

From Engineer to Alchemist, There and Back Again: An Alchemist Tale

Danilo Pianini – danilo.pianini@unibo.it

ALMA MATER STUDIORUM—Università di Bologna

Cesena, May 8th, 2012



1 Simulation

- Definitions
- Models
- When it's useful?

2 Alchemist

- SAPERE background
- Computational model
- Engine and architecture
- Case studies
- Performance

3 Development

- Development model
- Contributors
- Future activities



Simulation



Scientific Method

Traditional science workflow [Parisi, 2001]

- Traditional scientific method
 - identification
 - direct observation
 - theories / hypothesis
 - empirical observation
 - quantitative analysis
 - validation / invalidation



Definition of Simulation

- A new way for describing scientific theories

[Parisi, 2001]

- Simulation is the process with which we can study the dynamic evolution of a model system, usually through computational tools

[Banks, 1999]

- Simulation is the imitation of the operation of a real-world process or system over time



Simulation Requires a Model

M. Minsky – Models, Minds, Machines

A model (M) for a system (S), and an experiment (E) is anything to which E can be applied in order to answer questions about S .

- Representation / abstraction
- Formalisation
- Aggregation, Simplification, Omission

Building a model...

- How complex should be the model?
- Which assumptions should be done?



From Model to Simulation. . .

Computer simulation

- Models are designed runnable processes
- Simulation creates a **virtual laboratory**
 - controlled conditions
 - it's easy to modify the experiment (variables, parameters, etc.)
- Simulations imitate the operations of the modelled process
 - generation of an **artificial evolution** of the system

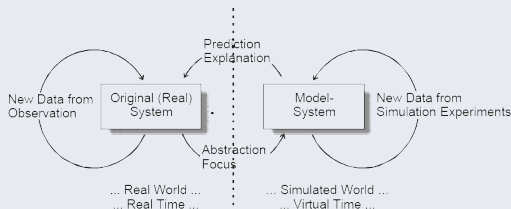


... and Back

- Deductions on the real system represented
- Evaluation of theories about the model

Model validation [Klugl and Norling, 2006]

- if the data is reliable;
- if prediction doesn't match the observed behaviour do not match
- $\Rightarrow$ the model must be revised



Why do we Need Simulations?

[Parisi, 2001, Klugl and Norling, 2006]

- The real system cannot actually be observed
- The time scale of the real system is too small or too large for observation
- The original system doesn't yet exist (or not anymore)
- The system is too complex



Alchemist



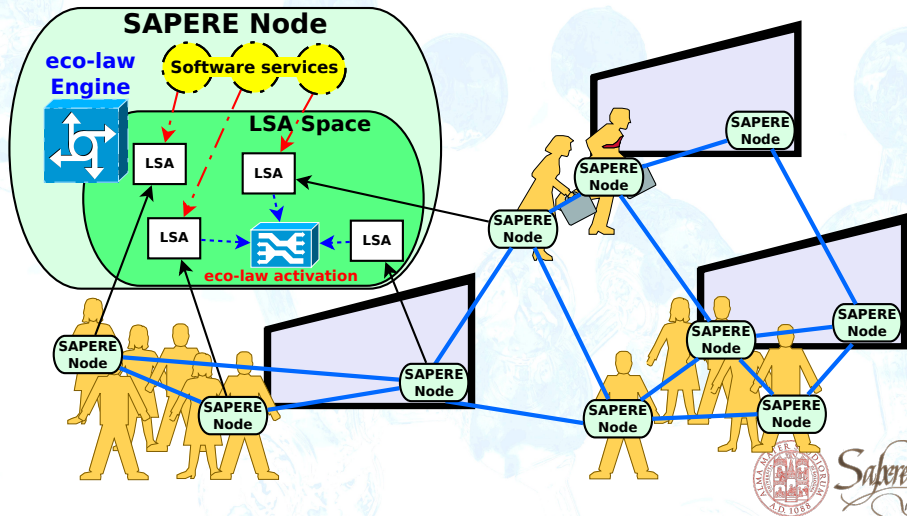
The SAPERE Project

<http://www.sapere-project.eu/>

- SAPERE (Self-aware Pervasive Service Ecosystems) is an EU STREP project under the FP7 FET Proactive Initiative: Self-Awareness in Autonomic Systems (AWARENESS)
- The objective of SAPERE is the development of a highly-innovative theoretical and practical framework for the decentralized deployment and execution of self-aware and adaptive services for future and emerging pervasive network scenarios



SAPERE World



Simulation of a SAPERE environment

The role of simulation

- Emergence cannot be fully designed
- It's crazy to deploy a whole ecosystem without any test

Fist class abstractions

- Dynamic environment
- Different, mobile, communicating nodes
- Programmability through a set of chemical-like laws
- Continuous Time Markov Chain (CTMC) model
- Autonomous agents



Two approaches

Classic ABM modelling

- High flexibility
- Topology as first-class abstraction
- Dynamics explicitly modelled
- No native support for CTMC model

Chemical inspired modelling

- Natively CTMC
- Very fast and reliable algorithms exist in literature
- Very limited topology: multicompartment at best
- Only classic reactions can change the world status: limited flexibility



Alchemist simulation approach

Base idea

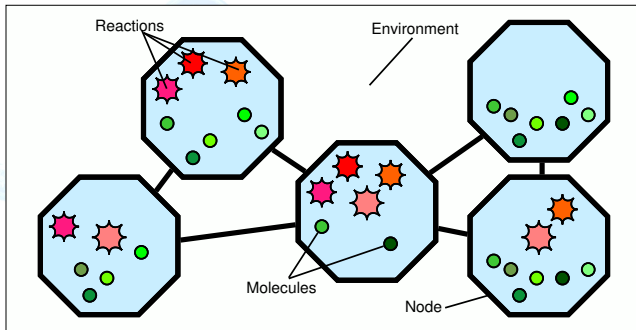
- Start from the existing work with stochastic chemical systems simulation
- Extend it as needed to reach desired flexibility

Goals

- Full support for Continuous Time Markov Chains (CTMC)
- Support for differently distributed events (e.g. Triggers)
- Rich environments, with obstacles, mobile nodes, etc.
- More expressive reactions
- Fast and flexible SSA engine



Enriching the environment description

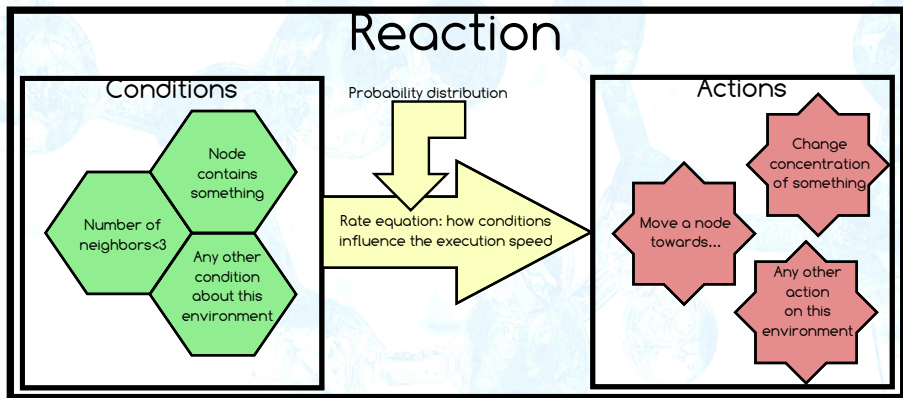


Alchemist world

- The Environment contains and links together Nodes
- Each Node is programmed with a set of Reactions
- Nodes contain Molecules
- Each Molecule in a node is described with a Concentration

Extending the concept of reaction

From a set of reactants that combine themselves in a set of products to:



In Alchemist, every event is an occurrence of a Reaction



SSA Algorithms

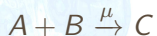
Several SSA exist, they follow the same base schema [Gillespie, 1977]:

- 1 Select next reaction using markovian rates
- 2 Execute it
- 3 Update the rates which may have changed



Do the math: reaction speed

Consider a chemical reaction in the form:



The probability that this reaction will trigger depends on:

- 1 Number of molecules A and B : the higher, the higher the probability those molecule will encounter and react
- 2 μ , a speed coefficient for the reaction

We define the propensity of a reaction as its speed in a precise instant of time as

$$a_{\mu} = \mu[A][B]$$



Do the math: next reaction choice

If we assume every reaction is a Poisson process, the probability for it to be the next one is:

$$P(\text{next} = \mu) = \int_0^\infty P(\mu, \tau) d\tau = \int_0^\infty a_\mu e^{-\tau \sum_j a_j} d\tau = \frac{a_\mu}{\sum_j a_j}$$



Do the math: next reaction time

We can also compute the next time of occurrence:

$$P(\tau)d\tau = \sum_j P(\mu, \tau)d\tau = \left(\sum_j a_j \right) e^{-\tau \sum_j a_j} d\tau$$

$$\sum_j a_j = \lambda \longrightarrow \lambda e^{-\lambda x}$$

$$F(x \leq t) = \int_{-\infty}^t \lambda e^{-\lambda x} dx = \left[-e^{-\lambda t} \right]_{-\infty}^t = e^{-\lambda t}$$

Now, given a uniformly distributed random r , it's possible to compute it's equivalent for the desired distribution:

$$e^{-\lambda t} = r \Rightarrow t = \frac{-\ln(r)}{\lambda}$$



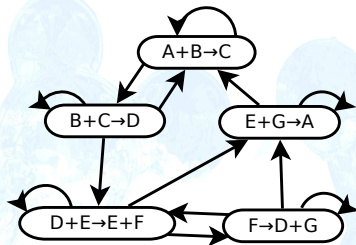
Existing algorithms

Direct method

- 1 Compute propensity for each reaction
- 2 Select next reaction probabilistically
- 3 Execute it



Existing algorithms



Direct method + Dependency graph

- 1 Compute the dependencies among reactions
- 2 Compute propensity for each reaction
- 3 Select next reaction probabilistically
- 4 Execute it
- 5 Update propensities only for potentially involved reactions
- 6 Goto 3

Existing algorithms

Next Reaction

- 1 Compute a putative execution time for each reaction
- 2 Store reactions in a binary heap
- 3 Pick the next reaction
- 4 Execute it
- 5 Compute putative times only for potentially involved reactions
- 6 Goto 3



Existing algorithms

[Slepoy et al., 2008]

- 1 Compute propensities
- 2 Split the reactions in groups by their propensity
- 3 Throw randoms until a reaction in a group is selected
- 4 Execute it
- 5 Update propensities only for potentially involved reactions
- 6 Goto 3



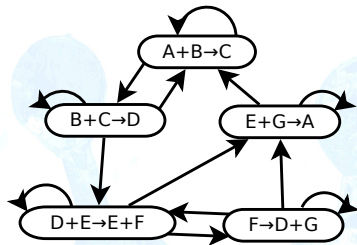
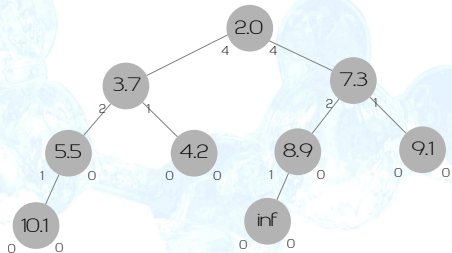
More flexibility!

What they miss is what we added

- Support for instantaneous events (∞ Markovian rate)
- Reactions can be added and removed during the simulation
- Support for non-exponential time distributed events (e.g. triggers)
- Dependencies among reactions are evaluated considering their “context”, speeding up the update phase



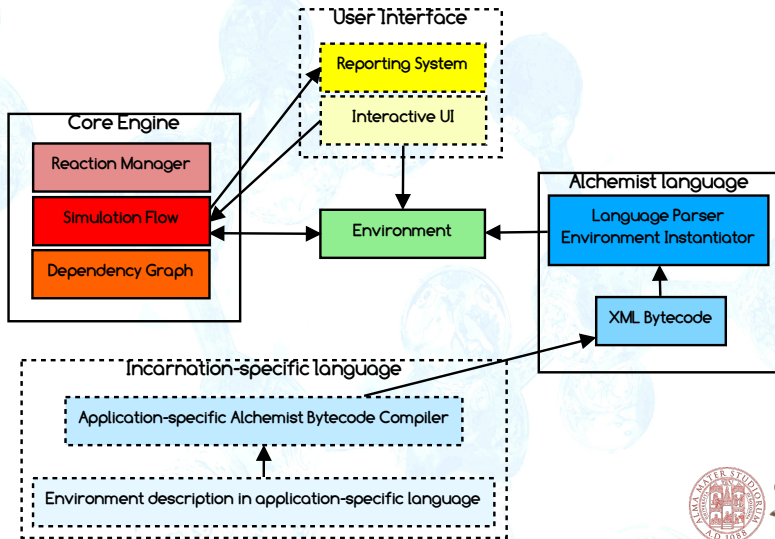
Smart data structures $\Rightarrow$ bleeding edge performances



Next Reaction efficient structures made dynamic

- Dynamic Indexed Priority Queue
 - Allow to access the next reaction to execute in $O(1)$ time
 - Worst case update in $\log_2(N)$ (average case a lot better)
 - Extended to ensure balancing with insertion and removal
- Dynamic Dependency Graph
 - Allows to smartly update only a subset of all the reaction
 - Extended with the concept of input and output context

Modular structure

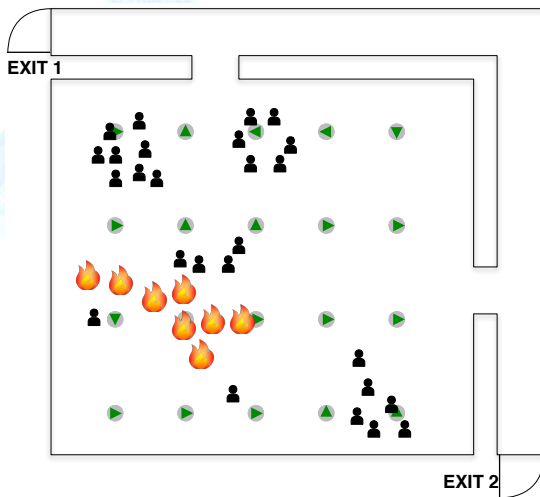


Some Features in short

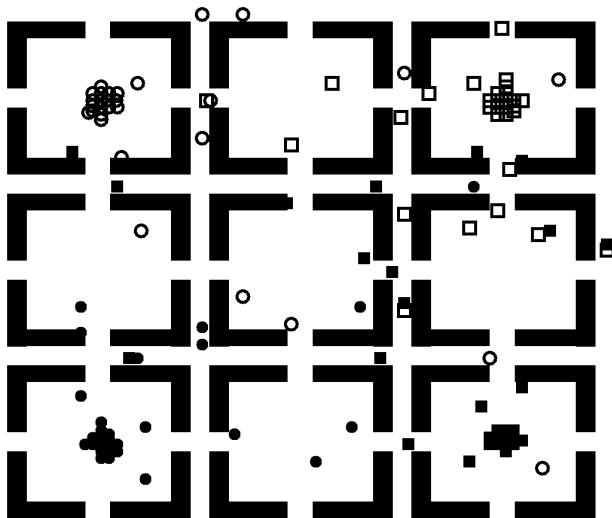
- Parallel executor
- Approximate Stochastic Model Checker
- Parallelised engine
- Alchemist2Blender
- PVeStA integration



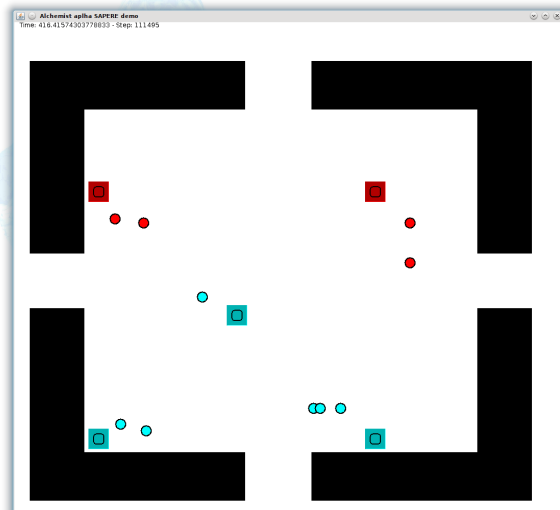
Crowd evacuation



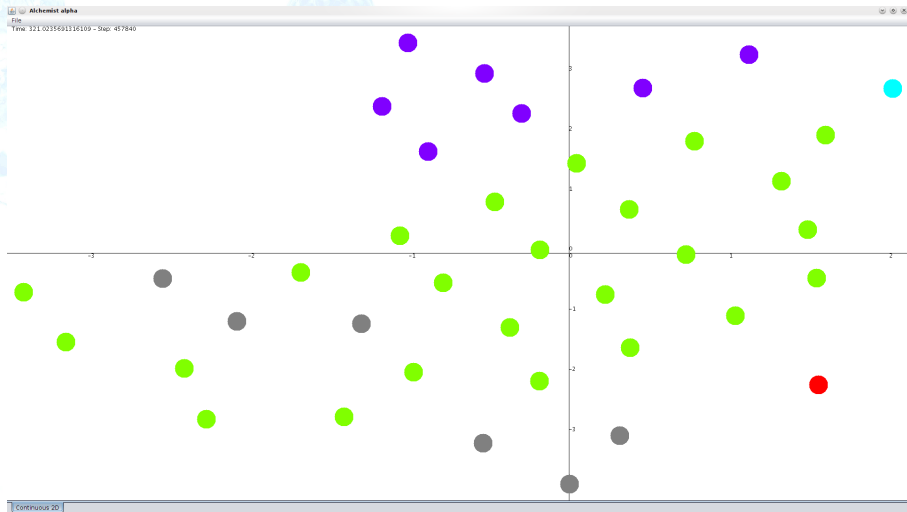
Crowd steering



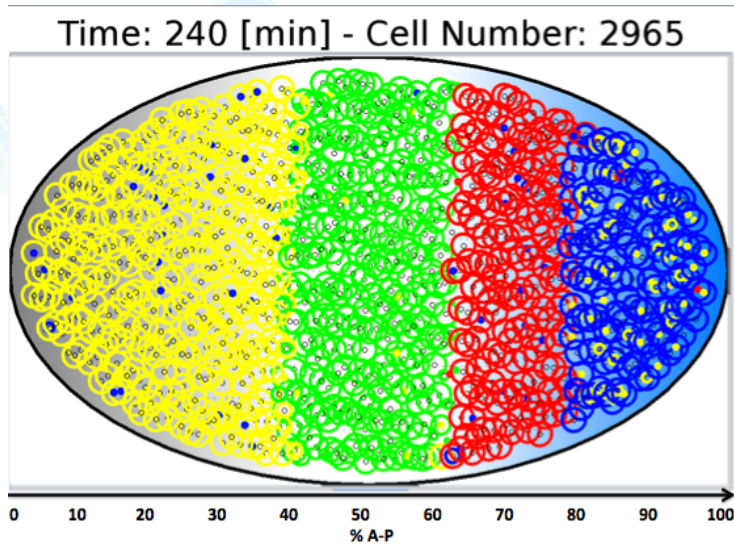
Adaptive Displays



Simple Morphogenesis proof of concept

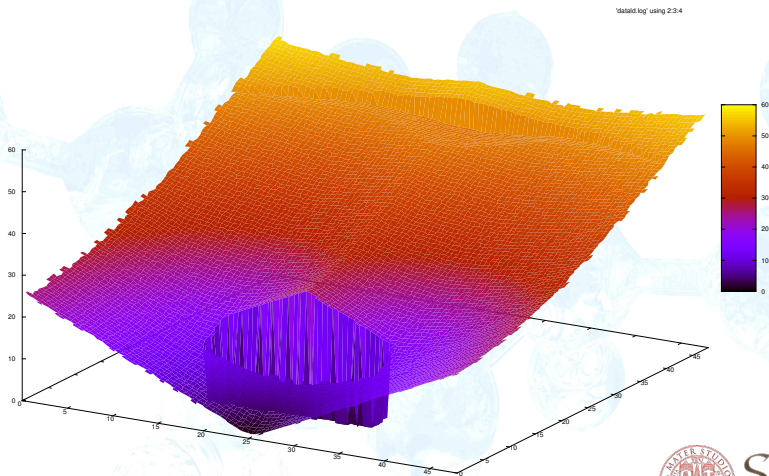


Morphogenesis of a *Drosophila Melanogaster*



Sabere
made

Anticipative adaptation



Realistic pedestrians

see the video...



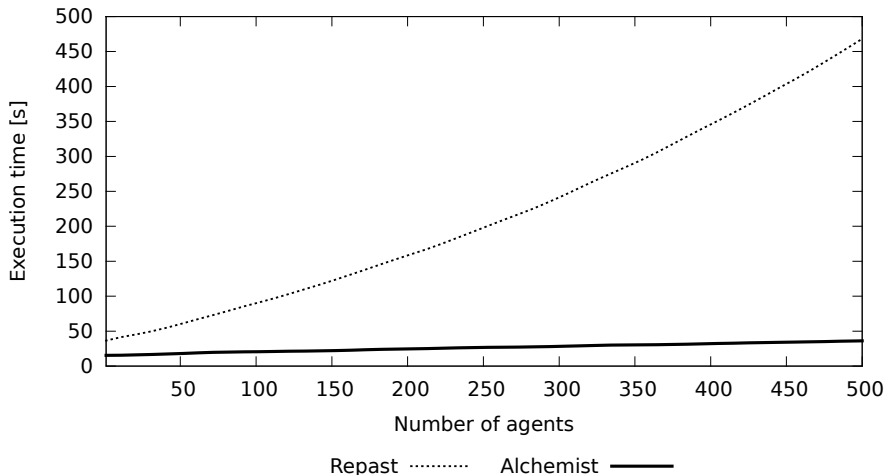
Testbed

- We realised the same crowd-steering application for both Alchemist and RePast, in order to evaluate the performance gap (if any)
- Source code for RePast and standalone Alchemist application available at
<http://www.apice.unibo.it/xwiki/bin/view/Alchemist/JOS>
- We choose a simplified use case which allows simulation in RePast without any change in its engine
 - This actually cripples out the most important Alchemist optimization for complex environment (the dependency graph)



Results

Performance comparison with Repast



Development



Distributed development

Alchemist is a training ground for some good team development practices

- Linux kernel-like development model
- Java
- XText
- Mercurial DCVS
- Bitbucket web-based code hosting
- Maven
- JUnit
- Source is released under GPL



People involved

- **Michele Bombardi** (Done)
 - Realistic pedestrians
- **Chiara Casalboni** (Done)
 - Realistic pedestrians
- **Francesca Cioffi** (Done)
 - Further experiments with Alchemist-SAPERE
- **Davide Ensini** (Done)
 - Approximate Stochastic Model Checker improvement
- **Enrico Galassi** (Done)
 - Alchemist-SAPERE high level language
- **Enrico Gualandi** (Ongoing)
 - Alchemist-SAPERE languages review



People involved

- **Luca Mella** (Done)
 - Tools for social network analysis
- **Alessandro Montalti** (Done)
 - MS-BioNet to AlchemistXML translator
- **Luca Nenni** (Done)
 - Alchemist2Blender
- **Enrico Polverelli** (Done)
 - Gradient based patterns
- **Michele Pratiffi** (Done)
 - Image importer for Alchemist
- **Giacomo Pronti** (Done)
 - SAPERE incarnation main author
- **Luca Ricci** (Ongoing)
 - Map importer for Alchemist
- **Andrea Vandin** (IMT Lucca)
 - PVeStA integration



There is a lot of work to do!

- MS-Bionet compatibility layer
- AlcheGRID
- Alchemist2Blender improvement
- OpenStreet Map importer and Google Map importer
- Blender Integration improvement
- Gnuplot integration
- CellML / SBML to AlchemistXML
- Chemistry Incarnation review
- Alchemist for Bio DSL
- Realistic biological gradients
- GPX tracks loader
- RDF to Alchemist-SAPERRE translator

These are examples, if you have something in mind, be proactive!



Bibliography I



Banks, J. (1999).

Introduction to simulation.

In Farrington, P., Nembhard, H. B., Sturrock, D. T., and Evans, G. W., editors, *Proceedings of the 1999 Winter Simulation Conference*, pages 7–13.



Gillespie, D. T. (1977).

Exact stochastic simulation of coupled chemical reactions.
Journal of Physical Chemistry, 81(25):2340–2361.



Klugl, F. and Norling, E. (2006).

Agent-based simulation: Social science simulation and beyond.
Technical report, The Eighth European Agent Systems Summer School (EASSS 2006), Annecy.



Bibliography II

 Parisi, D. (2001).
Simulazioni - La realtà rifatta al computer.
Società editrice il Mulino.

 Slepoy, A., Thompson, A. P., and Plimpton, S. J. (2008).
A constant-time kinetic Monte Carlo algorithm for simulation of large
biochemical reaction networks.
The Journal of Chemical Physics, 128(20):205101+.



From Engineer to Alchemist, There and Back Again: An Alchemist Tale

Danilo Pianini – danilo.pianini@unibo.it

ALMA MATER STUDIORUM—Università di Bologna

Cesena, May 8th, 2012

