Formally reasoning about strategic behaviour

Sasha Rubin

University of Naples “Federico II”, Italy

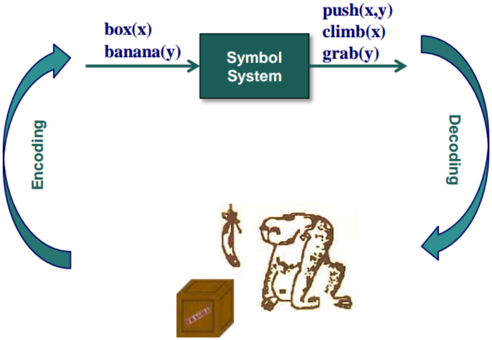
February 7, 2019, University of Sydney

What is an agent?

- An agent is something that *acts*, based on sensing the world.
- An agent is *rational* if it acts to achieve the best (expected) outcome. Russell, Norvig

What is an agent?

- An agent is something that *acts*, based on sensing the world.
- An agent is *rational* if it acts to achieve the best (expected) outcome. Russell, Norvig



…Multi-agent systems are everywhere!

Many real and imagined systems consist of multiple agents:

1. **distributed**, i.e., agents have their own view of the world.
2. **goal-directed**, i.e., agents strategise to achieve their own goals.

E.g.

- multiplayer games (winning tasks)
- robot assisted search-and-rescue (co-ordination tasks)
- autonomous transport vehicles (safety tasks)
- voting and auction protocols (fairness tasks)
- …

Research Challenge

How to predict and control agent behaviour?

Lots of disciplines involved

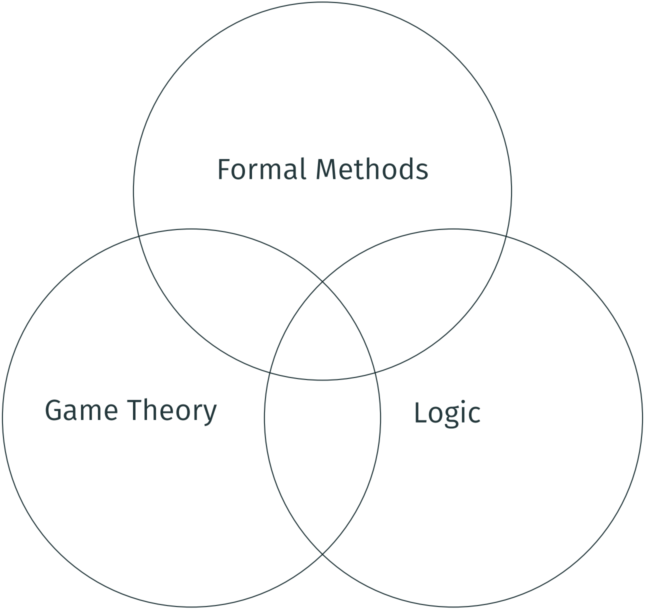
1. Humanities: How do we want agents to behave?
2. Computer engineering: How to engineer such agents?
3. Computer science: How to make sure agents behave as we want?
    3.1 Theoretical foundations
    3.2 Academic tools
    3.3 Industrial applications

Research Challenge

How to predict and control agent behaviour?

- **Methodology**
  Model agent tasks in logic, and devise algorithms for predicting/controlling.
- **Difficulty**
  Agents are distributed.

Methodology



Formal Methods 101

$$\textbf{Model-checking paradigm}^{1} = \begin{cases} \text{model} & M \\ \text{specify} & \varphi \\ \text{algorithm for} & M \models \varphi \end{cases}$$

[1] If formula φ talks about strategies, model-checking can usually be adapted to solve synthesis (not just verification).

Formal Methods 101

$$\textbf{Model-checking paradigm}^{1} = \begin{cases} \text{model} & M \\ \text{specify} & \varphi \\ \text{algorithm for} & M \models \varphi \end{cases}$$

Classical model-checking can't handle the complexities of multi-agent systems.

[1] If formula φ talks about strategies, model-checking can usually be adapted to solve synthesis (not just verification).

Formal Methods 101

Different types of Domains

# Ag	CS Systems	Planning Domains	Game Example
1	closed	deterministic	solitaire
2	open	non-deterministic	tic-tac-toe
$>$	distributed	multi-agent	bridge

Verification/Synthesis Problems

# Ag	Is there/find ...
1	a sequence of actions that results in a good state?
2	a winning strategy?
$>$	a strategy profile that is in equilibrium?

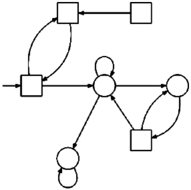
My current research goals

$$\textbf{Model-checking paradigm} = \begin{cases} \text{model} & M \\ \text{specify} & \varphi \\ \text{algorithm for} & M \models \varphi \end{cases}$$

1. Devise models that capture essential aspects of multi-agent systems.
2. Devise logics that can specify complex agent/system goals.
3. Devise algorithms for model-checking, or prove that none exists.

The usual tradeoff applies: the more expressive the logic and models, the less likely algorithms exist.

Models

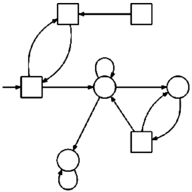


$M = \left\langle S, \mathrm{tr}, (Obs_a)_{a \in Ag} \right\rangle$ is a transition system modeling dynamics

S	set of states
$\mathrm{tr} : S \times Act \to S$	transition function
$Obs_a : S \to \Omega$	observation function

Special case: $Obs_a(s) = s$ Full observation

Models



$M = \langle S, \mathrm{tr}, (Obs_a)_{a \in Ag} \rangle$ is a transition system modeling dynamics

S	set of states
$\mathrm{tr} : S \times Act \to S$	transition function
$Obs_a : S \to \Omega$	observation function

Special case: $Obs_a(s) = s$ Full observation

- Strategy tells an agent how to act based on its observations
- Strategy for each agent determines a path in M

Logics

New logic: SL (strategy logic for agents with partial observation)

$$\overbrace{p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2}^{\text{Boolean}} \mid \overbrace{\mathit{next}\,\varphi \mid \varphi_1\,\mathit{until}\,\varphi_2}^{\text{Temporal}} \mid \overbrace{\exists x \varphi \mid x = y \mid (a_i\,\mathit{uses}\,x)\,\varphi}^{\text{Strategic}}$$

Atoms p: $\mathit{NotCrashed}(a_1)$, $\mathit{InRoom}(a_1, r_2)$, ...

Temporal goals ψ: $\mathit{always}\,\neg p$, $\mathit{eventually}\,p$, ...

Strategic properties:

- $\exists x_1\,(a_1\,\mathit{uses}\,x_1)\,\psi$ — I win!
- $\exists! x_1\,(a_1\,\mathit{uses}\,x_1)\,\psi$ — There is only one way to win!
- $\exists x_1 \exists x_2\,(a_i\,\mathit{uses}\,x_i)_i\,\psi$ — We win!
- $\exists x_1 \exists x_2\,(a_i\,\mathit{uses}\,x_i)_i\,\forall y \wedge_i [(a_i\,\mathit{uses}\,y)\,\psi_i \rightarrow \psi_i]$ — Nash equilibrium!
- $\exists x_1 \exists x_2\,(a_i\,\mathit{uses}\,x_i)_i\,[\psi \wedge \forall y\ \wedge_i (a_i\,\mathit{uses}\,y)\,\psi_{1-i}]$

Logics

New logic: SL (strategy logic for agents with partial observation)

$$\overbrace{p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2}^{\text{Boolean}} \mid \overbrace{\mathit{next}\,\varphi \mid \varphi_1\,\mathit{until}\,\varphi_2}^{\text{Temporal}} \mid \overbrace{\exists x \varphi \mid x = y \mid (a_i\,\mathit{uses}\,x)\,\varphi}^{\text{Strategic}}$$

Agent a_i has temporal goal ψ_i:

- Achievement: eventually reach room r_2.
- Safety: agent a_1 never collide with other agents.

Properties φ of agent strategies:

- Winning: strategy guarantees temporal goal.
- Equilibria: strategies form a Nash equilibrium.
- Evolutionary stable: all agents adopt it, no mutant can invade.

Algorithms for $M \models \varphi$

Distributed agents make things difficult…

Distributed agents make things difficult...

Distributed agents make things difficult...

- Multi-player games in TCS (70s) — undecidable
- Distributed synthesis in FM (90s) — undecidable
- Finite horizon DEC-POMDPs in AI (00s) — NEXP-complete

Insight. The source of these difficulties is the ability of agents, each with their own observations of the world, to communicate privately with each other.

Distributed agents make things difficult...

What can be done?

1. Restrict observability
2. Restrict private communication

Distributed agents make things difficult…

What can be done? Restrict observability

How? Hierarchical observation/information

Distributed agents make things difficult...

What can be done? Restrict observability

How? Hierarchical observation/information

- Peterson, Reif (1979)
- Pnueli, Rosner (1990)
- Kupferman, Vardi (2001)
- Finkbeiner, Schewe (2005)
- Berwanger, Mathew, vdBogaard (2016)
- Berthon, Maubert, Murano, R., Vardi (LICS 2017)

Reasonable concern: None of the examples mentioned have hierarchical observation

Distributed agents makes model-checking difficult...

What can be done? Restrict private communication

How? Broadcast-communication/public-actions

Distributed agents makes model-checking difficult...

What can be done? Restrict private communication

How? Broadcast-communication/public-actions

- Lomuscio, vdMeyden, Ryan, Wilke (2000, 2005) Broadcasting
- vDitmarsch, vdHoek ... Public Announcements Logic
- Kominis, Geffner (2017) Public Actions (Planning)
- Belardinelli, Lomuscio, Murano, R. (AAMAS 2017, IJCAI 2017)

Theorem. $M \models \varphi$ is decidable if M is a public-action transition system.

Models

New Model: Public action transition system M — the last action of each agent is observable to all agents.

E.g.,

- community-card games (bridge, poker)
- broadcasting distributed systems (tweeting)
- epistemic puzzles (muddy children)
- auctions (open-outcry)

3. Algorithms for $M \models \varphi$

Theorem. $M \models \varphi$ is decidable if M is a public-action transition system.

Also. Computational complexity[2] is (roughly) the same as fully-observable transition systems!

Middle ground.

Fully observable TS $<$ Public Action TS $<$ Unrestricted TS

[2] For formulas of quantifier depth k: in $(k+2)$-exptime and $(k-1)$-expspace-hard.

Proofs?

- Q. How did you come up with the algorithms?
- A. We used the automata-theoretic approach to model-checking.[3]
- Q. Ok, but what do the algorithms actually look like?
- A. Operations on *automata that run on trees*.
- Q. Huh?
- A. Think of it like manipulating regular expressions that match trees instead of strings!

[3] for which Vardi and Wolper won the Gödel prize.

Proofs?

WHENEVER I LEARN A NEW SKILL I CONCOCT ELABORATE FANTASY SCENARIOS WHERE IT LETS ME SAVE THE DAY.
OH NO! THE KILLER MUST HAVE FOLLOWED HER ON VACATION!
BUT TO FIND THEM WE'D HAVE TO SEARCH THROUGH 200 MB OF EMAILS LOOKING FOR SOMETHING FORMATTED LIKE AN ADDRESS!
IT'S HOPELESS!
EVERYBODY STAND BACK.
I KNOW REGULAR EXPRESSIONS.
PERL!
TAP TAP

Summary

- Research goals:
  1. Devise models that capture essential aspects of multi-agent systems.
  2. Devise logics that can specify complex agent/system goals.
  3. Devise algorithms for model-checking, or prove that none exists.
- Challenge: Combination of partial observability and private communication.
- Solutions: Impose hierarchical observation, or restrict private communication.

Important next steps (ongoing)

Probabilistic agents

Context. Many protocols rely on coin-flips.

Challenge. Verification problems are computationally intractable.

Project. Developing *Probabilistic Strategy Logic* for reasoning about stochastic multi-agent transition systems.

Methodology. Study restrictions on memory of probabilistic agents and reduce to the first-order theory of real arithmetic.

Important next steps (planned)

Applications

Context. Existing tools for multi-agent systems (*MCK* vdMeyden et. al.; *Eve* Wooldridge et. al.; *MCMAS* Lomuscio et. al.) can't handle logics and models above.

Challenge. High computational complexity, complex constructions.

Project. Engineer practical algorithms by translating to automated planning (a form of synthesis).

Methodology. Focus on finite paths of M.

Overview of other recent research

Formal methods …

1. …for strategising agents with qualitative temporal goals

   AAMAS'16/'17/'18, IJCAI'17, LICS'17
2. …for strategising agents with quantitative temporal goals

   IJCAI'17, CSL'18
3. …when the number of agents is not known apriori

   M&C'15 (book), IJCAR'16, VMCAI'18
4. …when the environment is partially-known

   PRIMA'15 (best-paper), AAMAS'15/'16, IJCAI'16/'17

Overview of other recent research

Formal methods …

1. …for strategising agents with qualitative temporal goals

   AAMAS'16/'17/'18, IJCAI'17, LICS'17

2. …for strategising agents with quantitative temporal goals

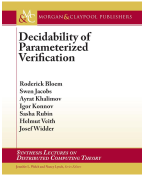
   IJCAI'17, CSL'18

3. …when the number of agents is not known apriori

   M&C'15 (book), IJCAR'16, VMCAI'18

4. …when the environment is partially-known

   PRIMA'15 (best-paper), AAMAS'15/'16, IJCAI'16/'17



Overview of other recent research

Formal methods …

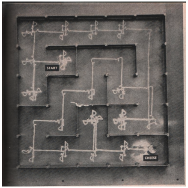
1. …for strategising agents with qualitative temporal goals

   AAMAS’16/’17/’18, IJCAI’17, LICS’17
2. …for strategising agents with quantitative temporal goals

   IJCAI’17, CSL’18
3. …when the number of agents is not known apriori

   M&C’15 (book), IJCAR’16, VMCAI’18
4. …when the environment is partially-known

   PRIMA’15 (best-paper), AAMAS’15/’16, IJCAI’16/’17



European Commission

About me

’04	Auckland	Automatic Structures	Khoussainov (cf. Cannon)
’07	Aachen	Automata, Logic	Grädel
’09	Cornell	Teaching	incl. REU
’12	Vienna	Verification	Veith, Bloem

’15	Naples		
		Verification	Vardi, Murano
		Planning	DeGiacomo, Geffner
		MAS	Wooldridge, Lomuscio

Organisation

’13-’16	Proj. co-ord. Handbook of Model Checking (Springer ’18)
’17	Italian Conf. Theoretical Computer Science
’17	Italian Conf. Computational Logic
’17	Wk. Formal Methods in Artificial Intelligence (Naples)
	Geffner (Planning), De Giacomo (KR), Wooldridge (MAS)
’19	Work. Formal Methods **and** Artificial Intelligence (Rennes)
	Nowé (RL), McIlraith (AI,Planning)

What sort of researcher am I?

1. I'm part of a growing group bringing insights from formal methods to bear on problems in AI.
   (De Giacomo, Lomuscio, Vardi, Wooldridge)
2. I'm rooted in Formal Methods and Knowledge Representation philosophies.
3. My focus has broadened over time:
   Automata and mathematics $\rightarrow$ Formal Methods $\rightarrow$ Artificial Intelligence