Nature-Inspired Coordination & Self-Organisation

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 2015/2016

Outline

1. Nature-inspired Coordination
2. Examples
3. Tuples
4. Trends

Nature-inspired Models for SOS

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 [LT06] summarises decades of research activities, putting emphasis on
    - autonomy of components
    - self-organisation of systems

Multi-Agent Systems for SOS

MAS as complex systems [OZ04]

- Agents as sources of complexity

  Autonomy Unpredictable behaviour

  Sociality Non-compositional behaviours

  Situatedness Unpredictable interaction with the environment

- Multi-Agent Systems (MAS) as sources of complexity
  - Multiplicity of interacting components
  - Global vs. local structure and behaviour—macro vs. micro level

MAS for complex systems [ZO04]

MAS as tools for

- Modelling complex systems
- Engineering complex system

Interaction & Coordination?

Interaction

- most of the complexity of complex computational systems – MAS included – comes from interaction [ORV06]
- along with an essential part of their expressive power [Weg97]

Coordination

- since coordination is essentially the science of managing the space of interaction [Weg97]
- coordination models and languages [Cia96] provide abstractions and technologies for the engineering of complex computational systems [COZ00]

Nature-inspired Coordination for MAS

Coordination issues in natural systems

- coordination issues did not first emerge in computational systems
- [Gra59] noted that in termite societies *"The coordination of tasks and the regulation of constructions are not directly dependent from the workers, but from constructions themselves."*

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 nature-inspired coordination models as the core of complex nature-inspired MAS

Outline

Outline

Stigmergy I

Stigmergy in insect societies

- nature-inspired models of coordination are grounded in studies on the behaviour of social insects, like ants or termites
- [Gra59] introduced the notion of stigmergy as the fundamental coordination mechanism in termite societies
- in ant colonies, pheromones act as environment markers for specific social activities, and drive both the *individual* and the *social* behaviour of ants

Stigmergy II

Stigmergy in computational systems

- nowadays, stigmergy generally refers to a set of nature-inspired coordination mechanisms mediated by the *environment*
- *digital pheromones* [PBS02] and other *signs* made and sensed in a shared environment [Par06] can be exploited for the engineering of adaptive and self-organising MAS

Chemical Coordination

Chemical reactions as (natural) coordination laws

- inspiration comes from the idea that complex physical phenomena are driven by the (relatively) simple chemical reactions
- coordinating the behaviours of a huge amount of agents, as well as the global system evolution

Chemical reactions as (computational) coordination laws

- Gamma [BLM90] is a *chemistry-inspired coordination* model—as for the CHAM (chemical abstract machine) model [Ber92]
- coordination in Gamma is conceived as the evolution of a space governed by chemical-like rules, globally working as a rewriting system [BFLM01]

Outline

Field-based Coordination

Computational fields as coordination laws

- field-based coordination models like Co-fields [MZ06] are inspired by the way masses and particles move and self-organise according to gravitational/electromagnetic fields
- there, computational force fields – generated either by the mobile agents or by the pervasive coordination infrastructure – propagate across the environment, and drive the actions and motion of the agent themselves

(Bio)chemical Coordination

Chemical reactions as coordination laws

- chemical tuple spaces [VCNO10] exploit the chemical metaphor at its full extent—beyond Gamma
- data, devices, and software agents are represented in terms of chemical reactants, and system behaviour is expressed by means of chemical-like laws
- which are actually time-dependent and stochastic
- embedded within the coordination medium
- biochemical tuple spaces [VC09] add *compartments*, *diffusion*, and *stochastic behaviour* of coordination primitives

Outline

Basic Issues of Nature-inspired Coordination I

Environment

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

! nowadays, everybody knows about the essential role of *environment* in a MAS [WOO07]

? *do we also know how to design and engineer MAS environment?*

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

? *do we know how to embed stochastic behaviours in a MAS?*

Outline

The Ancestor

LINDA [Gel85]

- LINDA is the ancestor of all tuple-based coordination models [RCD01]
- in LINDA, agents synchronise, cooperate, compete
  - based on tuples
  - available in the tuple spaces, working as the coordination media
  - by *associatively* accessing, consuming and producing tuples
- the same holds for any tuple-based coordination model

LINDA is *not* a Nature-inspired Model

Warning

LINDA is *not* a Nature-inspired Model

So, *why* LINDA?

Why tuple-based models?

Why Tuple-based Models? I

Expressiveness

- LINDA is sort of a *core* coordination model
- making it easy to face and solve many typical problems of complex distributed systems
- *complex* coordination problems are solved with *few, simple* primitives
- whatever the model used to measure expressiveness of coordination, tuple-based languages are highly-expressive [BGZ98]

Why Tuple-based Models? II

Environment-based coordination

- generative communication [Gel85] requires *permanent* coordination abstractions
- so, the *coordination infrastructure* provides agents with tuple spaces as coordination services
  - *coordination as a service* (CaaS) [VO06]
- they can be interpreted as coordination artefacts shaping computational *environment* [ORV+04]
  - and used with different levels of awareness by both intelligent and "stupid" agents [Omi13a]
- as such, they can be exploited to support environment-based coordination [RVO05]

Why Tuple-based Models? III

Extensibility

- whatever its expressiveness, LINDA was conceived as a coordination model for closed, parallel systems
- so, in fact, some relevant problems of today open, concurrent systems cannot be easily solved with LINDA either in practice or in theory
- as a result, tuple-based models have been extended with new simple yet powerful mechanisms
- generating a plethora of tuple-based coordination models [RCD01]

Why Tuple-based Models? IV

Nature-inspired extensions

- LINDA may *not* be nature-inspired, but many of its extensions *are*
- many of the coordination models depicted before
  - stigmergy [Par06]
  - field-based [MZ04a]
  - chemical [VCNO10] and biochemical [VC09]
- along with many others, such as
  - cognitive stigmergy [ROV+07]
  - pervasive ecosystems [VPMS12]
- are actually nature-inspired tuple-based coordination models

Toward Self-organising Coordination I

Just *some* is not enough

- capturing just *some* of the principles and mechanisms of natural systems does not ensure to capture their *essence*
- for instance, chemical coordination models such as Gamma and CHAM exploit the raw schema of computation as chemical reaction, but are *not* expressive enough to fully reproduce any non-trivial chemical system
- in fact, *e.g.*, even the simplest model for real chemical reactions requires a notion of *reaction rate*
- neither Gamma nor CHAM provide for such a notion, they are not expressive enough to fully match the behaviour of real chemical systems

Toward Self-organising Coordination II

Self-organising coordination [VCO09]

- most of the traditional coordination models feature abstractions enacting coordination laws that are typically *reactive*, (mostly) *deterministic*, and *global* as well
- in complex systems featuring self-* properties, instead, coordination patterns typically appear at the global level by emergence, from *probabilistic*, *time-dependent* coordination laws based on *local* criteria
- in particular, many coordination models either implicitly or explicitly recognise that full expressiveness requires addressing the issues of time dependency and stochasticity

Examples I

StoKlaim

- StoKlaim [DNLKM06] – a *stochastic* extension of the Linda-derived Klaim model for mobile coordination [DNFP98] – adds distribution rates to coordination primitives—thus making it possible the modelling of non-deterministic real-life phenomena such as failure rates and inter-arrival times

SwarmLinda

- SwarmLinda [TM04] enhances Linda implementation with swarm intelligence to achieve features such as scalability, adaptiveness, and fault-tolerance—by modelling tuple templates as ants, featuring probabilistic behaviour when looking for matching tuples in a distributed setting

Examples II

Time-aware ReSpecT

- ReSpecT [OD01] generally addresses *time dependency* by capturing time events and supporting the definition and enforcement of *timed coordination policies* [ORV05]—so, ReSpecT-programmed tuple centres can work as time-dependent abstractions for MAS coordination [ORV07]

Enough?

No.

- in the overall, the above-mentioned models *fail* to capture all the *essential features* of nature-inspired coordination
- this is why many novel research lines stretch existing tuple-based models to achieve the expressive power required to model and build MAS with a complexity comparable to natural systems [OV11]

Outline

Outline

Complexity as a Multi-disciplinary Notion

Complex systems everywhere

- The notion of *complexity* is definitely a *multi-disciplinary* one, ranging from physics to biology, from economics to sociology and organisation sciences
- Systems that are said *complex* are both natural and artificial ones

Natural vs. artificial complex systems

- We *observe* and *model* complex physical systems
- We *design* and *build* complex computational systems

Question

- Which features do *all* complex systems share independently of their nature?

Complexity & Interaction

> *. . . by a complex system I mean one made up of a large number of parts that interact in a non simple way [Sim62]*

Laws of complexity

- Some "laws of complexity" exists that characterise any complex system, *independently* of its specific nature [Kau03]
- The precise source of what all complex systems share is still unknown in essence

Interaction

- We argue that interaction – its *nature*, *structure*, *dynamics* – is the key to understand some fundamental properties of complex systems of any kind

Interaction in Statistical Mechanics I

Independence from interaction

- Some physical systems are described under the assumption of mutual independence among particles—that is, the behaviour of the particles is unaffected by their mutual interaction
  - e.g., ideal gas [Bol64]
- There, the probability distribution of the whole system is the product of those of each of its particles
- In computer science terms, the *properties of the system* can be compositionally derived by the *properties of the individual components* [Weg97]

→ Neither macroscopic sudden shift nor abrupt change for the system as a whole: technically, those systems have no phase transitions—of course, while the "independence from interaction" hypothesis holds

Interaction in Statistical Mechanics II

Interacting systems

- Introducing interaction among particles structurally *changes* the *macroscopic properties*, along with the *mathematical* ones
- Interacting systems are systems where particles *do not behave independently* of each other
- The probability distribution of an interacting system does not factorise anymore
- In computer science terms, an interacting system is *non-compositional* [Weg97]

Interaction in Statistical Mechanics III

Interacting vs. non-interacting systems

- Only interacting systems can describe real cases beyond the idealised ones
  - e.g., they can explain phase transitions – like liquid-gas transition – and much more, such as collective emerging effects
- While a system made of independent parts can be represented by isolated single nodes, an *interacting system* is better described by *nodes connected by lines* or higher-dimensional objects
- From the point of view of information and communication theories, an ideal non-interacting gas is a system of *non-communicating nodes*, whereas an interacting system is made of *nodes connected by channels*

Complexity in Statistical Mechanics I

The case of magnetic particles

- The simplest standard prototype of an interacting system is the one made of magnetic particles
- There, individual particles can behave according to a magnetic field which leaves their probabilistic independence undisturbed
- At the same time, two magnetic particles interact with each other, and the strength of their interaction is a crucial tuning parameter to observe a phase transition
  - If interaction is weak, the effect of a magnetic field is smooth on the system
  - Instead, if the interaction is strong – in particular, higher than a threshold – even a negligible magnetic field can cause a powerful *cooperative effect* on the system

Complexity in Statistical Mechanics II

Interaction is not enough

- Interaction is a necessary ingredient for complexity in statistical mechanics but definitely not a sufficient one
- Complexity arises when the possible equilibrium states of a system grow very quickly with the number of particles, regardless of the simplicity of the laws governing each particle and their mutual interaction
- Roughly speaking, complexity is much more related to size in number, rather than to complexity of the laws ruling interaction

→ we do *not* need *complex interaction* to make interaction lead to complexity

From Statistical Mechanics to Social Systems I

Large numbers

- The key point in statistical mechanics is to relate the *macroscopic* observables quantities – like pressure, temperature, etc. – to suitable *averages* of *microscopic* observables—like particle speed, kinetic energy, etc.
- Based on the *laws of large numbers*, the method works for those systems made of a large number of particles / basic components

From Statistical Mechanics to Social Systems II

Beyond the boundaries

- *Methods for complex systems* from statistical mechanics have expanded from physics to fields as diverse as biology [Kau93], economics [BP03, MS99], and computer science itself [MM09, Nis01]
- Recently, they have been applied to *social sciences* as well: there is evidence that the complex behaviour of many observed socio-economic systems can be approached with the *quantitative tools* from statistical mechanics
  - e.g., *Econophysics* for crisis events [Sta08]

From Statistical Mechanics to Social Systems III

Social systems as statistical mechanical systems

- A group of isolated individuals neither knowing nor communicating with each other is the typical example of a *compositional* social system
- No sudden shifts are expected in this case at the collective level, unless it is caused by strong external exogenous causes
- To obtain a *collective behaviour* displaying *endogenous* phenomena, the individual *agents* should meaningfully *interact* with each other
- The foremost issue here is that the nature of the interaction determines the nature of the collective behaviour at the aggregate level
  - e.g., a simple *imitative* interaction is capable to cause strong polarisation effects even in presence of extremely small external inputs

Modelling vs. Engineering

Physical vs. computational systems

- Physical systems are to be observed, understood, and possibly *modelled*
  - → For physical systems, the laws of interaction, and their role for complexity, are to be *taken as given*, to be possibly formalised mathematically by physicists
- Computational systems are to be *designed* and built
  - → For computational systems, the laws of interaction have first to be *defined* through amenable abstractions and computational models by computer scientists, then exploited by computer engineers in order to build systems

Coordinated Systems as Interacting Systems I

Coordination media for ruling interaction

- Defining the abstractions for ruling the interaction space in computational systems basically means to define their *coordination model* [GC92, Cia96, COZ99]
- *Global properties* of complex coordinated systems depending on interaction can be enforced through the *coordination model*, essentially based on its expressiveness [Zav98, DNO98]
    - For instance, tuple-based coordination models have been shown to be expressive enough to support self-organising coordination patterns for nature-inspired distributed systems [Omi13b]

Coordinated Systems as Interacting Systems II

The role of coordination models

Coordination models could be exploited

- to *rule* the *interaction space*
- so as to *define* new sorts of *global*, macroscopic *properties* for *computational systems*, possibly inspired by physical ones

Coordinated Systems as Interacting Systems III

Research perspectives

One should understand

- how to relate methods from statistical mechanics with coordination models
- whether notions such as *phase*, *phase transition*, or any other macroscopic system property, could be transferred from statistical mechanics to computer science
- what such notions would imply for computational systems
- whether new, original notions could apply to computational systems
- which sort of coordination model could support such notions

Socio-Technical Systems

Humans vs. software

- Nowadays, a particularly-relevant class of *social systems* is represented by socio-technical systems
- In socio-technical systems
  - active components are mainly represented by *humans*
  - whereas interaction is almost-totally regulated by the *software infrastructure*
  - where *software agents* often play a key role
- This is the case, for instance, of *social platforms* like FaceBook [Fac14] and LiquidFeedback [Liq14]

Physical & Computational Social Systems I

A twofold view of socio-technical systems

- The nature of socio-technical systems is twofold: they are both social systems and computational systems [VNBdV13, Omi12]
- As *complex social systems*, their complex behaviour is in principle amenable of mathematical modelling and prediction through notions and tools from statistical mechanics
- As *complex computational systems*, they are designed and built around some (either implicit or explicit) notion of coordination, ruling the interaction within components of any sort—be them either software or human ones

Physical & Computational Social Systems II

Computational systems meet physical systems

- In socio-technical systems, macroscopic properties could be
  - described by exploiting the conceptual tools from physics
  - enforced by the coordination abstractions
- Socio-technical systems could exploit both
  - the notion of complexity by statistical mechanics, along with the mathematical tools for behaviour modelling and prediction, and
  - coordination models and languages to suitably shape the interaction space

Physical & Computational Social Systems III

Vision

Complex socio-technical systems could be envisioned

- whose implementation is based on suitable coordination models
- whose macroscopic properties can be modelled and predicted by means of mathematical tools from statistical physics

thus reconciling the scientist and the engineer views over systems

Outline

Coordination for Simulation I

Simulation of complex systems is a multidisciplinary issue

- . . . ranging from physics to biology, from economics to social sciences
- no complex system of any sort can be studied nowadays without the support of suitable simulation tools
- nowadays, experiments done *in silico* are at least as relevant as those *in vitro* and *in vivo*

Coordination for Simulation II

Interaction issues are prominent in complex systems

- coordination technologies potential core of agent-based simulation frameworks
- in particular, self-organising nature-inspired coordination models are well suited for the simulation of complex systems
- so, coordination middleware could play a central role in the development of rich agent-based simulation frameworks for complex systems

Case Study: Simulating Intracellular Signalling Pathways I

Intracellular signalling pathways

- Intracellular signalling involves several molecular processes along with a huge amount of signalling elements, including several kinds of proteins
- Signal transduction pathways activated by G-proteins interact with one another to form a complex network that regulates diverse cellular components and controls a wide range of cellular processes [NRI02]
- The Ras-regulated signal transduction pathways are a classical example of this kind of network [Dow03]

Case Study: Simulating Intracellular Signalling Pathways II

Interaction issues in intracellular signalling pathways

- To model intracellular signalling systems, complex interaction that governs their behaviour should be first of all considered and understood
- Though determining the kinetic equations of the biochemistry involved in vital functions is important, managing *interactions* for the cell to make the correct physiological decisions is even more so
- Simulation of intracellular signalling pathways could be framed as mostly a coordination issue
-

Case Study: Simulating Intracellular Signalling Pathways III

Biochemical coordination

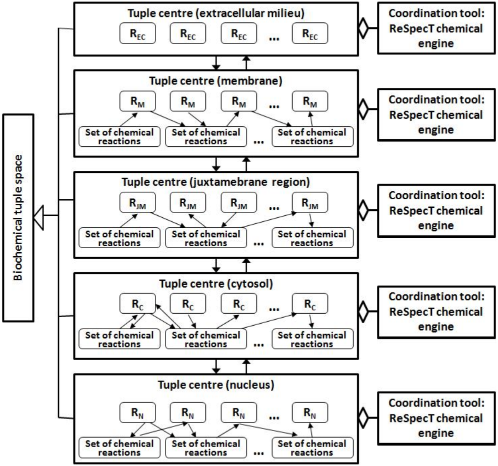
- *Biochemical tuple spaces* [VC09] are the core of a model for *self-organising coordination* (BTS-SOC)
- A biochemical tuple space is a tuple space working as a compartment where *biochemical reactions* take place
- Tuples in BTS-SOC are associated with an activity/pertinency value, resembling chemical *concentration*, and allowing chemical reactants to be represented as *tuples*
- Biochemical laws are represented as *coordination laws* by the coordination abstraction, evolving tuple concentration over time according to a rate in the same way as chemical substances into a solution
- Also, BTS-SOC laws allow for tuple *diffusion*, making it possible for products to cross compartment boundaries as a result of biochemical reactions

Case Study: Simulating Intracellular Signalling Pathways IV

Mapping cellular components and structures involved in intracellular signalling onto BTS-SOC abstractions [GPOS13]

Cellular components and structures involved in intracellular signalling	**Computational abstractions of the BTS-SOC model**
Extracellular milieu and intracellular compartments *(i.e., membrane, juxtamembrane region, cytosol, nucleus)*	Tuple centres
Signalling components *(i.e., membrane receptors, proteins, enzymes and genes)*	Chemical reactions sets
Signalling molecules *(i.e., first and secondary messengers)*, activation and deactivation signals	Reactants and concentrations recorded as tuples in the tuple centre

Case Study: Simulating Intracellular Signalling Pathways V



Some Final Remarks on Trans-disciplinary Research I

- The results of trans-disciplinary research efforts may appear quite obvious, once they are seen a posteriori
- Just above, a nature-inspired model developed in computational terms (biochemical tuple spaces) is exploited as a computational support to the simulation of a natural system (intracellular signalling pathways)
    - In other terms, from the natural world to the computational one, and back—and it works, as one might expect

Some Final Remarks on Trans-disciplinary Research II

- However, one should also understand that trans-disciplinary research succeeds when each translation of findings between the different fields involved actually enriches the associated concepts and techniques
    - Above, the BTS-SOC approach features the properties deriving from its biochemical inspiration along with those of tuple-based coordination for complex computation systems
    - When brought back to the 'natural domain' as a tool for biochemical simulation, BTS-SOC fits well for its natural inspiration, but its good performance in terms of expressive capabilities and computational efficiency *also* depends on its tuple-based structure

Some Final Remarks on Trans-disciplinary Research III

- So, while natural inspiration does not per se ensure the appropriateness of a computational approach to natural system simulation, it may in principle provide a sound grounding for the simulation of natural systems
    - Biochemical inspiration of the BTS-SOC model seems to couple well with the properties of tuple-based coordination
    - BTS-SOC turns out to be a suitable framework for the simulation of biochemical systems

Outline

Probability

- *Probability* measures how likely some event will occur
- At its core, probability provides a mathematical framework to describe *casual* events
- . . . where casual essentially means non-deterministic
- By definition, probability deals with *single occurrences* of events
- From a scientific viewpoint, a probabilistic description *per se* has *no predictive value*: it cannot really predict the precise outcome of a phenomenon
- In any case, probability provides an *a priori* model for non-deterministic phenomena

Statistics & Probability

- *Statistics* describes / analyses / interprets phenomena starting from the data available about them
- Whenever a phenomenon has no *a priori* mathematical model (at least, not yet), statistics is concerned with getting one *a posteriori* from the available data
- Accordingly, statistics is concerned with *several occurrences* of (non-deterministic) events
- Probability typically provide the mathematical tools to explain data & build a model

Stochastic Systems

- *Stochastic systems* are non-deterministic systems
- A stochastic system is one whose states are determined probabilistically
- More generally, any phenomenon requiring probability for its description is (at least in part) stochastically by definition
- Roughly speaking, a probabilistic model for a stochastic system provides a *predictive framework* for a non-deterministic system

→ We cannot predict the single occurrence of a non-deterministic event, but we can predict the overall outcome of repeated occurrences of a non-deterministic event

Non-determinism, Coordination & Stochastic Behaviour

- Autonomous systems such as *adaptive* and *self-** ones are *stochastic systems* at their very heart
- Accordingly, a foremost feature of computational models for adaptive and self-* systems is *non-determinism*
- In order to obtain stochastic behaviours of computational systems, suitable mechanisms for non-determinism should be provided
- Since most of the complexity featured by adaptive and self-* systems depends on the interaction among components, coordination models should feature non-deterministic coordination mechanisms for stochastic behaviour

Issues

- Devising out some *basic mechanisms* for stochastic coordination
- Finding a *minimal* set of primitives for most (all) of the most relevant stochastic systems
- Showing how such mechanisms could be embedded as *tuple-based* co-ordination primitives, in order to address the general need of complex computational system engineering
- Defining their formal semantics and implementing them as TuCSoN primitives

Don't Care Non-determinism in Tuple-based Models

- LINDA features *don't know* non-determinism handled with a *don't care* approach:

  don't know which tuple among the matching ones is retrieved by a getter operation (`in`, `rd`) can be neither specified nor predicted

  don't care nonetheless, the coordinated system is designed so as to keep on working whichever is the matching tuple returned

- Instead, adaptive and self-organising systems require stochastic behaviours like *"most of the time do this"*, *"sometimes do that"*
- Possibly with some quantitative specification of *"most of the time"* and *"sometimes"*

→ As it is, non-determinism in tuple-based models does not fit the need of stochastic behaviour specification

Linda "Local" Nature – In Time & Space

No context — In a single getter operation, only a *local*, point-wise property affects tuple retrieval: that is, the conformance of a tuple to the template, independently of the *spatial* context

- in fact, standard getter primitives return a matching tuple independently of the other tuples currently in the same space—so, they are *"context unaware"*

No history — Furthermore, in a sequence of getter operations, don't know non-determinism makes any prediction of the overall behaviour impossible. Again, then, only a point-wise property can be ensured even in *time*

- sequences of standard getter operations present no meaningful distribution over time

LINDA: How to Roll a Dice?

- We define tuple space `dice`
- We represent a six-face dice as a collection of six tuples: `face(1)`, ..., `face(6)`
- We roll a dice by `rd`-ing a `face/1` tuple from `dice`:

```
dice ? rd(face(X))
```

! We do *not* obtain the overall (stochastic) behaviour of a dice: for instance, it may reasonably happen that rolling the dice 10^9 times *always* results in `X / 1`—that is, we get "1" 10^9 times in a row.

uLinda: Probabilistic Non-determinism

- We define uniform coordination primitives (`uin`, `urd`) – first mentioned in [GVCO07] – as the *specialisation* of Linda getter primitives featuring probabilistic non-determinism instead of don't know non-determinism
- We call the new model uLinda [MO13c]
- Uniform primitives allow programmers to both specify and (statistically) predict the probability to retrieve one specific tuple among a bag of matching tuples
- Uniform primitives are the *"basic mechanisms enabling self-organising coordination"*—that is, a minimal set of constructs able (*alone*) to impact the observable properties of a coordinated system

uLinda: "Global" Nature

Situation & prediction

Uniform primitives replace don't know non-determinism with *probabilistic non-determinism* to

- situate a primitive invocation in space
  - uniform getter primitives return matching tuples based on the other tuples in the space—so, their behaviour is *context-aware*
- predict its behaviour in time
  - sequences of uniform getter operations tend to globally exhibit a *uniform distribution* over time

uLinda: How to Roll a Dice?

- Again, we define tuple space `dice`
- Again, we represent a six-face dice as a collection of six tuples: `face(1)`, ..., `face(6)`
- We roll a dice by `urd`-ing a `face/1` tuple from `dice`:

```
dice ?  urd(face(X))
```

! Now, we *do* obtain the overall (stochastic) behaviour of a dice:

context — at every roll, the six faces of the dice `X / 1`, ..., `X / 6` have the same *probability* $P = 1/6$ to be selected

history — in the overall, repeating several times a roll, the six faces will tend to converge towards a uniform distribution

Informal Semantics

Operationally, uniform primitives behave as follows:

1. When executed, a uniform primitive takes a *snapshot* of the tuple space, "freezing" its state at a certain point in time—and space, being a single tuple space the target of basic LINDA primitives
2. The snapshot is then exploited to assign a probabilistic value $p_i \in [0, 1]$ to any tuple $t_{i \in \{1..n\}}$ in the space—where n is the total number of tuples in the space
3. There, non-matching tuples have value $p = 0$, matching tuples have value $p = 1/m$ (where $m \leq n$ is the number of matching tuples), and the overall sum of probability values is $\sum_{i=1..n} p_i = 1$
4. The choice of the matching tuple to be returned is then statistically based on the computed probabilistic values

Formal Semantics I

[!] In order to define the semantics of (getter) uniform primitives, we rely upon a simplified version of the process-algebraic framework in [Bra08], in particular the ↑ operator, dropping multi-level priority probabilities.

uin semantics

[SYNCH-C] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \xrightarrow{T} \texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, v_1), .., (t_n, v_n)\}$

[CLOSE-C] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, v_1), .., (t_n, v_n)\}$
$$\hookrightarrow$$
$\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, p_1), .., (t_n, p_n)\}$

[EXEC-C] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{.., (t_j, p_j), ..\} \xrightarrow{t_j}_{p_j} P[t_j/T] \mid \langle t_1, .., t_n \rangle \backslash t_j$

Formal Semantics II

[!] As for standard LINDA getter primitives, the only difference between uniform reading (`urd`) and uniform consumption (`uin`) is the non-destructive semantics of the reading primitive—transition EXEC-R.

`urd` semantics

[SYNCH-C] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \xrightarrow{T} \texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, v_1), .., (t_n, v_n)\}$

[CLOSE-C] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, v_1), .., (t_n, v_n)\}$

$$\hookrightarrow$$

$\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{(t_1, p_1), .., (t_n, p_n)\}$

[EXEC-R] $\texttt{uin}_T.P \mid \langle t_1, .., t_n \rangle \uparrow \{.., (t_j, p_j), ..\} \xrightarrow{t_j}_{p_j} P[t_j/T] \mid \langle t_1, .., t_n \rangle$

Expressiveness: uLinda vs Linda

In [BGLZ05], authors demonstrate that Linda-based languages *cannot* implement probabilistic models.

PME proof

The gain in expressiveness brought by uLinda is formally proven in [MO13a], where uniform primitives are shown to be strictly more expressive than standard Linda primitives according to probabilistic modular embedding (PME) [MO13b].

In particular

$$\text{uLinda} \succeq_p \text{Linda} \quad \wedge \quad \text{Linda} \not\succeq_p \text{uLinda}$$
$$\Longrightarrow \quad \text{uLinda} \not\equiv_o \text{Linda}$$

where

- $\succeq_p$ stands for "probabilistically embeds"
- $\equiv_o$ means "(PME) observational equivalence"

Outline

Expressing Full Dynamics

Expressing the *full dynamics* of complex natural systems

- mostly, coordination models just capture *some* of the overall system dynamics
- which makes them basically *fail*
  - for instance, Gamma mimics chemical reactions, but does not capture essential issues in chemical processes such as reaction rates and concentration [BLM90, BFLM01]
  - instead, *(bio)chemical tuple spaces* fully exploit the chemical metaphor by providing time-dependent and stochastic chemical laws [VCNO10, VC09]
- more generally, the goal is to allow coordinated MAS to capture and express the full dynamics of complex natural systems

Outline

Core Mechanisms

Understanding the basic elements of expressiveness

- LINDA is a glaring example of a minimal set of coordination mechanisms providing a wide range of coordination behaviours
- the goal is understanding the minimal set of coordination primitives required to design complex stochastic behaviours
- for instance, *uniform coordination primitives* – that is, LINDA-like coordination primitives returning tuples matching a template with a uniform distribution [GVCO07] – seemingly capture the full-fledged dynamics of real chemical systems within the coordination abstractions

Outline

Blending Metaphors

Mixing abstractions & mechanisms from different conceptual sources

- most natural systems, when observed in their whole complexity, exhibit *layers* each one featuring its own metaphors and mechanisms
- correspondingly, many novel approaches to complex MAS coordination integrate diverse sources of inspiration, e.g.:
    - TOTA [MZ04b] exploits mechanisms from both stigmergic and field-based coordination
    - the SAPERE coordination model for pervasive service ecosystems [ZCF+11, VPMS12] integrates
        - the *chemical* metaphor for driving the evolution of coordination abstractions
        - *biochemical* abstractions for topology and diffusion
        - the notion of *ecosystem* in order to model the overall system structure and dynamics

Outline

Predicting Complex Behaviours

Engineering unpredictable systems around predictable abstractions

- coordination models are meant to harness the complexity of complex MAS [COZ00]
- coordination abstractions are often *at the core* of complex MAS
- while this does not make complex MAS generally predictable, it makes it possible in principle to make them *partially predictable*, based on the predictably of the core coordinative behaviour
- suitably-formalised coordination abstractions, along with a suitably-defined engineering methodology, could in principle ensure the predictability of given MAS properties within generally-unpredictable MAS

Outline

Knowledge-oriented Coordination I

Integrating nature-inspired with knowledge-oriented coordination

- intelligent MAS in knowledge intensive environments – as well as complex socio-technical systems, in general – require automatic understanding of data and information
- knowledge-oriented coordination exploits coordination abstractions enriched so as to allow for semantic interpretation by intelligent agents [Fen04, NOV13]
- for instance
  - chemical tuple spaces
  - SAPERE coordination abstractions and mechanisms
  - *semantic tuple centres* [NOVS11]

all relay on the semantic interpretation of coordination items

Knowledge-oriented Coordination II

Self-organisation of knowledge

- explicit search of information is going to become ineffective while the amount of available knowledge grows at incredible rates
- knowledge should autonomously organise and flow from producers to consumers
- knowledge self-organisation for knowledge-intensive MAS

Knowledge-oriented Coordination III

MoK (Molecules of Knowledge) [MO12a]

- Molecules of Knowledge is a a nature-inspired coordination model promoting knowledge self-organisation, where
  - sources of knowledge continuously produce and inject *atoms of knowledge* in biochemical compartments
  - knowledge atoms may then aggregate in *molecules* and diffuse
  - knowledge producers, managers and consumers are modelled as *catalysts*, whose workspaces are biochemical compartments, and their knowledge-oriented actions become enzymes influencing atoms aggregation and molecules diffusion
  - so as to make relevant knowledge spontaneously aggregate and autonomously move towards potentially interested knowledge workers
- the first application scenario for experimenting with MoK is *news management* [MO12b]

Conclusion I

History and evolution

- starting from early chemical and stigmergic approaches, nature-inspired models of coordination evolved to become the potential *core* of *complex MAS*—such as pervasive, knowledge-intensive, and self-* MAS
- in this talk we shorty surveyed their history, devise their main issues, and point out the most promising trends
- focussing in particular on tuple-based coordination models, and adopted a *systemic view* over MAS

Conclusion II

In the overall...

- nature-inspired models of coordination already have a long history behind them
- and apparently a huge *potential* for development still to be explored
- to provide core abstractions and technologies for the engineering of complex MAS

Bibliography I

Gérard Berry.
The chemical abstract machine.
Theoretical Computer Science, 96(1):217–248, April 1992.

Jean-Pierre Banâtre, Pascal Fradet, and Daniel Le Métayer.
Gamma and the chemical reaction model: Fifteen years after.
In Cristian S. Calude, Gheorghe Păun, Grzegorz Rozenberg, and Arto Salomaa, editors, *Multiset Processing. Mathematical, Computer Science, and Molecular Computing Points of View*, volume 2235 of *LNCS*, pages 17–44. Springer, 2001.

Mario Bravetti, Roberto Gorrieri, Roberto Lucchi, and Gianluigi Zavattaro.
Quantitative information in the tuple space coordination model.
Theoretical Computer Science, 346(1):28–57, 23 November 2005.

Bibliography II

Nadia Busi, Roberto Gorrieri, and Gianluigi Zavattaro.
A process algebraic view of Linda coordination primitives.
Theoretical Computer Science, 192(2):167–199, 1998.

Jean-Pierre Banâtre and Daniel Le Métayer.
The GAMMA model and its discipline of programming.
Science of Computer Programming, 15(1):55–77, November 1990.

Ludwig Boltzmann.
Lectures on Gas Theory.
University of California Press, 1964.

Bibliography III

Jean-Philippe Bouchaud and Marc Potters.
Theory of Financial Risk and Derivative Pricing: From Statistical Physics to Risk Management.
Cambridge University Press, Cambridge, UK, 2nd edition, December 2003.

Mario Bravetti.
Expressing priorities and external probabilities in process algebra via mixed open/closed systems.
Electronic Notes in Theoretical Computer Science, 194(2):31–57, 16 January 2008.

Paolo Ciancarini.
Coordination models and languages as software integrators.
ACM Computing Surveys, 28(2):300–302, June 1996.

Bibliography IV

Paolo Ciancarini, Andrea Omicini, and Franco Zambonelli.
Coordination technologies for Internet agents.
Nordic Journal of Computing, 6(3):215–240, Fall 1999.

Paolo Ciancarini, Andrea Omicini, and Franco Zambonelli.
Multiagent system engineering: The coordination viewpoint.
In Nicholas R. Jennings and Yves Lespérance, editors, *Intelligent Agents VI. Agent Theories, Architectures, and Languages*, volume 1757 of *LNAI*, pages 250–259. Springer, 2000.
6th International Workshop (ATAL'99), Orlando, FL, USA, 15–17 July 1999. Proceedings.

Rocco De Nicola, Gianluigi Ferrari, and Rosario Pugliese.
KLAIM: A kernel language for agent interaction and mobility.
IEEE Transaction on Software Engineering, 24(5):315–330, May 1998.

Bibliography V

Rocco De Nicola, Diego Latella, Joost-Pieter Katoen, and Mieke Massink.
StoKlaim: A stochastic extension of Klaim.
Technical Report 2006-TR-01, Istituto di Scienza e Tecnologie dell'Informazione "Alessandro Faedo" (ISTI), 2006.

Enrico Denti, Antonio Natali, and Andrea Omicini.
On the expressive power of a language for programming coordination media.
In *1998 ACM Symposium on Applied Computing (SAC'98)*, pages 169–177, Atlanta, GA, USA, 27 February – 1 March 1998. ACM.
Special Track on Coordination Models, Languages and Applications.

Bibliography VI

Julian Downward.
Targeting RAS signalling pathways in cancer therapy.
Nature Reviews Cancer, 3(1):11–22, 2003.

FaceBook.
Home page.
`http://www.facebook.com`, 2014.

Dieter Fensel.
Triple-space computing: Semantic web services based on persistent publication of information.
In Finn Arve Aagesen, Chutiporn Anutariya, and Vilas Wuwongse, editors, *Intelligence in Communication Systems*, volume 3283 of *LNCS*, pages 43–53, 2004.

Bibliography VII

IFIP International Conference (INTELLCOMM 2004), Bangkok, Thailand, 23–26 November 2004. Proceedings.

David Gelernter and Nicholas Carriero.
Coordination languages and their significance.
Communications of the ACM, 35(2):97–107, 1992.

David Gelernter.
Generative communication in Linda.
ACM Transactions on Programming Languages and Systems, 7(1):80–112, January 1985.

Bibliography VIII

Pedro Pablo González Pérez, Andrea Omicini, and Marco Sbaraglia.
A biochemically-inspired coordination-based model for simulating intracellular signalling pathways.
Journal of Simulation, 7(3):216–226, August 2013.
Special Issue: Agent-based Modeling and Simulation.

Pierre-Paul Grassé.
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, March 1959.

Bibliography IX

Luca Gardelli, Mirko Viroli, Matteo Casadei, and Andrea Omicini.
Designing self-organising MAS environments: The collective sort case.
In Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, editors, *Environments for MultiAgent Systems III*, volume 4389 of *LNAI*, pages 254–271. Springer, May 2007.
3rd International Workshop (E4MAS 2006), Hakodate, Japan, 8 May 2006. Selected Revised and Invited Papers.

Stuart A. Kauffman.
The Origins of Order: Self-organization and Selection in Evolution.
Oxford University Press, January 1993.

Stuart A. Kauffman.
Investigations.
Oxford University Press, January 2003.

Bibliography X

LiquidFeedback.
Home page.
http://liquidfeedback.org, 2014.

Jiming Liu and Kwok Ching Tsui.
Toward nature-inspired computing.
Communications of the ACM, 49(10):59–64, October 2006.

Marc Mézard and Andrea Montanari.
Information, Physics, and Computation.
Oxford University Press, Oxford, UK, January 2009.

Bibliography XI

Stefano Mariani and Andrea Omicini.
Molecules of Knowledge: Self-organisation in knowledge-intensive environments.
In Giancarlo Fortino, Costin Bădică, Michele Malgeri, and Rainer Unland, editors, *Intelligent Distributed Computing VI*, volume 446 of *Studies in Computational Intelligence*, pages 17–22. Springer, 2012.
6th International Symposium on Intelligent Distributed Computing (IDC 2012), Calabria, Italy, 24-26 September 2012. Proceedings.

Bibliography XII

Stefano Mariani and Andrea Omicini.
Self-organising news management: The *Molecules of Knowledge* approach.
In José Luis Fernandez-Marquez, Sara Montagna, Andrea Omicini, and Franco Zambonelli, editors, *1st International Workshop on Adaptive Service Ecosystems: Natural and Socially Inspired Solutions (ASENSIS 2012)*, pages 11–16, SASO 2012, Lyon, France, 10 September 2012.
Pre-proceedings.

Stefano Mariani and Andrea Omicini.
Probabilistic embedding: Experiments with tuple-based probabilistic languages.
In *28th ACM Symposium on Applied Computing (SAC 2013)*, pages 1380–1382, Coimbra, Portugal, 18–22 March 2013.
Poster Paper.

Bibliography XIII

Stefano Mariani and Andrea Omicini.
Probabilistic modular embedding for stochastic coordinated systems.
In Christine Julien and Rocco De Nicola, editors, *Coordination Models and Languages*, volume 7890 of *LNCS*, pages 151–165. Springer, 2013.
15th International Conference (COORDINATION 2013), Florence, Italy, 3–6 June 2013. Proceedings.

Bibliography XIV

Stefano Mariani and Andrea Omicini.
Tuple-based coordination of stochastic systems with uniform primitives.
In Matteo Baldoni, Cristina Baroglio, Federico Bergenti, and Alfredo Garro, editors, *From Objects to Agents*, volume 1099 of *CEUR Workshop Proceedings*, pages 8–15, Turin, Italy, 2–3 December 2013. Sun SITE Central Europe, RWTH Aachen University.
XIV Workshop (WOA 2013). Workshop Notes.

Rosario N. Mantegna and H. Eugene Stanley.
Introduction to Econophysics: Correlations and Complexity in Finance.
Cambridge University Press, Cambridge, UK, December 1999.

Bibliography XV

Marco Mamei and Franco Zambonelli.
Programming pervasive and mobile computing applications with the TOTA middleware.
In *Pervasive Computing and Communications*, pages 263–273, 2004.
2nd IEEE Annual Conference (PerCom 2004), Orlando, FL, USA, 14–17 March 2004. Proceedings.

Marco Mamei and Franco Zambonelli.
Programming pervasive and mobile computing applications with the TOTA middleware.
In *Pervasive Computing and Communications*, pages 263–273, 2004.
2nd IEEE Annual Conference (PerCom 2004), Orlando, FL, USA, 14–17 March 2004. Proceedings.

Bibliography XVI

Marco Mamei and Franco Zambonelli.
Field-Based Coordination for Pervasive Multiagent Systems. Models, Technologies, and Applications.
Springer Series in Agent Technology. Springer, March 2006.

Hidetoshi Nishimori.
Statistical Physics of Spin Glasses and Information Processing: An Introduction, volume 111 of *International Series of Monographs on Physics*.
Clarendon Press, Oxford, UK, July 2001.

Elena Nardini, Andrea Omicini, and Mirko Viroli.
Semantic tuple centres.
Science of Computer Programming, 78(5):569–582, May 2013.
Special section: Self-Organizing Coordination.

Bibliography XVII

Elena Nardini, Andrea Omicini, Mirko Viroli, and Michael Ignaz Schumacher.
Coordinating e-health systems with TuCSoN semantic tuple centres.
Applied Computing Review, 11(2):43–52, Spring 2011.

Susana R. Neves, Prahlad T. Ram, and Ravi Iyengar.
G protein pathways.
Science, 296(5573):1636–1639, May 2002.

Andrea Omicini and Enrico Denti.
From tuple spaces to tuple centres.
Science of Computer Programming, 41(3):277–294, November 2001.

Bibliography XVIII

Andrea Omicini.
Agents writing on walls: Cognitive stigmergy and beyond.
In Fabio Paglieri, Luca Tummolini, Rino Falcone, and Maria Miceli, editors, *The Goals of Cognition. Essays in Honor of Cristiano Castelfranchi*, volume 20 of *Tributes*, chapter 29, pages 543–556. College Publications, London, December 2012.

Andrea Omicini.
Agents writing on walls: Cognitive stigmergy and beyond.
In Fabio Paglieri, Luca Tummolini, Rino Falcone, and Maria Miceli, editors, *The Goals of Cognition. Essays in Honor of Cristiano Castelfranchi*, chapter 29, pages 543–556. College Publications, London, 2013.

Bibliography XIX

Andrea Omicini.
Nature-inspired coordination for complex distributed systems.
In Giancarlo Fortino, Costin Bădică, Michele Malgeri, and Rainer Unland, editors, *Intelligent Distributed Computing VI*, volume 446 of *Studies in Computational Intelligence*, pages 1–6. Springer, 2013.
6th International Symposium on Intelligent Distributed Computing (IDC 2012), Calabria, Italy, 24-26 September 2012. Proceedings.
Invited paper.

Bibliography XX

Andrea Omicini, Alessandro Ricci, Mirko Viroli, Cristiano Castelfranchi, and Luca Tummolini.
Coordination artifacts: Environment-based coordination for intelligent agents.
In Nicholas R. Jennings, Carles Sierra, Liz Sonenberg, and Milind Tambe, editors, *3rd international Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004)*, volume 1, pages 286–293, New York, USA, 19–23 July 2004. ACM.

Bibliography XXI

Andrea Omicini, Alessandro Ricci, and Mirko Viroli.
Time-aware coordination in ReSpecT.
In Jean-Marie Jacquet and Gian Pietro Picco, editors, *Coordination Models and Languages*, volume 3454 of *LNCS*, pages 268–282. Springer-Verlag, April 2005.
7th International Conference (COORDINATION 2005), Namur, Belgium, 20–23 April 2005. Proceedings.

Andrea Omicini, Alessandro Ricci, and Mirko Viroli.
The multidisciplinary patterns of interaction from sciences to Computer Science.
In Dina Q. Goldin, Scott A. Smolka, and Peter Wegner, editors, *Interactive Computation: The New Paradigm*, pages 395–414. Springer, September 2006.

Bibliography XXII

Andrea Omicini, Alessandro Ricci, and Mirko Viroli.
Timed environment for Web agents.
Web Intelligence and Agent Systems, 5(2):161–175, August 2007.

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

Bibliography XXIII

Andrea Omicini and Franco Zambonelli.
MAS as complex systems: A view on the role of declarative approaches.
In João Alexandre Leite, Andrea Omicini, Leon Sterling, and Paolo Torroni, editors, *Declarative Agent Languages and Technologies*, volume 2990 of *LNAI*, pages 1–17. Springer, May 2004.
1st International Workshop (DALT 2003), Melbourne, Australia, 15 July 2003. Revised Selected and Invited Papers.

H. Van Dyke Parunak.
A survey of environments and mechanisms for human-human stigmergy.
In Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, editors, *Environments for Multi-Agent Systems II*, volume 3830 of *LNCS*, pages 163–186. Springer, 2006.

Bibliography XXIV

H. Van Dyke Parunak, Sven Brueckner, and John Sauter.
Digital pheromone mechanisms for coordination of unmanned vehicles.
In Cristiano Castelfranchi and W. Lewis Johnson, editors, *1st International Joint Conference on Autonomous Agents and Multiagent systems*, volume 1, pages 449–450, New York, NY, USA, 15–19July 2002. ACM.

Davide Rossi, Giacomo Cabri, and Enrico Denti.
Tuple-based technologies for coordination.
In Andrea Omicini, Franco Zambonelli, Matthias Klusch, and Robert Tolksdorf, editors, *Coordination of Internet Agents: Models, Technologies, and Applications*, chapter 4, pages 83–109. Springer, January 2001.

Bibliography XXV

Alessandro Ricci, Andrea Omicini, Mirko Viroli, Luca Gardelli, and Enrico Oliva.
Cognitive stigmergy: Towards a framework based on agents and artifacts.
In Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, editors, *Environments for MultiAgent Systems III*, volume 4389 of *LNCS*, pages 124–140. Springer, May 2007.
3rd International Workshop (E4MAS 2006), Hakodate, Japan, 8 May 2006. Selected Revised and Invited Papers.

Bibliography XXVI

Alessandro Ricci, Mirko Viroli, and Andrea Omicini.
Environment-based coordination through coordination artifacts.
In Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, editors, *Environments for Multi-Agent Systems*, volume 3374 of *LNAI*, pages 190–214. Springer, February 2005.
1st International Workshop (E4MAS 2004), New York, NY, USA, 19 July 2004. Revised Selected Papers.

Herbert A. Simon.
The architecture of complexity.
Proceedings of the American Philosophical Society, 106(6):467–482, 12 December 1962.

Bibliography XXVII

H. Eugene Stanley.
Econophysics and the current economic turmoil.
American Physical Society News, 17(11):8, December 2008.
The Back Page.

Robert Tolksdorf and Ronaldo Menezes.
Using Swarm Intelligence in Linda Systems.
In Andrea Omicini, Paolo Petta, and Jeremy Pitt, editors, *Engineering Societies in the Agents World IV*, volume 3071 of *LNCS*, pages 49–65.
Springer, June 2004.
4th International Workshops (ESAW 2003), London, UK, 29-31 October 2003. Revised Selected and Invited Papers.

Bibliography XXVIII

Mirko Viroli and Matteo Casadei.
Biochemical tuple spaces for self-organising coordination.
In John Field and Vasco T. Vasconcelos, editors, *Coordination Languages and Models*, volume 5521 of *LNCS*, pages 143–162.
Springer, Lisbon, Portugal, June 2009.
11th International Conference (COORDINATION 2009), Lisbon, Portugal, June 2009. Proceedings.

Bibliography XXIX

Mirko Viroli, Matteo Casadei, Elena Nardini, and Andrea Omicini.
Towards a chemical-inspired infrastructure for self-* pervasive applications.
In Danny Weyns, Sam Malek, Rogério de Lemos, and Jesper Andersson, editors, *Self-Organizing Architectures*, volume 6090 of *LNCS*, chapter 8, pages 152–176. Springer, July 2010.
1st International Workshop on Self-Organizing Architectures (SOAR 2009), Cambridge, UK, 14-17 September 2009, Revised Selected and Invited Papers.

Bibliography XXX

Mirko Viroli, Matteo Casadei, and Andrea Omicini.
A framework for modelling and implementing self-organising coordination.
In Sung Y. Shin, Sascha Ossowski, Ronaldo Menezes, and Mirko Viroli, editors, *24th Annual ACM Symposium on Applied Computing (SAC 2009)*, volume III, pages 1353–1360, Honolulu, Hawai'i, USA, 8–12 March 2009. ACM.

Harko Verhagen, Pablo Noriega, Tina Balke, and Marina de Vos, editors.
Social Coordination: Principles, Artefacts and Theories (SOCIAL.PATH), AISB Convention 2013, University of Exeter, UK, 3–5 April 2013. The Society for the Study of Artificial Intelligence and the Simulation of Behaviour.

Bibliography XXXI

Mirko Viroli and Andrea Omicini.
Coordination as a service.
Fundamenta Informaticae, 73(4):507–534, 2006.
Special Issue: Best papers of FOCLASA 2002.

Mirko Viroli, Danilo Pianini, Sara Montagna, and Graeme Stevenson.
Pervasive ecosystems: a coordination model based on semantic chemistry.
In Sascha Ossowski, Paola Lecca, Chih-Cheng Hung, and Jiman Hong, editors, *27th Annual ACM Symposium on Applied Computing (SAC 2012)*, Riva del Garda, TN, Italy, 26–30 March 2012. ACM.

Peter Wegner.
Why interaction is more powerful than algorithms.
Communications of the ACM, 40(5):80–91, May 1997.

Bibliography XXXII

Danny Weyns, Andrea Omicini, and James J. Odell.
Environment as a first-class abstraction in multi-agent systems.
Autonomous Agents and Multi-Agent Systems, 14(1):5–30, February 2007.
Special Issue on Environments for Multi-agent Systems.

Gianluigi Zavattaro.
On the incomparability of Gamma and Linda.
Technical Report SEN-R9827, CWI, Amsterdam, The Netherlands, October 1998.

Bibliography XXXIII

Franco Zambonelli, Gabriella Castelli, Laura Ferrari, Marco Mamei, Alberto Rosi, Giovanna Di Marzo, Matteo Risoldi, Akla-Esso Tchao, Simon Dobson, Graeme Stevenson, Yuan Ye, Elena Nardini, Andrea Omicini, Sara Montagna, Mirko Viroli, Alois Ferscha, Sascha Maschek, and Bernhard Wally.
Self-aware pervasive service ecosystems.
Procedia Computer Science, 7:197–199, December 2011.
Proceedings of the 2nd European Future Technologies Conference and Exhibition 2011 (FET 11).

Bibliography XXXIV

Franco Zambonelli and Andrea Omicini.
Challenges and research directions in agent-oriented software engineering.
Autonomous Agents and Multi-Agent Systems, 9(3):253–283, November 2004.
Special Issue: Challenges for Agent-Based Computing.

Nature-Inspired Coordination & Self-Organisation

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 2015/2016