Embryomorphomic Engineering: From biological development to artificial multi-agent organisms

René Doursat
http://www.iscpif.fr/~doursat
Systems that are self-organized and architectured

the scientific challenge of complex systems: how can they integrate a true architecture?

the engineering challenge of complicated systems: how can they integrate self-organization?

make components evolve

decompose the system

free self-organization

self-organized architecture / architectured self-organization
Toward programmable self-organization

➢ Self-organized (complex) systems
  ✓ a myriad of self-positioning, self-assembling agents
  ✓ collective order is not imposed from outside (only influenced)
  ✓ comes from purely local information & interaction around each agent
  ✓ no agent possesses the global map or goal of the system
  ✓ but every agent may contain all the rules that contribute to it

➢ Architectured systems
  ✓ true intrinsic structure: non-trivial, complicated morphology
    ▪ hierarchical, multi-scale: regions, parts, details, agents
    ▪ modular: reuse, quasi-repetition
    ▪ heterogeneous: differentiation & divergence in the repetition
  ✓ random at the microscopic level, but reproducible (quasi deterministic) at the mesoscopic and macroscopic levels
1. Toward self-organized and architecture systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
**De facto complexity of engineering (ICT) systems**

- Ineluctable breakup into myriads of modules/components,

  - In hardware,
  - In software,
  - In networks, ...

- Number of transistors/year
- Number of O/S lines of code/year
- Number of network hosts/year
Complex systems in many domains

- leads us to rethink ICT in terms of **complex systems**
  - large number of elementary **agents** interacting **locally**
  - simple individual behaviors creating a complex **emergent** collective behavior
  - **decentralized** dynamics: no master blueprint or grand architect

- physical, biological, technical, social systems (natural or artificial)
Paris Ile-de-France

4th French Complex Systems Summer School, 2010

Lyon Rhône-Alpes

National
Complex systems: a vast archipelago

**Precursor and neighboring disciplines**

- **complexity**: measuring the length to describe, time to build, or resources to run a system
  - information theory (Shannon; entropy)
  - computational complexity (P, NP)
  - Turing machines & cellular automata

- **dynamics**: behavior and activity of a system over time
  - nonlinear dynamics & chaos
  - stochastic processes
  - systems dynamics (macro variables)

- **adaptation**: change in typical functional regime of a system
  - evolutionary methods
  - genetic algorithms
  - machine learning

- **systems sciences**: holistic (non-reductionist) view on interacting parts
  - systems theory (von Bertalanffy)
  - systems engineering (design)
  - cybernetics (Wiener; goals & feedback)
  - control theory (negative feedback)

- **multitude, statistics**: large-scale properties of systems
  - graph theory & networks
  - statistical physics
  - agent-based modeling
  - distributed AI systems

→ Toward a unified “complex systems” science and engineering?
From “statistical” to “morphological” CS

Most self-organized systems form “simple”/random patterns

(a) simple/random SO: pattern formation (spots, stripes), swarms (clusters, flocks), complex networks (hubs)...

texture-like order: repetitive, statistically **uniform**, information-poor – arising from amplification of fluctuations: **unpredictable** number/position of mesoscopic entities (spots, groups) – OR determined by the environment (trails)

... while “complicated” architectures are designed by humans

(d) direct design (top-down)
From “statistical” to “morphological” ... to artificial CS

(a) natural random self-organization

- The only natural emergent and structured forms are biological
- Mesoscopic organs and limbs have intrinsic, nonrandom morphologies – development is highly reproducible in number and position of body parts – heterogeneous elements arise under information-rich genetic control

(b) natural self-organized architectures

(c) engineered self-organization (bottom-up)

(d) direct design (top-down)

→ can we reproduce them in artificial systems?
From natural CS to designed CS (and back)

The challenges of complex systems (CS) research

Transfers among systems

**CS science:** understanding & modeling "natural" CS
(spontaneously emergent, including human-made):
- morphogenesis, neural dynamics, cooperative co-evolution, swarm intelligence

Exports
- decentralization
- autonomy, homeostasis
- learning, evolution

Imports
- observe, model
- control, harness
- design, use

**CS engineering:** designing a new generation of "artificial" CS
(harnessed & tamed, including nature):
- collective robotics, synthetic biology, energy networks
The need for morphogenetic abilities

Self-architecturing in natural systems → artificial systems

- morphogenetic abilities in biological modeling
  - organism development
  - brain development

- need for morphogenetic abilities in computer science & AI
  - self-forming robot swarm
  - self-architecturing software
  - self-connecting micro-components

- need for morphogenetic abilities in techno-social networked systems
  - self-reconfiguring manufacturing plant
  - self-stabilizing energy grid
  - self-deploying emergency taskforce

http://www.symbrian.eu

MAST agents, Rockwell Automation Research Center
{pvrba, vmarik}@ra.rockwell.com
1. Toward self-organized and architectured systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
From “statistical” to “morphological” CS in social insect constructions

*more intrinsic, sophisticated architecture*

- ant trail
- network of ant trails
- ant nest
- termite mound
From “statistical” to “morphological” CS in inert matter / insect constructions / multicellular organisms

physical pattern formation

ant trail

network of ant trails

social insect constructions

ant nest

termite mound

more intrinsic, sophisticated architecture

grains of sand + air

insects

new inspiration

biological morphogenesis

cells
Morphological (self-dissimilar) systems
compositional systems: pattern formation ≠ morphogenesis

“*I have the stripes, but where is the zebra?*” OR
“The stripes are easy, it’s the horse part that troubles me”
—attributed to A. Turing, after his 1952 paper on morphogenesis
From “statistical” to “morphological” CS

Physical pattern formation is “free” – Biological (multicellular) pattern formation is “guided”
From “statistical” to “morphological” CS

Multicellular forms = a bit of “free” + a lot of “guided”

- domains of free patterning embedded in a guided morphology
  - unlike Drosophila’s stripes, these pattern primitives are not regulated by different sets of genes depending on their position
  - spots, stripes in skin

- repeated copies of a guided form, distributed in free patterns
  - entire structures (flowers, segments) can become modules showing up in random positions and/or numbers

- flowers in tree
- segments in insect

ommatidia in compound eye
dragonfly, www.phy.duke.edu/~hsg/54

angelfish, www.sheddaquarium.org

cherry tree, www.phy.duke.edu/~fortney

images.encarta.msn.com
**Evo-Devo Engineering**

- **Model embryogenesis** → **export to engineering**
  - automated **observation** and reconstruction of developing organisms by image processing and learning/optimization methods
  - mathematical and computational (agent-based) **modeling**
  - **simulation** of recalculated embryos, real and fictitious

---

*FP6 Projects Embryomics, BioEmergences*
*Submitted ANR Projects MEC@GEN, SYNBIOTIC*
Overview of morphogenesis

- An abstract computational approach to development
  - as a fundamentally spatial phenomenon
  - highlighting its broad principles and proposing a computational model of these principles

- Broad principles
  1. biomechanics → collective motion → "sculpture" of the embryo
  2. gene regulation → gene expression patterns → "painting" of the embryo
    + coupling between shapes and colors

- Multi-agent models
  - best positioned to integrate both
  - account for heterogeneity, modularity, hierarchy
  - each agent carries a combined set of biomechanical and regulatory rules
Embryogenesis couples assembly and patterning

Sculpture → forms

"shape from patterning"

✓ the forms are "sculpted" by the self-assembly of the elements, whose behavior is triggered by the colors

Painting → colors

"patterns from shaping"

✓ new color regions appear (domains of genetic expression) triggered by deformations
Embryogenesis couples assembly and pattern formation.

SA = self-assembly ("sculpture")
PF = pattern formation ("painting")

(a) $\alpha$
(b) $\alpha$
(c) $\alpha_1, \alpha_2, \ldots, \alpha_3$
(d) $\alpha_3$
(e) $\alpha_{3,1}, \alpha_{3,2}, \alpha_{3,3}$
(f) $\alpha_{3,3}$

Genotype

Embryogenesis couples assembly and pattern formation.
Embryogenesis couples **mechanics** and **regulation**

### Cellular mechanics
- adhesion
- deformation / reformation
- migration (motility)
- division / death

### Genetic regulation

**Drosophila embryo**

---

Gene regulatory pattern formation

- Segmentation & identity domains in *Drosophila*
  - periodic A/P band patterns are controlled by a 5-tier gene regulatory hierarchy
  - intersection with other axes creates organ primordia and imaginal discs (identity domains of future legs, wings, antennae, etc.)

from Carroll, S. B., et al. (2001) *From DNA to Diversity*, p63
Embryogenesis couples mechanics and regulation

- **Cellular mechanics**
  - modification of cell size and shape
  - mechanical stress, mechano-sensitivity
  - differential adhesion
  - growth, division, apoptosis

- **Genetic regulation**
  - gene regulation
  - change of cell-to-cell contacts
  - diffusion gradients ("morphogens")
  - change of signals, chemical messengers
Embryogenesis couples **motion** and **patterns**

Collective motion **regionalized into patterns**

- Nadine Peyriéras, Paul Bourgine, Thierry Savy, Benoît Lombardot, Emmanuel Faure et al.
- Embryomics & BioEmergences
- zebrafish
- Hiroki Sayama (Swarm Chemistry)
  - http://bingeweb.binghamton.edu/~sayama/
  - SwarmChemistry/

Pattern formation that triggers motion

- http://zool33.uni-graz.at/schmickl
- Doursat
1. Toward self-organized and architectured systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
Overview of an embryomorphomic system

Recursive morphogenesis

genotype

grad₁

patt₁

div₂

patt₂

div₁

patt₃

div₃

grad₂

grad₃
$G_{SA}: r_c < r_e = 1 \ll r_0$
$p = 0.05$
patt

\[ G_{PF}: \{ w \} \]
div

\[ G_{SA} : r_c < r_e = 1 \ll r_0 \]
\[ p = 0.05 \]

grad

\[ \nabla \]

patt

\[ G_{PF} : \{w\} \]

\[ G_{SA} \cup G_{PF} \]
Virtual gene atlas

Programmed patterning (patt): the hidden embryo map

a) same swarm in different colormaps to visualize the agents’ internal patterning variables $X$, $Y$, $B_i$ and $I_k$ (virtual in situ hybridization)

b) consolidated view of all identity regions $I_k$ for $k = 1...9$

c) gene regulatory network used by each agent to calculate its expression levels, here: $B_1 = \sigma(1/3 - X)$, $B_3 = \sigma(2/3 - Y)$, $I_4 = B_1B_3(1 - B_4)$, etc.
Hierarchical embryogenesis

- **Morphological refinement by iterative growth**
  
  - details are not created in one shot, but gradually added...
  
  - ...while, at the same time, the canvas grows

  ![Image of iterative growth process](image)

Hierarchical embryogenesis

$r_c = .8, r_c = 1, r_0 = \infty$

$G_{SA} = r'_c = r'_0 = 1, p = .01$
Hierarchical embryogenesis

- All cells have the same GRN but execute different expression paths → determination / differentiation

- Microscopic (cell) randomness, but mesoscopic (region) predictability

\[ r_c = 0.8, r_e = 1, r_0 = \infty \]
\[ G_{SA} r'_c = r'_0 = 1, p = 0.01 \]
1. Toward self-organized and architectured systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
Evolutionary innovation by development

Development: the missing link of the Modern Synthesis...

"When Charles Darwin proposed his theory of evolution by variation and selection, explaining selection was his great achievement. He could not explain variation. That was Darwin’s dilemma."

"To understand novelty in evolution, we need to understand organisms down to their individual building blocks, down to their deepest components, for these are what undergo change."

—Marc W. Kirschner and John C. Gerhart (2005)
The Plausibility of Life, p. ix
The self-made puzzle of “evo-devo” engineering

Development: the missing link of the Modern Synthesis...

“To understand novelty in evolution, we need to understand organisms down to their individual building blocks, down to their deepest components, for these are what undergo change.”
The self-made puzzle of “evo-devo” engineering

Development: the missing link of the Modern Synthesis...

Genotype  \[ \approx \]  \[ \approx \]  Phenotype

- more or less direct representation
- generic elementary rules of self-assembly
- macroscopic, emergent level
- microscopic, componential level

\[ \text{genotype} \rightarrow \text{“Transformation”} \rightarrow \text{phenotype} \]
Toward “evo-devo” engineering

... and of Evolutionary Computation: toward “meta-design”

- organisms endogenously grow but artificial systems are built exogenously

Could engineers “step back” from their creation and only set generic conditions for systems to self-assemble?

Instead of building the system from the top (phenotype), program the components from the bottom (genotype)
The meta-design of complexity

Pushing design toward evolutionary biology

intelligent “hands-on” design
- heteronomous order
- centralised control
- designer as a micromanager
- rigidly placing components
- sensitive to part failures
- need to control and redesign

complicated systems: planes, computers

intelligent & evolutionary “meta-design”
- autonomous order
- decentralised control
- designer as a lawmaker
- allowing fuzzy self-placement
- insensitive to part failures
- prepare for adaptation & evolution
- complex multi-component systems
The evolutionary “self-made puzzle” paradigm

- Construe systems as self-assembling (developing) puzzles
- Design and program their pieces (the “genotype”)
- Let them evolve by variation of the pieces and selection of the architecture (the “phenotype”)

Genotype: rules at the micro level of agents
- ability to search and connect to other agents
- ability to interact with them over those connections
- ability to modify one’s internal state (differentiate) and rules (evolve)
- ability to provide a specialized local function

Phenotype: collective behavior, visible at the macro level
The evolutionary “self-made puzzle” paradigm

a. Construe systems as **self-assembling** (developing) **puzzles**

b. Design and **program their pieces** (the “genotype”)

c. Let them evolve by **variation of the pieces** and **selection** of the architecture (the “phenotype”)
Multi-agent evolutionary development (evo-devo)

Quantitative mutations: limb thickness

![Diagram showing wild type, thin-limb, and thick-limb interfaces with corresponding quantitative changes in limb thickness.](Diagram.png)
Multi-agent evolutionary development (evo-devo)

Quantitative mutations: body size and limb length

(a) small

(b) long-limb

(c) short-limb

\[
\begin{array}{c|c|c|c}
G_{PF} & G_{SA} & G_{PF} & G_{SA} \\
\hline
1 \times 1 & 1 \times 1 & 3 \times 3 & 3 \times 3 \\
\text{tip} & \text{tip} & \text{disc} & \text{disc} \\
p = .05 & g = .05 & p = .05 & g = .05 \\
g' = 8 & g' = 40 & g = 8 & g = 15 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
G_{PF} & G_{SA} & G_{PF} & G_{SA} \\
\hline
1 \times 1 & 1 \times 1 & 3 \times 3 & 3 \times 3 \\
\text{tip} & \text{tip} & \text{disc} & \text{disc} \\
p = .05 & p = .05 & p = .05 & p = .05 \\
g' = 8 & g' = 10 & g = 8 & g = 15 \\
\end{array}
\]
Qualitative mutations: limb position and differentiation

- **antennapedia**
  - **homology** by *duplication*
  - **divergence** of the homology

(a) antennapedia
(b) duplication (three-limb)
(c) divergence (short & long-limb)
Multi-agent evolutionary development (evo-devo)

- Qualitative mutations: number of limbs
Multi-agent evolutionary development (evo-devo)

➢ Qualitative mutations: 3rd-level digits

(a) (b) (c)
Multi-agent evolutionary development (evo-devo)

- Artificial phylogenetic tree

production of structural innovation
Work toward more accurate biological modeling

- More accurate mechanics
  - 3-D
  - Individual cell shapes
  - Collective motion, migration
  - Adhesion

- Better gene regulation
  - Recurrent links
  - Gene reuse
  - Kinetic reaction ODEs
  - Attractor dynamics

More work toward functional EC

What is missing...

1. the *function/purpose/behavior* of a developed organism
   - depending on the problem domain
   - 2-D/3-D modular robotics: move, grab, build, etc.
   - N-D networks: communication dynamics, collective computation

2. a *fitness measure*
   - assessing the value of the above function

3. a *systematic exploration*
   - by random, automated mutations
   - with statistics over many runs

4. a *comparison*
   - with other, non-developmental (or non-self-organized) approaches
   - on the same problems or benchmarks
Discussion

- **Questions that need to be addressed...**

  - **modularity?**
    - modularity of the genotype vs. phenotype
  
  - **compactness?**
    - repetitiveness: reuse of genes and gene regulation modules
    - vs. heterogeneity and uniqueness of structures
  
  - **innovation?**
    - how fine-grained development fosters the emergence of new structures
  
  - **open-ended evolution?**
    - don’t set a specific goal, harvest from surprising organisms
1. Toward self-organized and architectured systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
Programmable techno-social networks

Harnessing complex networks

ubiquitous computing & communication capabilities create entirely new myriads of user-device interactions from the bottom up

explosion in size and complexity of techno-social networks in all domains: energy, education, healthcare, business, defense

de facto complex systems with spontaneous collective behavior that we don’t quite understand or control yet

time to design new collaborative technologies to harness this decentralisation and emergence
From "scale-free" to architectured networks

- single-node composite branching
- iterative lattice pile-up
- clustered composite branching
Self-knitting networks

- Not random, but *programmable* attachment

- a generalisation of embryogenesis in $n$ dimensions

- the node routines are the "genotype" of the network
Order influenced (not imposed) by the environment
Abstract model of self-made network

- **Formation of a specific, reproducible structure**
  - nodes attach randomly, but only to a few available ports

1. Chains
2. Lattices
3. Clusters
4. Modules
Abstract model of self-made network

- **Simple chaining**
  - link creation ($L$) by programmed port management ($P$)

![Diagram](image.png)

- Ports can be "occupied" or "free", "open" or "closed"
Abstract model of self-made network

- **Simple chaining**
  - port management \((P)\) relies on gradient update \((G)\)
  - each node executes \(G, P, L\) in a loop
  - \(P\) contains the logic of programmed attachment

\[
G \xrightarrow{} P \xrightarrow{} L
\]

if \((x + x' == 4)\) {
  close \(X, X'\)
} else {
  open \(X, X'\)
}

\(x \leftrightarrow x'\)

- Each node executes \(G, P, L\) in a loop.
- \(P\) contains the logic of programmed attachment.
Abstract model of self-made network

- Simple chaining
Abstract model of self-made network

- **Lattice formation by guided attachment**
  - two pairs of ports: \((X, X')\) and \((Y, Y')\)
  - without port management \(P\), chains form and intersect randomly
Abstract model of self-made network

- **Lattice formation by guided attachment**
  - only specific spots are open, similar to beacons on a landing runway

```
if (x == 0 or (x > 0 & Y'(x-1, y) is occupied))
  { open X' }
else
  { close X' }
```

\[ Y \]

lattice growing in waves
Cluster chains and lattices

- several nodes per location: reintroducing randomness but only within the constraints of a specific structure

Abstract model of self-made network
Abstract model of self-made network

- Cluster chains and lattices
Abstract model of self-made network

Modular structures by local gradients

- modeled here by different coordinate systems, \((X_a, X'_a), (X_b, X'_b), \text{ etc., and links cannot be created different tags}\)
Abstract model of self-made network

- Modular structures by local gradients

✓ the node routines are the “genotype” of the network

```plaintext
close Xa
if (xa == 2) { create Xb, X'b }
if (xa == 4) { create Xc, X’c }
if (xa == 5) { close X’a } else { open X’a }
close Xb
if (xb == 2) { close X’b } else { open X’b }
close Xc
if (xc == 3) { close X’c } else { open X’c }
```

...
Self-organized programmable networks

- **strong intrinsic morphology** – no influence from the environment
- **no intrinsic morphology** – complete adaptation to the environment

- **intrinsic morphologies** that are non-trivial *and* adapt dynamically to their environment
Development

Polymorphism

Evolution

“wildtype” ruleset A

ruleset A

(b)

ruleset A'

(b)

ruleset A'
Spatial Computers

Robot Swarms

Biological Computing

Sensor Networks

Reconfigurable Computing

Cells during Morphogenesis

Modular Robotics

Slide from Jake Beal’s course on Spatial Computing, 2009 (CSAIL, MIT)
Evo-Devo Engineering

Methodologies and tools

- an original, young field of investigation without a strong theoretical framework yet – but close links with many established disciplines, which can give it a more formal structure through their own tools
  - cellular automata, pattern formation
  - collective motion, swarm intelligence (Ant Colony Optim. [Dorigo])
  - gene regulatory networks: coupled dynamical systems, attractors
  - spatial computing languages: PROTO [Beal] and MGS [Giavitto]
    (top-down compilation)
  - evolution: genetic algorithms, computational evolution [Banzhaf]
  - Iterative Function Systems (IFS) [Lutton]

→ goal: going beyond agent-based experiments and find an abstract description on a macroscopic level, for better control and proof
http://iridia.ulb.ac.be/ants2010
→ Special Session on Morphogenetic Engineering

Exploring various engineering approaches to the artificial design and implementation of autonomous systems capable of developing complex, heterogeneous morphologies

Thank you
1. Toward self-organized and architectured systems

2. Biological development as a two-side challenge
   Heterogeneous motion vs. moving patterns

3. Embryomorphic engineering
   Embryogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies