Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects

Emanuela Merelli
Università di Camerino

Reykjavik University - 18 September 2007
Plan

- Computational Systems Biology
- Strength and weakness of the formal approach
- Multi-level modeling in systems biology
- Agent-based modeling and simulation
- Case Study: from carbohydrate oxidation process

- UNICAM CoSy Research Group
- LITBIO project
- Erasmus Mundus Master Course in *Formal Methods for Systems Engineering* (FOMSE)
- Conclusion
The challenge of the 21st century will be to understand how biological molecules in living systems integrate to complex systems, how these systems function and their evolution.

We are scaling up from computational molecular biology (i.e. genomes, transcriptomes, proteomes, metabolomes) to computational systems biology.
The science of **building models** of biological systems, from cells to organisms, by:
- systematically perturbing the system (biologically, genetically, or chemically); and
- monitoring gene, protein and other pathways.

Models are used in **drug research** for understanding their effects and side-effects and for building personalized medicine.
Systems Biology is typically require tools to

- integrate the huge amount of biological data in order to
- understand the behaviour of biological systems, and to
- study the relationships and interactions between the various parts of a biological system, such as organelles, cells, physiological systems, organisms etc.
Most of the biological systems are complex in nature

▶ have many of its components coupled in a non-linear fashion
▶ exhibit the emergence property from simpler interactions rules, i.e. the formation of complex patterns
▶ ...

Thus the proposal of a multi-level approach in modelling biological systems would allow zooming-in and zooming-out into through models
From molecules to cell compartments, organisms, ... ecosystems

the global behaviour of the system can be determined by defining the lower-level interaction rules among components
Glycolysis metabolic pathway

\[ \text{glucose} + 2\text{NAD}^+ + 2\text{ADP} + 2\text{Pi} \rightarrow 2\text{pyruvate} + 2\text{NAD} + 2\text{H}^+ + 2\text{ATP} + 2\text{H}_2\text{O} \]
The Carbohydrate Oxidation is the energy production process performed by two active components of the Cell: Cytoplasm and Mitochondrion.

Metabolic pathways

- **Cytoplasm (without oxygen)**
  - Glycolysis *(convert 1 glucose into 2 pyruvate + ...)*
  - Lactic Fermentation *(reduce pyruvate to lactate)*
  - Alcoholic Fermentation *(reduce pyruvate to ethanol)*

- **Mitochondrion (with oxygen)**
  - Mitochondrial Inner Membrane
    - Transportation *(pyruvate from cytopl to MM)*
    - Respiration Chain *
  - Mitochondrial Matrix
    - Partial oxidation of pyruvate
    - Kreb’s cycle
Cytoplasm: a lower-level bio-complexity

To model a portion \((10^{-15} \text{ liters})\) of a cytoplasm, we must simulate the movement and the interactions of about 27 millions of autonomous entities.

Each interaction has associated 2 kinetic constants; in many cases those values are still undiscovered; each day one of them could be discovered by one of the hundreds of research groups distributed all over the world.
Multi-level modelling

A model (M) for a system (S) and experiment (E) is anything to which E can be applied in order to answer questions about S. M. Minsky, *Models, minds, machines*. 1965

This definition implies the co-existence of several models for any system.

A model can be distinguished w.r.t their organizational level:

- Is the system described at *macro* or at *meso* or at *micro* level?
- Are concentration changes the focus of interest, or the behaviour of individuals?
- ...

Emanuela Merelli Università di Camerino

Agent Based Modelling and Simulation in Systems Biology: M
It has been observed that the language used by biologists for verbal communications is **vague** and avoid clear prediction.

Also, the use of colourful diagrams, so much beloved from biologists, is often usually useless for computer reasoning and quantitative analysis.

**Formal techniques** are instead likely to give the chance to inherit the wide range of existing methods and tools provided by computer science (such as property design and verification, and automated reasoning).
The Ordinary Differential Equation (ODE) is the classical approach arising from the biochemical point of view.

A network of interactions (chemical reactions) between molecules (metabolites, proteins, genes) is established and ODE are used to numerically describe the continuous variations in the concentration of substances.

ODE beset with the **combinatorial explosion**
Process abstraction

Using the process abstraction opens up new possibilities for understanding molecular systems.

Computers and biomolecular systems both start from a small set of elementary components from which, layer by layer, more complex entities are constructed with evermore sophisticated functions.

The framework should also allow the system to be **zoomed-in** and **zoomed-out** at different levels of abstraction.
In recent years, there has been increasing interest in computational models of biological systems based on various calculi of communicating processes such as the stochastic $\pi$-calculus.


In 2003, Aviv Regev and Edu Shapiro proposed “the $\pi$-calculus as an abstraction for biomolecular modelling”
Albeit π-calculus allows to reflect the inherent concurrency, communication and stochasticity of cellular systems, each new case reveals drawbacks in modelling biological systems.

The minimality of the calculus makes it heavy to manage models with complex coordination and dependencies. It forces the modeler to encode high-level interaction patterns into low-level means offered by the calculus.

The encodings makes models harder to write and understand, and obscure the original high-level biological model with a plethora of irrelevant computational details.
A criticism to $\pi$-calculus

Perhaps it depends on ... the asymmetry of the send/receive communication model

Whenever two molecules $A$ and $B$ bind to form a complex $AB$, they each create a private channel. The protocol for exchanging these new private channels is itself complicated by the directionality of communication

[D. Duchier, C. Klutter *Biomolecular agents as multi-behavioural concurrent objects* ENTCS, 150, 2006]
A criticism to $\pi$-calculus

Perhaps it depends on ... the asymmetry of the send/receive communication model

not very suitable to model the *simultaneous exchange*!

Whenever two molecules $A$ and $B$ bind to form a complex $AB$, they each create a private channel. The protocol for exchanging these new private channels is itself complicated by the directionality of communication

[D. Duchier, C. Klutter *Biomolecular agents as multi-behavioural concurrent objects* ENTCS, 150, 2006]
**Coordination**, in CS, refers to languages and models for coordinating, configuring, and, generally, glueing together components. e.g., LINDA
Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

Emanuela Merelli
Università di Camerino

Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects
Shall agent-based paradigm help to overcome the limitation of the understandability of the \(\pi\)-calculus by offering a multi-level modelling environment and ...

... exploit the Grid computational resources the simulation of biological systems?
Milner’s agent point of view

In 1989 Milner wrote ...

- “each of the several part of the system has its own identity, which persist through time; we shall call these parts agents”

- ... Communication and concurrency notions, both essential in understanding complex dynamic systems. On the one hand such a system has diversity, being composed of several parts each acting concurrently with, and independently of, other parts; on the other hand a complex system has unity (or else we shall not call it a system) achieved through communication among its parts.

- ... underling both these notions is the assumption that each of the several parts of such a system has its own identity, which persists through time; we shall call these parts agents

- ... This lead to use the term agent very broadly, to mean any system whose behaviour consists of discrete actions; an which for one purpose we take to be atomic may, for other purpose, be decomposed into sub-agents acting concurrently and interacting.
An **agent** is an “encapsulated” computer system **situated** in some environment and capable of **flexible, autonomous** action in that environment in order to meet its design objectives

An autonomous agent is a computer system capable of flexible, autonomous (problem-solving) action, situated in dynamic, open, unpredictable and typically multi-agent domains.

The agent has the control over its internal state and over its own behaviour.
A situated agent is a computer system capable of flexible, autonomous (problem-solving) action, **situated** in dynamic, open, unpredictable and typically multi-agent domains.

**The agent perceives the environment through sensors and acts through effectors**
A computer system capable of **flexible**, autonomous (problem-solving) action, situated in dynamic, eventually open, unpredictable and typically multi-agent domains.

**reactive:** the agent responds in timely fashion to environmental change

**proactive:** the agent acts in anticipation of future goals
Reactive agents can be defined by a 6-tuple

\[ < E, P, A, \text{see}, \text{do}, \text{action} > \]

where

- \( E \) is the set of states for the environment
- \( P \) is a partition of \( E \) (representing a relevant abstraction of the environment from the agent’s point of view)
- \( A \) is a set of actions
- \( \text{see} : E \rightarrow P \)
- \( \text{action} : P \rightarrow A \)
- \( \text{do} : A \times E \rightarrow E \)

These agents observe the environment (\( \text{see} \)), find the appropriate action (\( \text{action} \)), and act (\( \text{do} \))
Proactive agents can be defined by a 9-tuple
\[ < D, E, P, A, d_0, \text{see}, KB, \text{do}, \text{action} > \]
where
- \( D \) is a set of predicate calculus databases
- \( E, P, A, \text{see} \) and \( \text{do} \) are the same as for reactive agents
- \( d_0 \) is the initial KB
- \( \text{action} : D \times P \rightarrow A \)
- \( KB : D \times P \rightarrow D \)

These agents observe the environment (\text{see}), choose their next action according to some kind of reasoning on their current knowledge (\text{action}), act (\text{do}), and update their knowledge base.
Mobile agents are agents that move across distributed environments (execution platforms)

Mobile agents for Global computing and mobile code for Grid computing are new paradigms for distributed computing that complement such technologies as distributed objects, remote compute servers, and distributed programming systems such as PVM and MPI.
Multi-agent system

The description of a system consists of a collection of interactive agents and a set of coordination rules.

An agent has a name and a set of behaviours. He can adopt different behaviours at different times depending on the environmental conditions.

At meso level of modelling biological systems, the elementary interactions consist of the binding or unbinding of two agents, the modification of the state of an interface (site), and the deletion or creation of an agent.
Step 1: Transformation of glucose into pyruvate.

Step 2: Transformation of acetyl CoA into NADH.

Step 3: Transportation of NADH into ATP.

Legend:
- Metabolic pathway
- Compounds reaction
- Molecules

Compartment 1: Cytoplasm
Compartment 2.1: Mitochondrial Matrix
Compartment 2: Mitochondrion
Compartment 2.2: Inner Mitochondrial Membrane
Initial Requirements

Syst. Req. Model

Ag. Impl. Model

Domain Description

Agent Structure Definition

Ontology Description

Code Model

Agent Identification

Agent Behavior Description

Roles Identification

Protocols Description

Deployment Configuration

Tasks Specification

Code Reuse

Agent Test

Agent Society Model

Code Completion

Society Test

UNICAM Complex Systems Research Group (CoSy)

KEGG Interaction Diagrams

KEGG Interaction Diagrams: Phosphoplation

SBML Description

Glycolysis model

Emanuela Merelli Università di Camerino

Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects
Carbohydrate oxidation: the PASSI agent identification UML diagram

**Agent Cytosol** simulates the role of Cytoplasm in the Glycolysis, Lactic and Alcoholic Fermentations functions.

**Agent Inner Mitochondrial Membrane** simulates the role of IMM in the Transport and Electron Respiration Chain functions.

**Agent Mitochondrial Matrix** simulates the role of MM in the Partial Oxidation of Pyruvate and Kreb’s Cycle functions.

**Environment**: Service Agent simulates the execution environment in which any cell reaction takes place.
Roles in Lactic Fermentation function
Carbohydrate oxidation: the PASSI agent identification UML diagram
Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

KEGG Interaction Diagrams
KEGG Interaction Diagrams: Phosphorylation
SBML Description
Glycolysis model
UNICAM Complex Systems Research Group (CoSy)

Emanuela Merelli Università di Camerino
Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects
UML Sequence diagram of the preliminary phase of glycolysis

1. in(2.7.1.1, glucose)
2. return(glucose)
3. in(UDP)
4. return(UDP)
5. reaction
6. out(UDP, glucose-6-P)
7. out(ADP)
8. in(5.3.1.9, glucose-6-P)
9. return(glucose-6-P)
10. reaction
11. out(11, fructose-6-P)
12. in(2.7.1.11, fructose-6-P)
13. return(fructose-6-P)
14. in(UDP)
15. return(UDP)
16. reaction
17. out(13, fructose 1, 1, 6-bisphosphate)
18. out(ADP)
19. in(5.3.1.13, fructose 1, 6-bisphosphate)
20. return(fructose 1, 6-bisphosphate)
21. reaction
22. out(1, dihydroxyacetone phosphate, 12, glyceraldehyde 3 phosphate)
23. in(5.3.1.1, dihydroxyacetone phosphate)
An agent-based "calculator" for glycolysis
But ... Zooming-in
But ... molecules move
... and then interact by affinity
... and then interact by affinity
Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

KEGG Interaction Diagrams
KEGG Interaction Diagrams: Phosphorylation
SBML Description
Glycolysis model
UNICAM Complex Systems Research Group (CoSy)
<?xml version="1.0" encoding="UTF-8" ?>
<sbml xmlns="http://www.sbml.org/sbml/level2"
xmlns:jd="http://www.sys-bio.org/sbml"
xmlns:sl2="http://projects.eml.org/bcb/sbml/level2"
level="2" version="1">
  <model metaid="metaid_0000001" id="Glycolysis_Nielsen"
name="Nielsen1998_Glycolysis">
    <notes>
      <body xmlns="http://www.w3.org/1999/xhtml">

        The reaction looks like this:
        <br />
        reaction_1: GLC + ATP -> F6P + ADP;
        <br />
        reaction_2: F6P + ATP -> FBP + ADP;
        <br />
        reaction_3: FBP => 2 * GAP;
        <br />
        reaction_4: GAP + NAD -> DPG + NADH;
        <br />
        reaction_5: DPG + ADP => PEP + ATP;
        <br />
        reaction_6: PEP + ADP -> PYR + ATP;
        <br />
        reaction_7: PYR -> ACA;
        <br />
        reaction_8: ACA + NADH => EtOH + NAD;
        <br />
        reaction_9: AMP + ATP => 2 * ADP;
        <br />
        reaction_10: F6P -> P;
        <br />
        flow reactor:
        <br />
        flow = 1.1e-2 reactorvolume/s
        <br />
        inflow concentrations:
        <br />
        ATP = 3.5 mM
        <br />
        ADP = 1.1 mM
        <br />
        NADH = 0.24 mM
        <br />
        NAD = 4.0 mM
        <br />
        GLC = 50 mM
        <br />
      </body>
    </notes>
  </model>
</sbml>
- <reaction metaid="metaid_0000059" id="reaction_1"
reversible="false"> - <annotation> - <rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:vCard="http://www.w3.org/2001/vcard-rdf/3.0#"> - <rdf:Description
rdf:about="#metaid_0000059"> - <dc:relation> - <rdf:Bag>
  <rdf:li rdf:resource="http://www.ebi.ac.uk/IntEnz/#EC 2.7.1.2"/>
  <rdf:li rdf:resource="http://www.ebi.ac.uk/IntEnz/#EC 5.3.1.9"/>
</rdf:Bag>
</dc:relation>
</rdf:Description>
</rdf:RDF>
</annotation>
-
- {\bf <listOfReactants>
  <speciesReference species="GLC"/>
  <speciesReference species="ATP"/>
</listOfReactants>
- <listOfProducts>
  <speciesReference species="F6P"/>
  <speciesReference species="ADP"/>
</listOfProducts> }
- <kineticLaw> - <math xmlns="http://www.w3.org/1998/Math/MathML"> - <apply>
times
...
</apply>
</math>
</kineticLaw>
</annotation>

Glycolysis

\[ \text{glucose} + 2\text{ATP} + 4\text{ADP} + 2\text{Pi} + 2\text{NADox} \rightarrow \]
\[ 2\text{pyruvate} + 2\text{ADP} + 4\text{ATP} + 2\text{NADrid} + 2\text{H}_2\text{O} \]
### Step Reactions

<table>
<thead>
<tr>
<th>Step</th>
<th>Reactions</th>
<th>Enzyme</th>
<th>$\Delta G^\circ$ (Kcal mol$^{-1}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>glucose, ATP $\rightleftharpoons$ glucose-6-phosphate, ADP, $H^+$</td>
<td>hexokinase</td>
<td>-4.0</td>
</tr>
<tr>
<td>2</td>
<td>glucose-6-phosphate $\rightleftharpoons$ fructose-6-phosphate</td>
<td>phos. glu. isomerase</td>
<td>+0.4</td>
</tr>
<tr>
<td>3</td>
<td>fructose-6-phosphate, ATP $\rightarrow$ fructose1,6-bisphosphate, ADP, $H^+$</td>
<td>phosphofructokinase</td>
<td>-3.4</td>
</tr>
<tr>
<td>4</td>
<td>fructose1,6-bisphosphate $\rightleftharpoons$ dihydroxyacetone phosphate, glyceraldehyde 3-phosphate</td>
<td>aldolase</td>
<td>+3.7</td>
</tr>
<tr>
<td>5</td>
<td>dihydroxyacetone phosphate $\rightleftharpoons$ glyceraldehyde 3-phosphate</td>
<td>trio. phos. isomerase</td>
<td>+1.8</td>
</tr>
<tr>
<td>6</td>
<td>glyceraldehyde 3-phosphate, Pi, NAD$^+$ $\rightleftharpoons$ 1,3-bisphosphoglycerate, NADH, $H^+$</td>
<td>glyceraldehyde 3phos. dehydro. se</td>
<td>+1.5</td>
</tr>
<tr>
<td>7</td>
<td>1,3-bisphosphoglycerate, ADP $\rightleftharpoons$ 3-phosphoglycerate, ATP</td>
<td>phosphoglyceratekinase</td>
<td>-4.5</td>
</tr>
<tr>
<td>8</td>
<td>3-phosphoglycerate $\rightleftharpoons$ 2-phosphoglycerate</td>
<td>phosphoglyceromutase</td>
<td>+1.1</td>
</tr>
<tr>
<td>9</td>
<td>2-phosphoglycerate $\rightleftharpoons$ phosphoenolpyruvate, H$_2$O</td>
<td>enolase</td>
<td>+0.4</td>
</tr>
<tr>
<td>10</td>
<td>phosphoenolpyruvate, ADP, $H^+$ $\rightleftharpoons$ enolpyruvate, ATP</td>
<td>pyruvatekinase</td>
<td>-7.5</td>
</tr>
<tr>
<td>11</td>
<td>enolpyruvate $\rightarrow$ pyruvate</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
LITBIO Project

In-silico experiments → Resource Discovery → BioSimulation

- WeakFlow
- XPDL to MMS Compiler
- Workflow Executors
- BioWMS
- Resourceome
- Hermes
  - agent-based middleware
- Orion
- OGSA

Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

KEGG Interaction Diagrams
KEGG Interaction Diagrams: Phosphorylation
SBML Description
Glycolysis model
UNICAM Complex Systems Research Group (CoSy)

Emanuela Merelli Università di Camerino
Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects
Erasmus Mundus Master Course in *Formal Methods for Systems Engineering* (FOMSE)

The master curricula are specified by three tracks. Every student chooses a track, which defines the mobility plan (between two sites, at the end of the first year) of the student. The choice is driven by the specialization/thesis/research interest specified in the name of the track. Every track has a minimum number of ECTS (30) covering the following Basic Topics:

- Process Algebra
- Distributed and Concurrent Computation
- Theory of Computation
Erasmus Mundus Partnership
Master Tracks

- **Systems Biology**: Year 1 Reykjavik (Basic Topics + Courses in Bioinformatics and Systems Biology + free choices). Year 2 Camerino (Specialization + Research Credits + Thesis)

- **Software Systems Engineering**: Year 1 Camerino (Basic Topics + Service Oriented Architectures + free choices). Year 2 Eindhoven (Specialization + Research Credits + Thesis)

- **Intelligent Systems**: Year 1 Eindhoven (Basic Topics + Algorithms for Intelligent Systems + free choices). Year 2 Reykjavik (Specialization + Research Credits + Thesis)
Coordinator
Flavio Corradini

Researchers
Diletta Cacciagrano
Nicola Cannata
Rosario Culmone
Maria Rita Di Berardini
Emanuela Merelli
Luca Tesei

PhD Students
Ezio Bartocci
Francesco De Angelis
Francesca Piersigilli
Barbara Re
Oliviero Riganelli
Leonardo Vito
Roberta Alfieri
Emanuela Merelli
Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

KEGG Interaction Diagrams
KEGG Interaction Diagrams: Phosphorilation
SBML Description
Glycolysis model

UNICAM Complex Systems Research Group (CoSy)

University of Camerino

Laurea
- Computer Science

Laurea Magistrale
- Computer Science
- Bioinformatics

Thanks!

Emanuela Merelli Università di Camerino
Computational Systems Biology
Strength and weakness of the formal approaches
Multi-level modelling for systems biology
Case Study: Oxidation of carbohydrate

KEGG Interaction Diagrams
KEGG Interaction Diagrams: Phosphorilation
SBML Description
Glycolysis model
UNICAM Complex Systems Research Group (CoSy)

University of Camerino

Laurea
- Computer Science

Laurea Magistrale
- Computer Science
- Bioinformatics

Thanks!

Emanuela Merelli Università di Camerino
Agent Based Modelling and Simulation in Systems Biology: Mobile Aspects