle \psi_1 \mathcal{U} \psi_2$:

$\rho \leftarrow [false]_M;\ \tau \leftarrow [\psi_2]_M$

while $\tau \not\subseteq \rho$ **do**

$\rho \leftarrow \rho \cup \tau;\ \tau \leftarrow Pre(A, \rho, \bar{0}) \cap [\psi_1]_M$

return ρ

case $\phi' = \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$ where $b > \bar{0}$:

$\rho \leftarrow [false]_M;\ \tau \leftarrow [false]_M$

foreach $b' < b$ **do**

$\tau \leftarrow Pre(A, [\langle\langle A^{b'} \rangle\rangle \psi_1 \mathcal{U} \psi_2]_M, b - b') \cap [\psi_1]_M$

while $\tau \not\subseteq \rho$ **do**

$\rho \leftarrow \rho \cup \tau;\ \tau \leftarrow Pre(A, \rho, \bar{0}) \cap [\psi_1]_M$

return ρ

BOX-STRATEGY (RB-ATL)

function BOX-STRATEGY($M, \langle\langle A^b \rangle\rangle \Box \psi$)

  case $\phi' = \langle\langle A^{\bar{0}} \rangle\rangle \Box \psi$:

    $\rho \leftarrow [true]_M$; $\tau \leftarrow [\psi]_M$

    while $\rho \not\subseteq \tau$ **do**

      $\rho \leftarrow \tau$; $\tau \leftarrow Pre(A, \rho, \bar{0}) \cap [\psi]_M$

   return ρ

  case $\phi' = \langle\langle A^b \rangle\rangle \Box \psi$ where $b > \bar{0}$:

    $\rho \leftarrow [false]_M$; $\tau \leftarrow [false]_M$

    foreach $b' < b$ **do**

      $\tau \leftarrow Pre(A, [\langle\langle A^{b'} \rangle\rangle \Box \psi]_M, b - b') \cap [\psi]_M$

      while $\tau \not\subseteq \rho$ **do**

        $\rho \leftarrow \rho \cup \tau$; $\tau \leftarrow Pre(A, \rho, \bar{0}) \cap [\psi]_M$

   return ρ

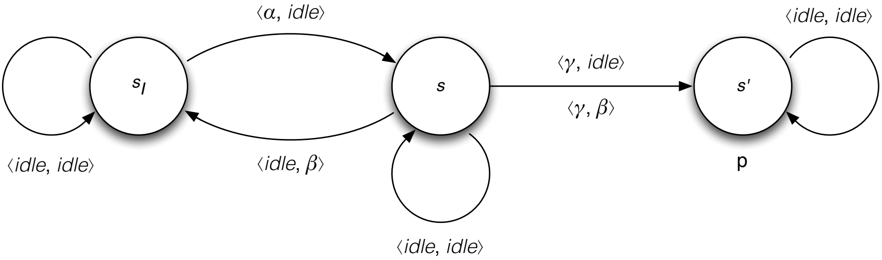
RB±ATL

- RB-ATL considers only consumption of resources
- a natural question is what happens if actions can produce as well as consume resources
- RB±ATL is a generalisation of RB-ATL where actions can produce resources

RB±ATL: syntax and semantics

- syntax and semantics are the same as RB-ATL, but production of resources is allowed
- $c : S \times Agt \times Act \to \mathbb{Z}^r$ (the integer in position i indicates consumption or production of resource res_i by the action a)
- if one agent consumes 10 units of resource and another agent produces 10 units of resource, the cost of their joint action is 0
- b-strategies are defined as before (the prefix of every computation generated by the strategy costs less than b)

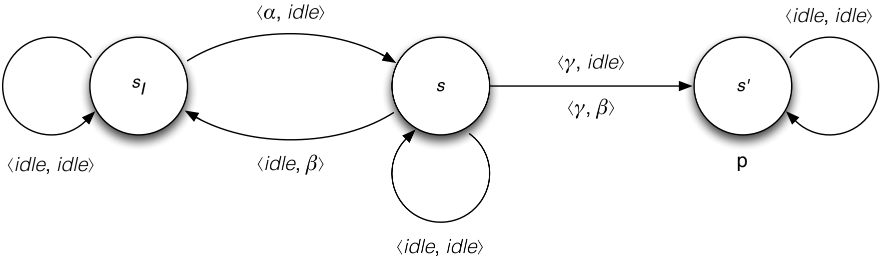
Example: two agents a_1, a_2, two resources r_1, r_2



Actions available to the first agent:

$d(s_I, a_1) = \{\alpha, idle\}$, $d(s, a_1) = \{\gamma, idle\}$, $d(s', a_1) = \{idle\}$

Example: two agents a_1, a_2, two resources r_1, r_2



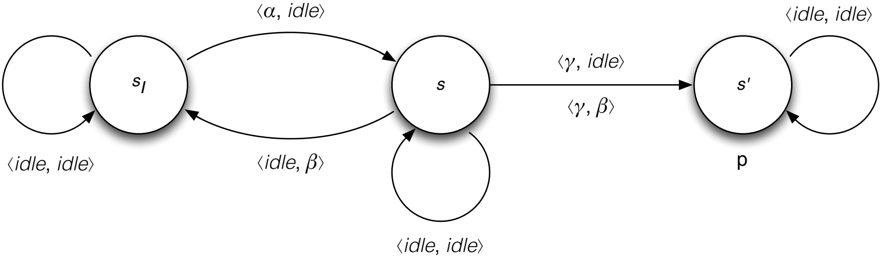
Actions available to the first agent:

$$d(s_I, a_1) = \{\alpha, idle\}, d(s, a_1) = \{\gamma, idle\}, d(s', a_1) = \{idle\}$$

Actions available to the second agent:

$$d(s_I, a_2) = \{idle\}, d(s, a_2) = \{\beta, idle\}, d(s', a_2) = \{idle\}$$

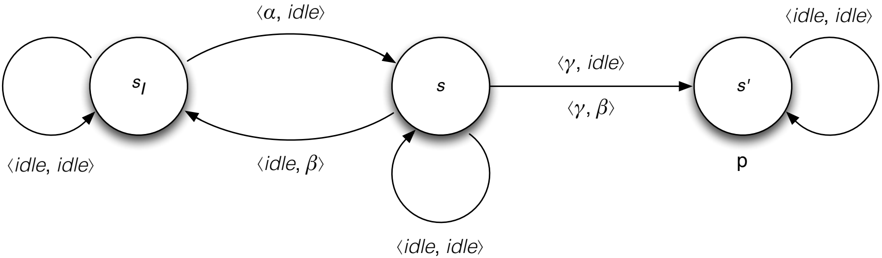
Example: two agents a_1, a_2, two resources r_1, r_2



Costs of actions:

$c(s_I, a_1, \alpha) = \langle -2, 1 \rangle$,
$c(s, a_1, \gamma) = \langle 5, 0 \rangle$,
$c(s, a_2, \beta) = \langle 1, -1 \rangle$

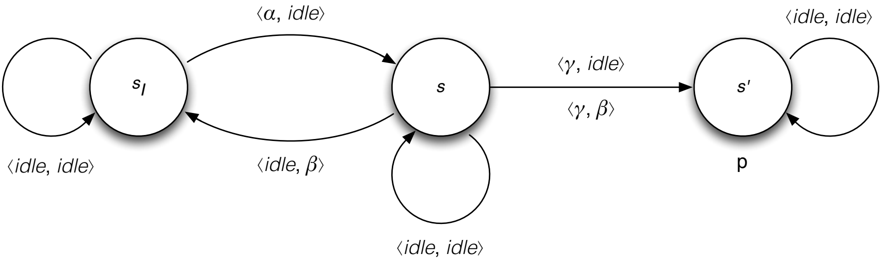
Example: strategy F_1 for a_1



$s_I \mapsto \alpha$
$s_I s \mapsto \gamma$
$s_I s s' \ldots \mapsto idle$
$out(s_I, F_1) = \{s_I, s, s', s', ...\}$

$c(s_I, a_1, \alpha) = \langle -2, 1\rangle$
$c(s, a_1, \gamma) = \langle 5, 0\rangle$
$c(s, a_1, idle) = \langle 0, 0\rangle$

Example: strategy F_1 for a_1



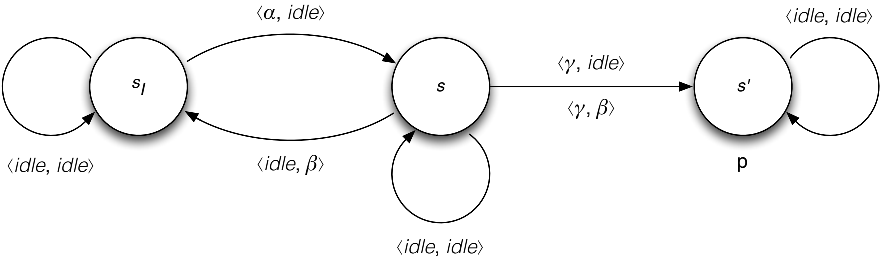
F_1 is a $\langle 3, 1 \rangle$-strategy:

$\langle -2, 1 \rangle \leq \langle 3, 1 \rangle$ $\qquad c(s_I, a_1, \alpha) = \langle -2, 1 \rangle$

$\langle -2, 1 \rangle + \langle 5, 0 \rangle \leq \langle 3, 1 \rangle$ $\qquad c(s, a_1, \gamma) = \langle 5, 0 \rangle$

$\langle -2, 1 \rangle + \langle 5, 0 \rangle + \langle 0, 0 \rangle \ldots \leq \langle 3, 1 \rangle$ $\qquad c(s, a_1, idle) = \langle 0, 0 \rangle$

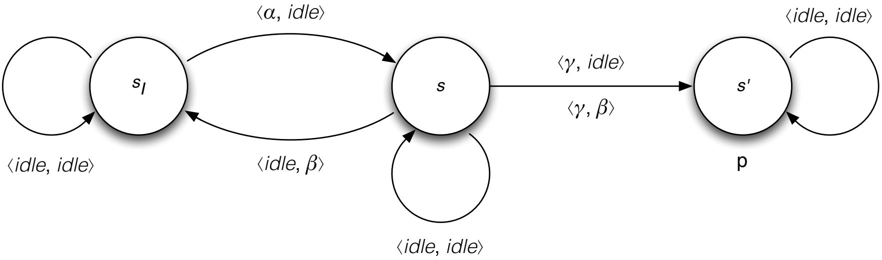
A strategy F for $A = \{a_1, a_2\}$



$s_I \mapsto \langle \alpha, idle \rangle$
$s_I s \mapsto \langle idle, \beta \rangle$
$s_I s s_I \mapsto \langle \alpha, idle \rangle, \ldots,$
$s_I s s_I s s_I s s_I s \mapsto \langle \gamma, idle \rangle$
$s_I s s_I s s_I s s_I s s' \ldots \mapsto \langle idle, idle \rangle$

$c(s_I, a_1, \alpha) = \langle -2, 1 \rangle$
$c(s, a_2, \beta) = \langle 1, -1 \rangle$
(repeat $s_I s s_I$ 4 times)
$c(s, a_1, \gamma) = \langle 5, 0 \rangle$

A strategy F for $A = \{a_1, a_2\}$



$out(s_I, F) = \{s_I, s, s_I, s, s_I, s, s_I, s, s', s', ...\}$

F is a $\langle 0, 1 \rangle$-strategy

Model-checking RB±ATL

The model-checking problem for RB±ATL is the question whether, for a given RB-CGS structure M, a state s in M and an RB±ATL formula ϕ, $M, s \models \phi$.

Theorem (Alechina, Logan, Nguyen, Raimondi 2014):
The model-checking problem for RB±ATL is decidable

Model-checking algorithm for RB±ATL

function RB±ATL-LABEL(M, ϕ)
  for $\phi' \in Sub(\phi)$ **do**
    case $\phi' = p,\ \neg\psi,\ \psi_1 \wedge \psi_2$
      standard, see [Alur et al. 2002]
    case $\phi' = \langle\langle A^b \rangle\rangle \bigcirc \psi$
      $[\phi']_M \leftarrow Pre(A, [\psi]_M, b)$
    case $\phi' = \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$
      $[\phi']_M \leftarrow \{\ s \mid s \in S \wedge$ UNTIL±STRATEGY$(node_0(s, b), \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2)\}$
    case $\phi' = \langle\langle A^b \rangle\rangle \Box \psi$
      $[\phi']_M \leftarrow \{\ s \mid s \in S \wedge$ BOX±STRATEGY$(node_0(s, b), \langle\langle A^b \rangle\rangle \Box \psi)\}$
  return $[\phi]_M$

Search tree nodes

- UNTIL±STRATEGY and BOX±STRATEGY proceed by depth-first and-or search of M
- for each tree node n, $s(n)$ returns its state, $p(n)$ returns the nodes on the path to n and $e_i(n)$ returns the resource availability on the i-th resource in $s(n)$ as a result of following $p(n)$
- $node_0(s, b)$ returns the root node ($s(n_0) = s$, $p(n_0) = [\,]$ and $e_i(n_0) = b_i$ for all resources i)
- $node(n, \sigma, s')$ returns a node n' where $s(n') = s'$, $p(n') = [p(n) \cdot n]$ and for all resources i, $e_i(n') = e_i(n) - cost_i(\sigma)$.

UNTIL±STRATEGY (RB±ATL)

function UNTIL±STRATEGY($n, \langle\langle A^b \rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2$)
    if $s(n) \not\models \langle\langle A \rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2$ **or**
      $\exists n' \in p(n) : s(n') = s(n) \wedge (\forall j : e_j(n') \geq e_j(n))$ **then**
        return *false*
    for $i \in \{i \in Res \mid \exists n' \in p(n) : s(n') = s(n) \wedge (\forall j : e_j(n') \leq e_j(n)) \wedge$
$e_i(n') < e_i(n)\}$ **do**
        $e_i(n) \leftarrow \infty$
    if $s(n) \models \psi_2$ **or** $e(n) = \bar{\infty}$ **then**
        return *true*
    for $\sigma \in \{\sigma \in D_A(s(n)) \mid cost(\sigma) \leq e(n)\}$ **do**
        $strat \leftarrow true$
        for $s' \in out(s(n), \sigma)$ **do**
            $strat \leftarrow strat \wedge$ UNTIL±STRATEGY($node(n, \sigma, s'), \langle\langle A^b \rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2$)
        if *strat* **then**
            return *true*
    return *false*

BOX±STRATEGY (RB±ATL)

function BOX±STRATEGY(n, $\langle\langle A^b \rangle\rangle \Box \psi$)
  if $s(n) \not\models \langle\langle A \rangle\rangle \Box \psi$ **or**
    $\exists n' \in p(n) : s(n') = s(n) \wedge (\forall j : e_j(n') > e_j(n))$ **then**
      return *false*
  if $\exists n' \in p(n) : s(n') = s(n) \wedge (\forall j : e_j(n') \leq e_j(n))$ **then**
      return *true*
  for $\sigma \in \{\sigma \in D_A(s(n)) \mid cost(\sigma) \leq e(n)\}$ **do**
      $strat \leftarrow true$
      for $s' \in out(s(n), \sigma)$ **do**
          $strat \leftarrow strat \wedge$ BOX±STRATEGY($node(n, \sigma, s'), \langle\langle A^b \rangle\rangle \Box \psi$)
      if *strat* **then**
          return *true*
  return *false*

Complexity

- the model-checking problem for RB$\pm$ATL is EXPSPACE-hard
- special cases have lower complexity:
  - one resource: PSPACE
  - no production (RB-ATL): PTIME in formula and transition system, exponential in the number of resources

Open problems

There are many open problems in both areas

- other tractable cases of resource reasoning
- modelling combinations of reasoning and acting in resource logics
- accounting for the costs of observation and communication in dynamic epistemic logic
- etc.

Infinite bound versions

Since the infinite resource bound version of RB-ATL modalities correspond to the standard ATL modalities, we write

- $\langle\langle A^{\bar{\infty}} \rangle\rangle \bigcirc \phi$ as $\langle\langle A \rangle\rangle \bigcirc \phi$
- $\langle\langle A^{\bar{\infty}} \rangle\rangle \phi \mathcal{U} \psi$ as $\langle\langle A \rangle\rangle \phi \mathcal{U} \psi$
- $\langle\langle A^{\bar{\infty}} \rangle\rangle \Box \phi$ as $\langle\langle A \rangle\rangle \Box \phi$

Auxiliary functions: $split(b)$

$split(b)$ is a function that takes a resource bound b and returns the set of all pairs $(d, d') \in \mathbb{N}_\infty \times \mathbb{N}_\infty$ such that:

1. $d + d' = b$
2. $d_i = d'_i = \infty$ for all $i \in \{1, \dots, r\}$ where $b_i = \infty$
3. d has at least one non-0 value

The set of all pairs (d, d') is partially ordered in increasing order of d' (i.e., if $d'_1 < d'_2$, then (d_1, d'_1) precedes (d_2, d'_2))

Auxiliary functions: $Sub^+(\phi_0)$

$Sub^+(\phi_0)$ includes all subformulas of ϕ_0, $Sub(\phi_0)$, and in addition:

- if $\langle\langle A^b \rangle\rangle \Box \psi \in Sub(\phi_0)$, then $\langle\langle A^{d'} \rangle\rangle \Box \psi \in Sub^+(\phi_0)$ for all d' such that $(d, d') \in split(b)$
- if $\langle\langle A^b \rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2 \in Sub(\phi_0)$, then $\langle\langle A^{d'} \rangle\rangle \psi_1 \,\mathcal{U}\, \psi_2 \in Sub^+(\phi_0)$ for all d' such that $(d, d') \in split(b)$

$Sub^+(\phi_0)$ is partially ordered in increasing order of complexity and of resource bounds (e.g., if $b' \leq b$, $\langle\langle A^{b'} \rangle\rangle \Box \psi$ precedes $\langle\langle A^b \rangle\rangle \Box \psi$)

$Pre(A, \rho, b)$

$Pre(A, \rho, b)$ is a function which takes a coalition A, a set $\rho \subseteq S$ and a bound b, and returns the set of states s in which A has a joint action σ_A with $cost(s, \sigma_A) \leq b$ such that $out(s, \sigma_A) \subseteq \rho$

UNTIL-STRATEGY (RB-ATL)

function UNTIL-STRATEGY$(M, \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2)$
  case $\phi' = \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$ where $\forall i\ b_i \in \{0, \infty\}$:
    $\rho \leftarrow [false]_M;\ \tau \leftarrow [\psi_2]_M$
    while $\tau \not\subseteq \rho$ **do**
      $\rho \leftarrow \rho \cup \tau;\ \tau \leftarrow Pre(A, \rho, b) \cap [\psi_1]_M$
   return ρ
  case $\phi' = \langle\langle A^b \rangle\rangle \psi_1 \mathcal{U} \psi_2$ where $\exists i\ b_i \notin \{0, \infty\}$:
    $\rho \leftarrow [false]_M;\ \tau \leftarrow [false]_M$
    foreach $d' \in \{d' \mid (d, d') \in split(b)\}$ **do**
      $\tau \leftarrow Pre(A, [\langle\langle A^{d'} \rangle\rangle \psi_1 \mathcal{U} \psi_2]_M, d) \cap [\psi_1]_M$
      while $\tau \not\subseteq \rho$ **do**
        $\rho \leftarrow \rho \cup \tau;\ \tau \leftarrow Pre(A, \rho, \bar{0} \overset{\infty}{\leftarrow} b) \cap [\psi_1]_M$
   return ρ

BOX-STRATEGY (RB-ATL)

function BOX-STRATEGY($M, \langle\langle A^b \rangle\rangle \Box \psi$)
  case $\phi' = \langle\langle A^b \rangle\rangle \Box \psi$ where $\forall i\ b_i \in \{0, \infty\}$:
    $\rho \leftarrow [true]_M;\ \tau \leftarrow [\psi]_M$
    while $\rho \not\subseteq \tau$ **do**
      $\rho \leftarrow \tau;\ \tau \leftarrow Pre(A, \rho, b) \cap [\psi]_M$
   return ρ
  case $\phi' = \langle\langle A^b \rangle\rangle \Box \psi$ where $\exists i\ b_i \notin \{0, \infty\}$:
    $\rho \leftarrow [false]_M;\ \tau \leftarrow [false]_M$
    foreach $d' \in \{d' \mid (d, d') \in split(b)\}$ **do**
      $\tau \leftarrow Pre(A, [\langle\langle A^{d'} \rangle\rangle \Box \psi]_M, d) \cap [\psi]_M$
      while $\tau \not\subseteq \rho$ **do**
        $\rho \leftarrow \rho \cup \tau;\ \tau \leftarrow Pre(A, \rho, \bar{0} \overset{\infty}{\leftarrow} b) \cap [\psi]_M$
   return ρ