Normative Multi-Agent Systems: An Introduction

Marina De Vos

Department of Computer Science
University of Bath
cssmdv@bath.ac.uk

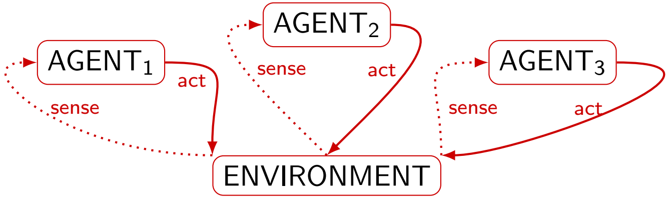
art-ai UNIVERSITY OF BATH

History Part 1

- Autonomous system
- Social Technical Systems
  - Autonomy
  - Guidance
  - Guardrails
- $\Rightarrow$ Norms, Values
- External to the agent
- Institutions/Normative Frameworks

Socio-technical Systems

- Humans agents and Virtual agents interact with "some" autonomy
- An agent can be more useful in the context of others:
  - Can concentrate on tasks within competence
  - Can delegate other tasks
  - Can use ability to communicate, coordinate, negotiate
  - But lacks organisation...



History

- From Utility/Game Theory to Institutions:
  - GT enables strategic analysis
  - ... but games are (relatively) simple
  - participants make bounded rational choices
- Negotiation and Contract Net:
  - Typically one-shot encounters
  - Components in more complex scenarios
- More complex frameworks with stronger guarantees:
  - Coalition: A group of agents, different skills
  - Virtual Organization: A group of agents, subject to a particular, agreed regulatory framework
  - Virtual Institution: A pattern of (regulated/governed) actions, sanctions, roles and goals

Why Normative Systems are Essential

- Unconstrained behaviour is not freedom
- (Multiple) Institutions abstract the interaction frameworks needed for guiding behaviour
- Agents can negotiate institutional change
- Institutions can be repositories of emergent behaviour
- Institutions can be formalized and reasoned about with limited computational resources

Agents and Organizations

- Agents $\Longleftrightarrow$ Autonomy
  - Agents are motivated by their own objectives, beliefs...
  - $\Rightarrow$ may take up organizational role if it serves their purposes
- Organization $\Longleftrightarrow$ Regulation
  - Organizations (too) have their own purpose
  - Exist independently of the agents populating it
- Fundamental tension

Requirements Engineering

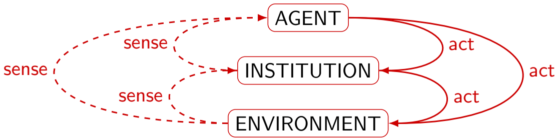
- … or workflows for agents
- System evolution: closed $\rightarrow$ semi-open $\rightarrow$ open
- (Good) governance: evaluating risks and monitoring compliance
- How can component actions be regulated without compromising their integrity or revealing information?
  - Contracts: service level agreements
  - Monitoring/Auditing framework
  - Roles, powers, permissions, authentication
- Virtual $\leftrightarrow$ physical world interaction: *counts-as*
- Institutions are a non-invasive way to constrain software components in open architectures

Need for organization

- Do agents need organizations?
- Do agents need to know/reason about the organization?
- Do MAS need organizations?
  - Interaction in MAS cannot be based on communication alone
  - MAS engineering requires high level agent-independent abstractions
  - Explicit social concepts, defining the society in which agents participate

What is an Institution?

- A set of rules:
  - capable of describing correct
  - and incorrect action,
  - obligations acquired through correct action
  - and sanctions levied for incorrect action
  - while maintaining a record through its internal state.
- An institution is a set of rules that interprets some but not necessarily all of an agent's actions as correct or incorrect within that context: the norm-regulated agent.

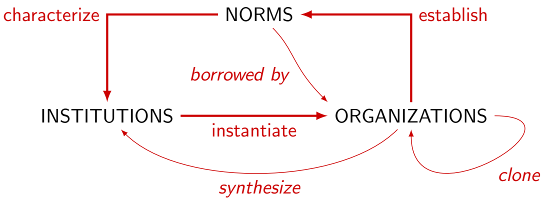


Institutions

- Institutions facilitate and enforce the normative character of organizations
- Describe exchange mechanisms
- Specify coordination structures
- Determine interaction and communication forms within the organization
- Connect organizational and individual perspectives
- Make explicit the social norms governing behaviour, external to the agents

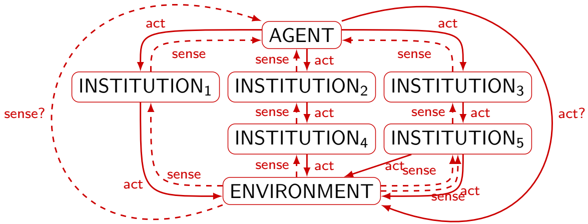
Institutions *and Norms*

- Assertion: an institution *is* its norms
- What is a norm? Informal or formal constraint on action
- **Definition:** a principle of right action binding upon the members of a group and serving to guide, control, or regulate proper and acceptable behavior [Merriam-Webster dictionary]



What is a *Multi*-institution?

- But there is not just one institution
- An agent acts in several institutions, concurrently, even simultaneously
- An institution has restricted competence; aggregation provides complex legal and/or social contexts
- Thus: a multi-institution is a combination of institutions providing the complete interpretation of an agent's actions.



Multiple Institutions

- A single institution can capture the full normative behaviour, but a monolithic structure may be undesirable:
  - Single institutions with a limited range of interaction can be analysed and re-used more easily — institution libraries
  - Institutions are situated in a social and legal framework with whose norms they must interoperate, so institutional workflows are unavoidable
- Institutional composition is a different process in which a *single* internally consistent institution is synthesized from several institutional specifications.
- A multi-institution is a workflow of several connected institutions, each with their own identity and probably with conflicting norms.

Organizations

- Organizations are concrete collections of individuals working together for some purpose
- Game theory and institution theory help the mechanism designer build organizations to achieve desired outomes
- Game theory and institution theory help the agent designer build strategies to achieve desired outcomes
- Auctions and contract net (etc.) are mechanisms that enable organizations, while themselves being organizational structures.
- Thus we assert the following:
  - An organization is an instance of an institution
  - An institution is a set of norms
  - A norm serves to guide or regulate agent behaviour

Institutions

Human institutions have a long history: origins in society or laws made by society.

- So common that we operate unaware of them
- Furthermore we play (or combine a set of) rôles
- Institutions offer a basis for trust and security:
  - decrease uncertainty
  - reduce conflict of meaning
  - create expectations of outcome
  - simplify the decision process

Categorization of norms

- Abstract: high level, expressing what *ought* (not) to be
- Concrete: middle level, capturing actions or changes of state that depend on particular conditions
- Procedural, also called protocols: sequences of actions that (typically) if followed ensure (higher level) norm-compliance
- The technical challenge is how to prove that a specification at one level is consistent with one at another level

Institutions: examples

Institutions are everywhere—formal and informal, legal and social—you just have to know for what to look...

- Conversation, negotiation, argument
- Lecture, seminar, problem class
- Shop: served *vs.* self-service
- Business: sole-trader, partnership, Ltd. company, plc, cooperative, charity, non-profit organization, ...
- Market: stock market, energy trading, brokering (stocks, flights), auction

Norms: examples

- The distribution of radio frequencies between bidders should take into account their established interests
- If estimated fish stock is x tonnes and viable mass is y tonnes $\Rightarrow$ catch should be $< (x - y)/\#$ fishermen
- A front-office trader should not carry out settlements
- Polluter pays? Kyoto protocol, carbon credits
- Who should have this liver?
- Don't led market players design your mechanism

Turning requirements into software

Perhaps building institutions is just software engineering—compare with safety-critical systems:

> *In contrast to software engineering, the purpose of institution specification is not to capture the requirements, but simply to separate "good" or socially acceptable behaviour from "bad" or unacceptable behaviour and to specify appropriate sanctions for violations.*

- Following conventional software engineering practices, the two approaches can be characterized as:
  - bottom-up: typified by the FishMarket
  - top-down: norm-driven approach

  But both are evolutionary in style and rely on specialized tools

The FishMarket Approach

Origins in Noriega's thesis [Pablo Noriega 1997 and developed further by Rodríguez [Rodriguez-Aguilar 2001].

Methodology (and tools, ie. Islander) encourage the identification of the following components:

- scenes are where agents interact, following *conversation protocols* and uttering *illocutions*.
- transitions labelled with constraints on agents moving between scenes
- performative structure defines the connectivity of scenes/transitions
- rôles that act like types to determine the set of permitted actions/illocutions of an agent

Performative structure

- Describes the overall structure of the organization
- A directed graph of scenes—containing conversation protocols (see later)
- There may be multiple instance of some scenes—e.g. one for each auction, for each organ/tissue exchange
- Additionally different node-end connectors are used to indicate logical relationships between arriving agents
- A *bottom-up* approach, where we define interactions in terms of protocols and then link them together to form an institution (the performative structure).

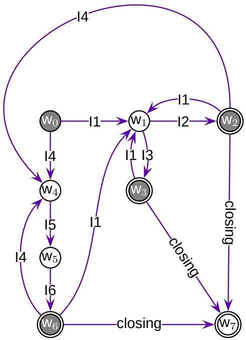
Rôles

- Rôle hierarchy: to encode power relationships
- Rôle (in)compatibility: to encode whether an agent may play certain combinations of roles simultaneously, concurrently or never:
  - Agent cannot play two rôles ever, e.g. dealing and settlement
  - Agent cannot play two rôles at the same time, e.g. buyer and seller

The Blanes Fishmarket

Case study: the fish auction in the town of Blanes on the Costa Brava. For a more detailed discussion see [Rodríguez et al. 1997].

A conversation: Buyers' settlement



- A grey filled node denotes an access node
- A double perimeter denotes an exit node
- A grey filled double perimeter denotes an access/exit node

and the illocutions are:

- I1 (request (?x b) (?y bac) (update-credit ?value))
- I2 (accept (!y bac) (!x b) (accept "update-credit"))
- I3 (deny (!y bac) (!x b) (deny ?deny-code))
- I4 (request (?x b) (?y bac) statement)
- I5 (inform (!y bac) (!x b) (statement (?g1 ?p1) ... (?gn ?pn)))
- I6 (pay (!x b) (!y bac) (payment ?value))

What the agents may say

1 (request (?x hf|hc|tb) (?y rrm) (admission ?id_agent ?role ?hospital_certificate))

2 (deny (!y rrm) (!x hf|hc|tb) (deny ?deny_reason))

3a (accept (!y rrm) (!x hc) (accept_hc))

3b (accept (!y rrm) (!x tb) (accept_tb))

3c (accept (!y rrm) (!x hf) (accept_hf))

3d (accept (!y rrm) (!x ho) (accept_ho))

4a (inform (?x hf) (?y rrm) (petition_tissue ?id_hospital ?urgency_level ?time_to_deliver ?piece_type (?piece_parameters) (?info_recipient)))

4b (inform (?x hf) (?y rrm) (petition_organ ?id_hospital ?time_for_availability ?piece_type (?piece_parameters) (?info_donor)))

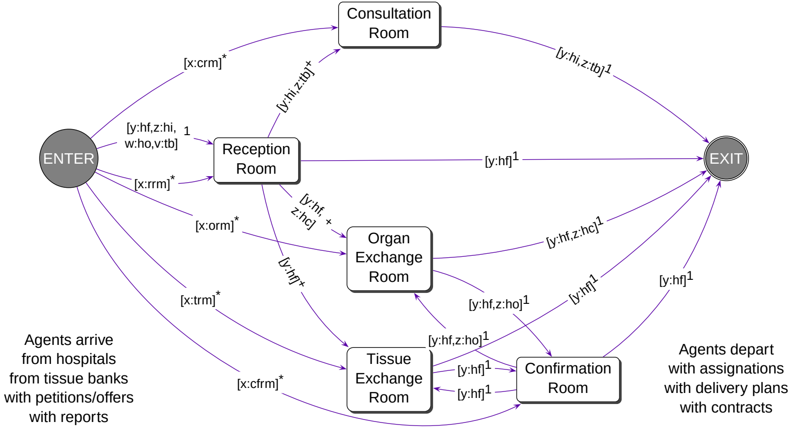
5 (inform (!y rrm) (!x hf) (petition_state ?id_petition ok|faulty))

6 (inform (?y rrm) (?x hf) (init_exchange ?piece_type ?id_exchange_room))

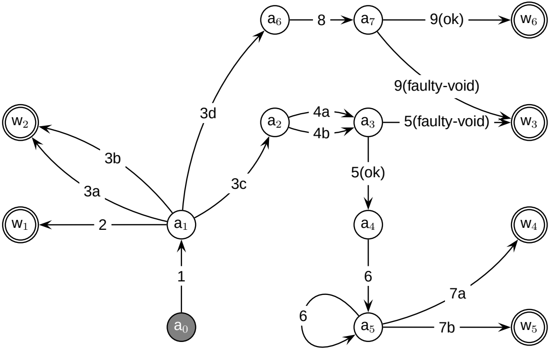
7a (request (?x hf) (?y rrm) (tissue_exchange_entrance_request !id_exchange_room))

Tissue and Organ Distribution

Case study: Organs: supply-driven. Tissues: demand-driven. For a detailed discussion see [Vázquez-Salceda et al. 2003].



Norms+rules+procedures=policies



InstAL: A Norm-driven approach I

- InstAL: A top-down approach to institutional modelling views an institution as [Cliffe, De Vos, and Padget 2006a; Cliffe, De Vos, and Padget 2006b]:
  - A set of *institutional states* that evolve
  - in response to *institutional events*.
  - where an institutional state is a set of *institutional facts*
- These are the *observables* identified earlier

A Norm-driven approach II

- How are institutional facts created?
  - Searle [John R. Searle 1995] identifies two kinds of facts
    - Brute facts that are observable in the physical world
    - and institutional facts that are neither observable, nor have any meaning outside their institution
  - Institutional facts are created by an action in the physical world that counts as taking that action in the institutional world.
  - Thus the observation of an agent action can lead to the creation of an institutional fact within the institution in which the agent is participating.

Social States

Several types of institutional facts are considered:

- **Permission:** An agent's Ability to carry out some action without sanction.
- **Obligation:** Facts stating that an agent is obliged to have done some action before some deadline.
- **Institutional Power:** (after Jones & Sergot) institutional facts describe an agent's capacity to affect the social state by performing *meaningful* institutional actions.
- **Domain Facts:** Those relating internally to the institution in question. (i.e. marina Ows X)

Events

- Account for (possible) changes in state
- May be:
  - Domain Events (exogenous): observed from the environment.
  - Institutionally generated (internal): generated by the institution
- Events may generate other events: Conventional generation.

Conventional Generation

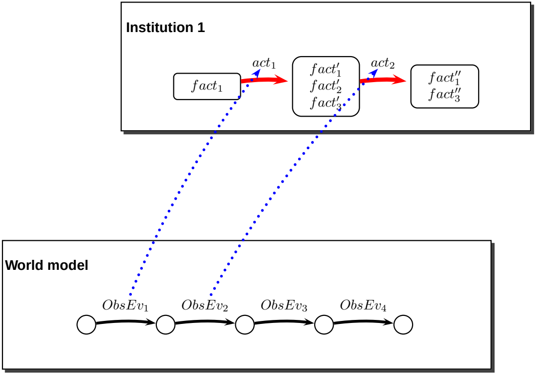
Origins in theory of action (Goldman, Searle, Jones & Sergot)

- "Doing X [in environment A] counts as doing Y [in environment B] iff Z"
- Allows us to abstract institutional actions from real world ones, i.e.:
  - "Saying 'aye' in an auction counts as an offer to buy some goods at the current price"
  - "Clicking 'buy it now' counts as an offer to buy some goods at a given price on amazon"
- Generation is assumed to be atomic (i.e. generated events occur concurrently with events which generate them)

Regulation

- Not all sequences of action are desirable
- We specify regulatory rules to identify "bad" paths of events
- Two regulatory mechanisms are considered:
  - **Obligation:** "You should do X before Y happens"
  - **Permission:** "You should not do X"
- **Violations:** When the above rules are broken *violation events* are generated for:
  - The failure to perform an action before a deadline.
  - Performing an action without permission.

Specification Model



Representing State

- capture (in)formal contextualized expectations of behaviour
- Three kinds of normative facts:
  1. `perm(doX(agent1,...))`
  2. `obl(doX(agent1,...),beforeY,otherwiseZ)` or `obl(achieveX(agent1,...),beforeY,otherwiseZ)`
  3. `pow(doX,(agent1,...))`

  where
  - `achieveX` is a condition over normative facts
  - `beforeY` is either an event or a condition and
  - `otherwiseZ` is a violation event
- Domain facts depending on model

Actions change the (normative) world

- counts-as: $\mathcal{G}$: external action $\xrightarrow[\text{if(conditions)}]{\text{generates}}$ normative action
- normative facts represented by fluents
  1. fluent $\Rightarrow$ true if present, false otherwise
  2. inertial fluent:

     $\mathcal{C}^{\uparrow}$: action $\xrightarrow[\text{if(conditions)}]{\text{initiates}}$ fluent

     $\mathcal{C}^{\downarrow}$: action $\xrightarrow[\text{if(conditions)}]{\text{terminates}}$ ~~fluent~~

     good for facts true for a period with start and finish actions
  3. non-inertial fluent: $\mathcal{C}^{\mathcal{N}}$: $\xrightarrow[\text{if(conditions)}]{}$ fluent

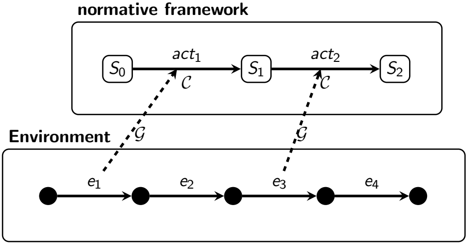
     good for facts expressed as a combination of inertial + non-inertial fluents

Making it work

- mathematical model:
  sets + relations $(\mathcal{G}, \mathcal{C}) \rightsquigarrow$ labelled transition system

$$\Delta \xrightarrow{e_1} S_1 \xrightarrow{e_2} S_2 \xrightarrow{e_3} \ldots$$

- conceptual model:



InstAL rules

1. `type AType;`: AType specifies the variables we use, e.g. Article
2. `fluent/nonintertial` f`;`: definition of (non inertial) fluent f
3. x `generates` y `[ if` z `];`: x an event (action), y one or more events; conditional z
4. x `initiates` y `[ if` z `];`: x an event, y one or more fluents; add to model state conditional on z
5. x `terminates` y `[ if` z `];`: as initiates, but remove fluents conditional on z
6. x `when` y: x non-inertial fluent, y a condition over the model state
7. `initially` x `[ if` y `];` : x one or more fluents in the initial model state, conditional on y

Implementation

- computational model:

  InstAL $\xrightarrow{\text{translator}}$ AnsProlog $\xrightarrow{\text{clingo}}$ answer sets $\xrightarrow{\text{visualisers}}$

- python front-end
- compiler: InstAL to Answer Set Programming
- python API to Clingo (answer set solver, C++)
- answer sets delivered in JSON
- visualization tools generate images from traces

Example

```
Instal

% the sale proceeds if sale is empowered and both
    participants have the correct assets
sale(Seller,Buyer) generates intSale(Seller,Buyer) if
    hasGood(Seller), hasMoney(Buyer);

% sale split between buyer and seller responsibilities
intSale(Seller,Buyer) generates transferReq(Seller),
    paymentReq(Buyer);
sale(Seller,Buyer) initiates contract(Seller,Buyer);

ASP
occurred(intSale(Seller,Buyer),I) :- occurred(sale(Seller,
    Buyer),I),holdsat(hasGood(Seller),I),holdsat(hasMoney(
    Buyer),I),holdsat(pow(legal,intSale(Seller,Buyer)),I),
    instant(I), agent(Seller),agent(Buyer).

occured(transferReq(Seller),I) :- occured(intSale(Seller,
    Buyer),I),holdsat(pow(legal,transferReq(Seller)),I),
    instant(I),agent(Seller),agent(Buyer).
occured(paymentReq(buyer),I) :-        occured(intSale(
    Seller,Buyer),I),holdsat(pow(legal,paymentReq(Buyer)),I)
    ,instant(I),agent(Seller),agent(Buyer).


initiated(contract(Seller,Buyer),I) :-
occured(sale(Seller,Buyer),I),holdsat(live(legal),I),instant
    (I),agent(Seller),agent(Buyer).
```

Beyond the Basics

- Multi-Institutions
- Conflict detection - debugging
- Design-time vs runtime
- Norm Change/Emergence
- Value $< - >$ norms

Collaborators

- Tina Balke
- Owen Cliffe
- Domenico Corapi
- Sabrina Kirrane
- Tingting Li
- Jack McKinlay
- Alessandra Mileo
- Fahid Mohammed
- Andreassa Morris Martin
- Julian Padget
- Oliver Ray
- Alessandra Russo
- Ken Satoh

References I

- Cliffe, Owen, Marina De Vos, and Julian Padget (May 2006a). "Answer Set Programming for Representing and Reasoning about Virtual Institutions". In: *Computational Logic for Multi-Agents (CLIMA VII)*. Hakodate, Japan.
- — (May 2006b). "Specifying and Reasoning about Multiple Institutions". In: *Coordination, Organization, Institutions and Norms in Agent Systems (COIN'06)*. Hakodate, Japan.
- John R. Searle (1995). *The Construction of Social Reality*. Allen Lane, The Penguin Press.
- Pablo Noriega (1997). "Agent mediated auctions: The Fishmarket Metaphor". PhD thesis. Universitat Autonoma de Barcelona.

References II

- Rodríguez, Joan-Antoní et al. (Apr. 1997). "FM96.5 A Java-based Electronic Auction House". In: *Proceedings of 2nd Conference on Practical Applications of Intelligent Agents and MultiAgent Technology (PAAM'97)*. ISBN 0-9525554-6-8. London, UK, pp. 207–224. URL: a%20href=http://www.iiia.csic.es/Projects/fishmarket/PAAM97.ps.gz%3Ehttp://www.iiia.csic.es/Projects/fishmarket/PAAM97.ps.gz%3C/a.
- Rodriguez-Aguilar, Juan A. (2001). "On the Design and Construction of Agent-mediated Institutions". PhD thesis. Universitat Autonoma de Barcelona.

References III

Vázquez-Salceda, Javier et al. (2003). "Formalizing an Electronic Institution for the distribution of Human Tissues". In: *Artificial Intelligence in Medicine* 27.3. ISSN: 0933-3657, pp. 233–258. URL: a%20href=http://www.cs.bath.ac.uk/~jap/Papers/aime.ps%3Ehttp://www.cs.bath.ac.uk/~jap/Papers/aime.ps%3C/a.