

ANONYMITY AND INFORMATION HIDING IN  
MULTIAGENT SYSTEMS:  
A KNOWLEDGE-BASED APPROACH

Joseph Halpern

&

Kevin O'Neill

Computer Science Department  
Cornell University

# Motivation

---

Anonymity is important to people in a variety of situations:

- Browsing the web
- Sharing files with other Internet users
- Sending messages
- Real-life situations, such as:
  - making large donations
  - whistle-blowing

Often people will be reluctant to engage in some behavior unless they can receive guarantees that their anonymity will be protected.

# Achieving Anonymity

There are protocols and systems that guarantee anonymity in restricted situations:

- DC-nets based on the “dining cryptographers” protocol
- Anonymizer
- Crowds
- Herdovore

These systems are all quite different, and offer different kinds of anonymity guarantees.

We want to be able to compare the guarantees provided by these systems using a formal framework.

# A Formal Framework for Anonymity

Ideally, a formal framework for anonymity should:

- let us define different kinds of anonymity guarantees in a precise, intuitive way.
- model real-world systems.
- provide a way to verify formally that a given system provides a desired anonymity guarantee.

# An Example

---

An anonymous message-passing system:

- Agents send messages (i.e., email) to other agents in the system
- When sending a message, agents may sometimes want to ensure that:
  - the message is sent anonymously
  - the message is received anonymously
  - the message is both sent and received anonymously

# Defining Anonymity

---

- We define anonymity as an instance of “information hiding”, where we ask:
  - what information needs to be hidden?
  - who does it need to be hidden from?
  - how well does it need to be hidden?
- Anonymity is closely related to the *knowledge* of the agents interacting with the system.
  - Our definitions of anonymity use knowledge in a formal way.
- We relate anonymity to our earlier work on secrecy and noninterference.

# Representing Multiagent Systems:

Our model lets us represent all possible behaviors of the system as well as the state of the agents who use the system.

- $n$  agents, each in some local state  $s_i$  at a given point in time
- The whole system in some global state  $(s_1, \dots, s_n, s_e)$
- A run  $r$  is a function from time to global states
- A *point* of the system is a pair  $(r, m)$ —a particular execution sequence at a particular point in time
- A system  $\mathcal{R}$  is a set of runs

# Local States and Knowledge

We write  $r_i(m) = s_i$  if  $i$  has local state  $s_i$  at point  $(r, m)$ .

At the point  $(r, m)$ , agent  $i$  considers possible all the points  $(r', m')$  such that  $r_i(m) = r'_i(m')$ .

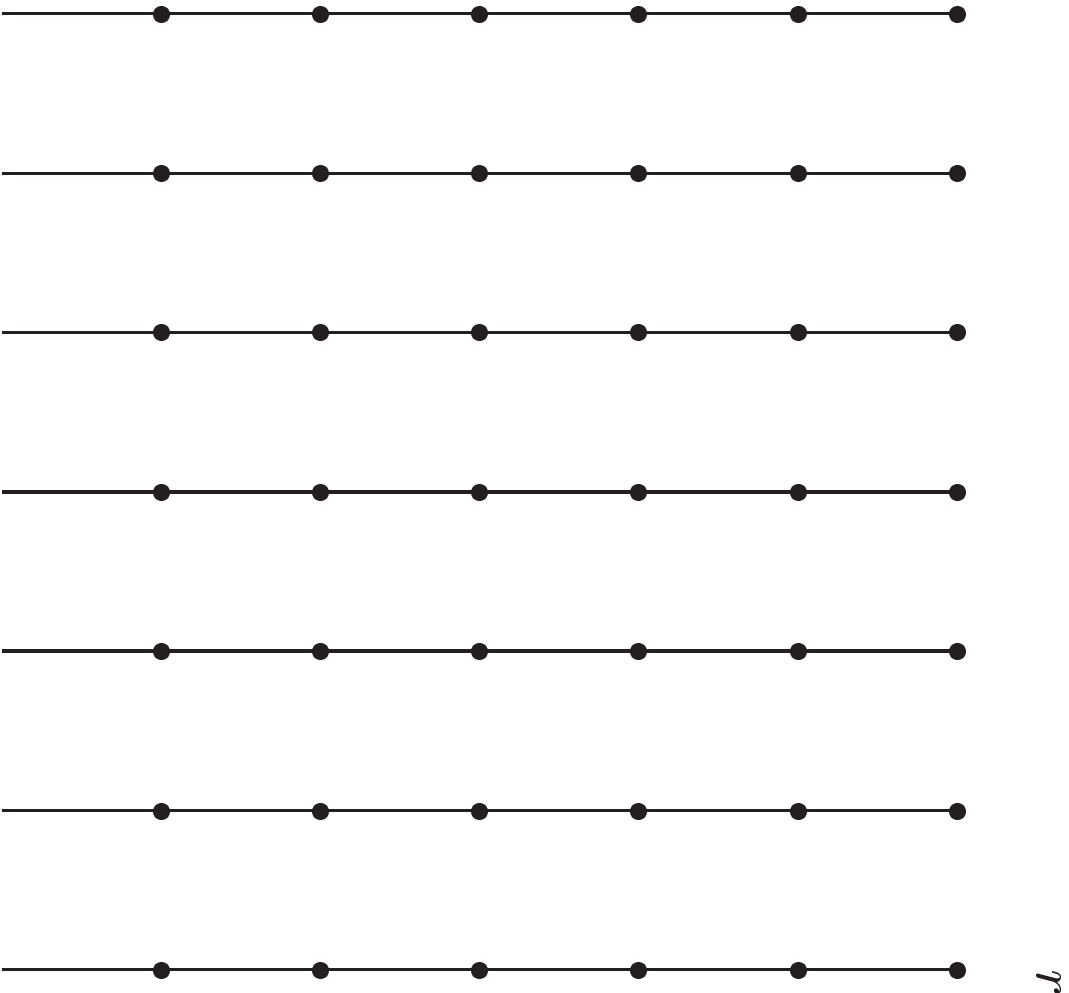
If a fact  $\varphi$  is true at all points that  $i$  considers possible, we say that “ $i$  knows the fact  $\varphi$ ”.

- Denoted “ $K_i\varphi$ ”

If a fact  $\varphi$  is true at some point that  $i$  considers possible, we say that “ $i$  considers possible the fact  $\varphi$ ”.

- Denoted “ $P_i\varphi$ ”
- $P_i\varphi$  iff  $\neg K_i\neg\varphi$

A system:



$$r(\mathbf{1}) = (s_e, s_1, \dots, s_n)$$

# Defining Anonymity

We define anonymity in terms of actions and the agents who perform them.

- Let  $\delta(i, a)$  represent the fact that  $i$  has performed action  $a$
- Action  $a$ , performed by agent  $i$ , is *minimally anonymous* with respect to agent  $j$  in  $\mathcal{R}$  if the formula “ $\neg K_j[\delta(i, a)]$ ” is always true.
- If an observer  $j$  knows that  $i$  sent a message, then  $i$  doesn't have any anonymity, at least with respect to  $j$ .

Minimal anonymity is a very weak condition:

- Minimal anonymity holds as long as  $j$  is not 100% sure that  $i$  performed action  $a$ .

## **A Stronger Version of Anonymity**

An agent  $i$  might want to ensure that observers think it possible that many agents, perhaps all the agents in some “anonymizing set”  $A$ , could have performed the anonymous action.

Action  $a$ , performed by agent  $i$ , is *anonymous up to A* with respect to agent  $j$  in  $\mathcal{R}$  if the following formula is always true:

$$\delta(i, a) \Rightarrow \bigwedge_{i' \in A} P_j[\delta(i', a)].$$

Anonymity up to  $A$  is clearly more restrictive than minimal anonymity.

- *Total anonymity* is an even stronger condition.

# Probabilistic Definitions of Anonymity

Problems with “possibilistic” guarantees:

- Suppose an observer  $o$  thinks that any of 101 agents in a set  $A$  could have performed an action  $a$ .
- What if  $o$  has a probability of 0.99 that  $i$  performed  $a$ , and a probability of 0.0001 that any of the other 100 agents performed  $a$ ?
- Here anonymity up to  $A$  doesn't provide much comfort to  $i$ ...

We describe how probability can be added to the multiagent systems framework, and we give examples of stronger guarantees of anonymity that use probability.

- Previous formalizations have not dealt with probability.

# Conditional Anonymity

---

Consider an anonymous message-passing system.

- Even if the *system* makes it impossible to trace the message to my identity, the *content* of my messages may leak information.
- Observers will have prior probabilities on what various agents might do in a given system.
  - Neither Joe nor Kevin is likely to make a multimillion-dollar donation to Cornell!
- This makes it fundamentally difficult to give strong probabilistic anonymity guarantees for a real-world system.
- We give a new definition of *conditional anonymity*.
  - It's related to our (much stronger) definition of secrecy.

# Related Work

---

Others have formalized anonymity:

- using epistemic logics [Syverson and Stubblebine, 1999];
- using CSP [Schneider and Sidiropoulos, 1996];
- using function views [Hughes and Shmatikov, 2002].
  - Actually, our work was inspired by Vitaly’s talk at the fall SPYCFE meeting!
- Many of our definitions have been given before, but we show that these different definitions can all be captured cleanly in one framework.

Analyses of real-world anonymity systems:

- Shmatikov [2002] analyzes the Crowds system using a probabilistic model checker.

# One Application: CSP and Anonymity

Schneider and Sidiropoulos define anonymity in terms of CSP.

- Let  $A$  be a set of “anonymous events”.
- A process  $P$  is strongly anonymous on  $A$  if  $f_1^{-1}(f_A(P)) = P$  (where  $f_A$  is a particular renaming function).
- This definition is not very intuitive, but can be used to verify real-world protocols using model checkers for CSP.

We show that this definition is a special case of our definitions.

- A process  $P$  can be associated with a set of runs  $\mathcal{R}_P$ , and the set  $A$  with a particular action  $a$  and set of agents  $I_A$ .
- Theorem:  $P$  is strongly anonymous on  $A$  if and only if actions in  $A$  are anonymous up to  $I_A$ .

# For the Future

---

Verification is an eventual goal:

- Using the knowledge-based framework directly [van der Meyden, 1998],
- Or indirectly, using a related framework such as CSP or the  $\pi$ -calculus.

We would like to say more about the relationship between the knowledge-based system framework and the process algebra framework:

- We want a canonical translation from processes to multiagent systems so that information-hiding properties specified using knowledge make sense for systems specified using process algebras.