On Autonomy
Definitions & Acceptations

Autonomous Systems
Sistemi Autonomi

Andrea Omicini
andrea.omicini@unibo.it

Dipartimento di Informatica – Scienza e Ingegneria (DISI)
Alma Mater Studiorum – Università di Bologna

Academic Year 2013/2014

Outline

Oxford Dictionary of English (2nd Edition revised 2005)

Etimology

Early 17th cent.: from Greek *autonomia*, from *autonomos* ‘having its own laws’, from *autos* ‘self’ + *nomos* ‘law’.

Dictionary

autonomy

- the right or condition of self-government
- a self-governing country or region
- freedom from external control or influence; independence.
- (in Kantian moral philosophy) the capacity of an agent to act in accordance with objective morality rather than under the influence of desires

Oxford Thesaurus of English (2nd Edition revised 2008)

Thesaurus

autonomy

- self-government, independence, self-rule, home rule, sovereignty, self-determination, freedom, autarchy;
- self-sufficiency, individualism.

Merriam-Webster I

Dictionary

autonomy

1. the quality or state of being self-governing; especially: the right of self-government
2. self-directing freedom and especially moral independence
3. a self-governing state

Synonyms accord, free will, choice, self-determination, volition, will

Antonyms dependence (also dependance), heteronomy, subjection, unfreedom

Merriam-Webster II

Thesaurus

autonomy

1. the act or power of making one's own choices or decisions: accord, free will, choice, self-determination, volition, will
2. the state of being free from the control or power of another: freedom, independence, independency, liberty, self-determination, self-governance, self-government, sovereignty

Outline

Internet Encyclopedia of Philosophy I

Many acceptations of autonomy

general an individual's capacity for self-determination or self-governance

folk inchoate desire for freedom in some area of one's life

personal the capacity to decide for oneself and pursue a course of action in one's life

moral the capacity to deliberate and to give oneself the moral law, rather than merely heeding the injunctions of others

political the property of having one's decisions respected, honored, and heeded within a political context

Internet Encyclopedia of Philosophy II

Individual autonomy

- after Kant, autonomy is an essential trait of the *individual*, and strictly related with its morality
- then, with the relation between its inner self and its individual actions
- that is, mind and behaviour

Outline

Unmanned Systems Integrated Roadmap FY 2011-2036 I

Automatic vs. autonomous

Automatic systems are fully pre-programmed and act repeatedly and independently of external influence or control. An automatic system can be described as self-steering or self-regulating and is able to follow an externally given path while compensating for small deviations caused by external disturbances. However, the automatic system is not able to define the path according to some given goal or to choose the goal dictating its path.

Autonomous systems are self-directed toward a *goal* in that they do not require outside control, but rather are governed by laws and strategies that direct their behavior. Initially, these control algorithms are created and tested by teams of human operators and software developers. However, if *machine learning* is utilized, autonomous systems can develop modified strategies for themselves by which they select their behavior. An autonomous system is self-directed by choosing the behavior it follows to reach a human-directed goal.

Unmanned Systems Integrated Roadmap FY 2011-2036 II

Four Levels of Autonomy for Unmanned Systems *(Guy Edwards)*

Various levels of autonomy in any system guide how much and how often humans need to interact or intervene with the autonomous system:

- Human Operated
- Human Delegated
- Human Supervised
- Fully Autonomous injunctions of others

Unmanned Systems Integrated Roadmap FY 2011-2036 III

Level	Name	Description
1	Human Operated	A human operator makes all decisions. The system has no autonomous control of its environment although it may have information-only responses to sensed data.
2	Human Delegated	The vehicle can perform many functions independently of human control when delegated to do so. This level encompasses automatic controls, engine controls, and other low-level automation that must be activated or deactivated by human input and must act in mutual exclusion of human operation.
3	Human Supervised	The system can perform a wide variety of activities when given top-level permissions or direction by a human. Both the human and the system can initiate behaviors based on sensed data, but the system can do so only if within the scope of its currently directed tasks.
4	Fully Autonomous	The system receives goals from humans and translates them into tasks to be performed without human interaction. A human could still enter the loop in an emergency or change the goals, although in practice there may be significant time delays before human intervention occurs.

Unmanned Systems Integrated Roadmap FY 2011-2036 IV

Autonomy & Unpredictability

- The special feature of an autonomous system is its ability to be goal-directed in *unpredictable situations*.
- This ability is a significant improvement in capability compared to the capabilities of automatic systems.
- An autonomous system is able to *make a decision* based on a set of rules and/or limitations.
- It is able to determine *what information is important* in making a decision.
- It is capable of a *higher level of performance* compared to the performance of a system operating in a predetermined manner.

Outline

Agents & Goals [Castelfranchi, 1995] I

Goal-autonomous agent

A *goal-autonomous agent* is an agent endowed with its own *goals*

Agents & Goals [Castelfranchi, 1995] II

Fully socially autonomous agent

An agent is *fully socially autonomous* if

- it has its own goals: *endogenous*, not derived from other agents' will
- it is able to *make decisions* concerning *multiple conflicting goals* (being them its own goals or also goals adopted from outside)
- it adopts *goals from outside*, from other agents; it is liable to *influencing*
- it adopts other agents' goals as a consequence of a *choice* among them and other goals
- it adopts other agents' goals only if it sees the adoption as a way of enabling itself to achieve some of its own goals (i.e., the autonomous agent is a *self-interested* agent)
    - it is not possible to directly modify the agent's goals from outside: any modification of its goals must be achieved by modifying its *beliefs*
    - thus, the control over beliefs becomes a filter, an additional control over the adoption of goals
- it is impossible to change automatically the beliefs of an agent
    - the adoption of a belief is a special "decision" that the agent takes on the basis of many criteria
    - this protects its *cognitive autonomy*

Outline

Evolution of Programming Languages: The Picture

- [Odell, 2002]

	Monolithic Programming	Modular Programming	Object-Oriented Programming	Agent Programming
Unit Behavior	Nonmodular	Modular	Modular	Modular
Unit State	External	External	Internal	Internal
Unit Invocation	External	External (CALLed)	External (message)	Internal (rules, goals)

Evolution of Programming Languages: Dimensions

Historical evolution

- Monolithic programming
- Modular programming
- Object-oriented programming
- Agent programming

Degree of modularity & encapsulation

- Unit behaviour
- Unit state
- Unit invocation

Monolithic Programming

- The basic unit of software is the whole program
- Programmer has full control
- Program's state is responsibility of the programmer
- Program invocation determined by system's operator
- Behaviour could not be invoked as a reusable unit under different circumstances
  - modularity does not apply to unit behaviour

Modular Programming

- The basic unit of software are structured loops / subroutines / procedures / ...
  - this is the era of procedures as the primary unit of decomposition
- Small units of code could actually be reused under a variety of situations
  - modularity applies to subroutine's code
- Program's state is determined by externally supplied parameters
- Program invocation determined by CALL statements and the likes

Object-Oriented Programming

- The basic unit of software are objects & classes
- Structured units of code could actually be reused under a variety of situations
- Objects have local control over variables manipulated by their own methods
  - variable state is persistent through subsequent invocations
  - object's state is encapsulated
- Object are passive—methods are invoked by external entities
  - modularity does not apply to unit invocation
  - object's control is not encapsulated

Agent-Oriented Programming

- The basic unit of software are *agents*
  - encapsulating everything, in principle
    - by simply following the pattern of the evolution
  - whatever an agent is
    - we do not need to define them now, just to understand their desired features
- Agents could in principle be reused under a variety of situations
- Agents have control over their own state
- Agents are active
  - they cannot be invoked
  - agent's control is encapsulated
- Agents are *autonomous* entities

Outline

Autonomy as the Foundation of the Definition of Agent

Lex Parsimoniae: Autonomy

- Autonomy as the only fundamental and defining feature of agents
- Let us see whether other typical agent features follow / descend from this somehow

Computational Autonomy

- Agents are autonomous as they *encapsulate* (the thread of) *control*
- Control does not pass through agent boundaries
  - only data (knowledge, information) crosses agent boundaries
- Agents have no interface, cannot be controlled, nor can they be invoked
- Looking at agents, MAS can be conceived as an aggregation of multiple distinct *loci* of control interacting with each other by exchanging information

(Autonomous) Agents (Pro-)Act

Action as the essence of agency

- The etimology of the word *agent* is from the Latin agens
- So, agent means "the one who acts"
- Any coherent notion of agency should naturally come equipped with a model for agent actions

Autonomous agents are pro-active

- Agents are literally active
- Autonomous agents encapsulate control, and the rule to govern it

→ Autonomous agents are pro-active by definition

  - where pro-activity means "making something happen", rather than waiting for something to happen

Agents are Situated

The model of action depends on the context

- Any “ground” model of action is strictly coupled with the context where the action takes place
- An agent comes with its own model of action
- Any agent is then strictly coupled with the environment where it lives and (inter)acts
- Agents are in this sense are intrinsically *situated*

Agents are Reactive I

Situatedness and reactivity come hand in hand

- Any model of action is strictly coupled with the context where the action takes place
- Any action model requires an adequate *representation* of the world
- Any *effective* representation of the world requires a *suitable* balance between environment *perception* and representation

→ Any effective action model requires a suitable balance between environment perception and representation

  - however, any non-trivial action model requires some form of perception of the environment—so as to check action pre-conditions, or to verify the effects of actions on the environment

- Agents in this sense are supposedly *reactive* to change

Agents are Reactive II

Reactivity as a (deliberate) reduction of proactivity

- An autonomous agent could be built / choose to merely react to external events
- It may just wait for something to happen, either as a permanent attitude, or as a temporary opportunistic choice
- In this sense, autonomous agents may also be reactive

Reactivity to change

- Reactivity to (environment) change is a different notion
- This mainly comes from early AI failures, and from robotics
- It stems from agency, rather than from autonomy
- However, this issue will be even clearer when facing the issue of artifacts and environment design

(Autonomous) Agents Change the World

Action, change & environment

- Whatever the model, any model for action brings along the notion of *change*
  - an agent acts to change something around in the MAS
- Two admissible targets for change by agent action

  agent an agent could act to change the state of another agent
  - since agents are autonomous, and only data flow among them, the only way another agent can change their state is by providing them with some information
  - change to other agents essentially involves *communication actions*

  environment an agent could act to change the state of the environment
  - change to the environment requires *pragmatical actions*
  - which could be either physical or virtual depending on the nature of the environment

Autonomous Agents are Social

From autonomy to society

- From a philosophical viewpoint, autonomy only makes sense when an individual is immersed in a society
  - autonomy does not make sense for an individual in isolation
  - no individual alone could be properly said to be autonomous
- This also straightforwardly explain why any program in any sequential programming language is not an autonomous agent *per se* [Graesser, 1996, Odell, 2002]

Autonomous agents live in a MAS

- Single-agent systems do not exist in principle
- Autonomous agents live and interact within agent societies & MAS
- Roughly speaking, MAS are the only "legitimate containers" of autonomous agents

Autonomous Agents are Interactive

Interactivity follows, too

- Since agents are subsystems of a MAS, they interact within the global system
  - by essence of systems in general, rather than of MAS
- Since agents are autonomous, only data (knowledge, information) crosses agent boundaries
- Information & knowledge is exchanged between agents
  - leading to more complex patterns than message passing between objects

Autonomous Agents Do not *Need* a Goal

Agents govern MAS computation

- By encapsulating control, agents are the main forces governing and pushing computation, and determining behaviour in a MAS
- Along with control, agent should then encapsulate the *criterion* for regulating the thread(s) of control

Autonomy as self-regulation

- The term "autonomy", at its very roots, means self-government, self-regulation, self-determination
  - "internal unit invocation" [Odell, 2002]
- This does *not* imply in any way that agents *needs* to have a goal, or a task, to be such—to be an agent, then
- However, this *does* imply that autonomy captures the cases of goal-oriented and task-oriented agents
  - where goals and tasks play the role of the criteria for governing control

Agents as Autonomous Components

Definition (Agent)

Agents are *autonomous computational entities*

genus agents are computational entities

differentia agents are autonomous, in that they encapsulate control along with a criterion to govern it

Agents are *autonomous*

- From autonomy, many other features stem
  - autonomous agents *are* interactive, social, proactive, and situated;
  - they *might* have goals or tasks, or be reactive, intelligent, mobile
  - they live within MAS, and *interact* with other agents through *communication actions*, and with the environment with *pragmatical actions*

Outline

Complex Systems

. . . by a complex system I mean one made up of a large number of parts that interact in a non simple way [Simon, 1962]

Which "parts" for complex systems?

- is autonomy of "parts" a necessary precondition?
- is it also sufficient?

Which kind of systems are we looking for?

- what is *autonomy* for a system as a whole?
- where could we find significant examples?

Nature-inspired Models

Complex natural systems

- such as physical, chemical, biochemical, biological, social systems
- natural system exhibit *features*
  - such as distribution, opennes, situation, fault tolerance, robustness, adaptiveness, . . .
- which we would like to understand, capture, then bring to *computational* systems

Nature-Inspired Computing (NIC)

- For instance, NIC [Liu and Tsui, 2006] summarises decades of research activities
- putting emphasis on
  - autonomy of components
  - self-organisation of systems

Autonomy & Interaction I

Self-organisation

- Autonomy of *systems* requires essentially the same features that self-organising systems exhibit
- ... such as opennes, situation, fault tolerance, robustness, adaptiveness, ...

(say it again)

> *... by a complex system I mean one made up of a large number of parts that interact in a non simple way [Simon, 1962]*

Autonomy & Interaction

- "parts" should be *autonomous*
- *interaction* is essential as well

MAS & Complexity I

Autonomy & Interaction

- Multi-Agent Systems (MAS) are built around autonomous components
- How can MAS deal with self-organisation, properly handling interaction among components?

The observation of self-organising termite societies led to the following observation:

> *The coordination of tasks and the regulation of constructions are not directly dependent from the workers, but from constructions themselves [Grassé, 1959]*

MAS & Complexity II

Coordination as the key issue

- many well-known examples of natural systems – and, more generally, of complex systems – seemingly rely on simple yet powerful coordination mechanisms for their key features—such as self-organisation
- it makes sense to focus on coordination models as the core of complex MAS
- . . . since they are conceived to deal with the complexity of interaction [Omicini, 2013]

Basic Issues of Nature-inspired Coordination I

Environment

- environment is essential in nature-inspired coordination
  - it works as a mediator for component interaction — through which the components of a distributed system can communicate and coordinate indirectly
  - it is active — featuring autonomous dynamics, and affecting component coordination
  - it has a structure — requiring a notion of *locality*, and allowing components of any sort to *move* through a topology

Basic Issues of Nature-inspired Coordination II

Stochastic behaviour

- complex systems typically require probabilistic models
  - *don't know* / *don't care* non-deterministic mechanisms are not expressive enough to capture all the properties of complex systems such as biochemical and social systems
  - probabilistic mechanisms are required to fully capture the dynamics of coordination in nature-inspired systems
  - coordination models should feature (possibly simple yet) expressive mechanisms to provide coordinated systems with stochastic behaviours

Coordination Middleware I

Middleware

- Coordination middleware to build complex software environment
- Coordination abstractions to embed situatedness and stochastic behaviours
- Coordination abstractions to embed social rules–such as *norms*, and *laws*

Coordination Middleware II

Nature-inspired middleware

- starting from early chemical and stigmergic approaches, nature-inspired models of coordination evolved to become the *core* of *complex distributed systems*—such as pervasive, knowledge-intensive, intelligent, and self-* systems
- this particularly holds for tuple-based coordination models [Omicini and Viroli, 2011]
- tuple-based middleware as a perspective technology for complex autonomous systems

Outline

Conclusion I

Autonomy

- Many different notions of autonomy
- which needs to be discussed, related, and considered altogether in a coherent conceptual and technological framework

Agents

- Components should be autonomous in autonomous systems
- Agent models and technologies are the most likely answers to the issues of autonomy of components

Interaction

- Autonomous systems require self-organisation patterns
- Interaction is another essential dimension of self-organisation

Conclusion II

Coordination

- Coordination models and technologies for governing interaction
- . . . including social norms and laws

Nature-inspired coordination

- Nature-inspired coordination models for autonomous systems
- Tuple-based coordination models are the most likely candidates to face the issues of autonomy of complex systems

Bibliography I

Castelfranchi, C. (1995).
Guarantees for autonomy in cognitive agent architecture.
In Wooldridge, M. J. and Jennings, N. R., editors, *Intelligent Agents*, volume 890 of *Lecture Notes in Computer Science*, pages 56–70. Springer Berlin Heidelberg.
ECAI-94 Workshop on Agent Theories, Architectures, and Languages (ATAL) Amsterdam, The Netherlands 8–9 August 1994. Proceedings.

Bibliography II

Graesser, S. F. A. (1996).
Is it an agent, or just a program?: A taxonomy for autonomous agents.
In Müller, J. P., Wooldridge, M. J., and Jennings, N. R., editors, *Intelligent Agents III Agent Theories, Architectures, and Languages: ECAI'96 Workshop (ATAL) Budapest, Hungary, August 12–13, 1996 Proceedings*, volume 1193 of *Lecture Notes In Computer Science*, pages 21–35. Springer.

Grassé, P.-P. (1959).
La reconstruction du nid et les coordinations interindividuelles chez Bellicositermes natalensis et Cubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs.
Insectes Sociaux, 6(1):41–80.

Bibliography III

Liu, J. and Tsui, K. C. (2006).
Toward nature-inspired computing.
Communications of the ACM, 49(10):59–64.

Odell, J. (2002).
Objects and agents compared.
Journal of Object Technologies, 1(1):41–53.

Omicini, A. (2013).
Nature-inspired coordination models: Current status, future trends.
ISRN Software Engineering, 2013.
Article ID 384903, Review Article.

Bibliography IV

Omicini, A. and Viroli, M. (2011).
Coordination models and languages: From parallel computing to self-organisation.
The Knowledge Engineering Review, 26(1):53–59.
Special Issue 01 (25th Anniversary Issue).

Simon, H. A. (1962).
The architecture of complexity.
Proceedings of the American Philosophical Society, 106(6):467–482.

On Autonomy
Definitions & Acceptations

Autonomous Systems
Sistemi Autonomi

Andrea Omicini
andrea.omicini@unibo.it

Dipartimento di Informatica – Scienza e Ingegneria (DISI)
Alma Mater Studiorum – Università di Bologna

Academic Year 2013/2014