Embryomorphic Engineering:
From biological development to self-organized computational architectures

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

- free self-organization

- (evolutionary) design

- self-organized architecture / architectured self-organization

- make components evolve

- decompose the system

- 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?
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
# Embryomorph Engineering

<table>
<thead>
<tr>
<th>1.</th>
<th>Toward self-organized <em>and</em> architectured systems</th>
</tr>
</thead>
</table>
| 2. | Biological development as a two-side challenge  
Heterogeneous motion vs. moving patterns |
| 3. | Embryomorphic engineering  
Morphogenesis as a multi-agent self-assembly process |
| 4. | Evo-devo engineering  
Evolutionary innovation by development |
| 5. | Extension to self-knitting network topologies |
Complex systems in many domains

- 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)

- **pattern formation**
  - ○ = matter
- **biological development**
  - ○ = cell
- **the brain & cognition**
  - ○ = neuron
- **insect colonies**
  - ○ = ant
- **Internet & Web**
  - ○ = host/page
- **social networks**
  - ○ = person
CS in geographical space (F)

Paris Ile-de-France

National

Lyon Rhône-Alpes

4th French Complex Systems Summer School, 2010
CS in conceptual space: 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?
Emergence

- The system has properties that the elements do not have
  - ex: micro units form macro patterns: rolls, spiral waves, stripes, spots
  - ex: “ignorant” individuals make intelligent collective decisions: insect colonies, neurons, market traders (?)

- These properties cannot be easily inferred or deduced
  - ex: liquid water or ice emerging from H₂O molecules
  - ex: cognition and consciousness emerging from neurons

- Different properties can emerge from the same elements/rules
  - ex: the same molecules of water combine to form liquid or ice crystals
  - ex: the same cellular automaton rules change behavior with initial state

- Global properties make local (sophisticated) rules at a higher level
  → jumping from level to level through emergence

- Counter-examples of emergence without self-organization
  - ex: well-informed leader (orchestra conductor, military officer)
  - ex: global plan (construction area), full instructions (orchestra)
From natural to engineered emergence (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
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?

MESOBIONICS

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?

MESOBIONICS
De facto complexity of engineering (ICT) systems

- Ineluctable breakup into myriads of modules/components,
  - in hardware,
  - software,
  - or networks, ...

- number of transistors/year
- number of O/S lines of code/year
- number of network hosts/year
Taming complex ICT toward 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.symbion.eu

MAST agents, Rockwell Automation Research Center
{pvrba, vmarik}@ra.rockwell.com
Toward “evo-devo” engineering

➢ From design to “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 design (ID)
- intelligent meta-design (IMD)
- evolutionary meta-design (EMD)
- undesigned evolution (UE)

100% design: the system stays more or less under the control of a designer
0% design

engineering → 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

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”)

- **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”)

![Diagram of self-made puzzle evolution](image)
1. Toward self-organized and architectured systems

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

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

4. Evo-devo engineering  
   Evolutionary innovation by development

5. Extension to self-knitting network topologies
Devo-inspired engineering

Replacing biological development in the center

- Computational, spatially explicit models of development and evolution with possible outcomes toward hyperdistributed, decentralized engineering systems

DEVO  PROGNET  EVOSPACE
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

more intrinsic, sophisticated architecture

network of ant trails

social insect constructions

ant trail

ant nest

termite mound

grains of sand + air

insects

new inspiration

biological morphogenesis

cells
Morphological (self-dissimilar) systems
compositional systems: pattern formation $\neq$ morphogenesis

“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
angelfish, www.sheddaquarium.org

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

segments in insect
centipede, images.encarta.msn.com
Evo-devo engineering

- **Model embryogenesis ↔ import/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
Morphogenesis 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

---

Niki de Saint Phalle, "patterns from shaping"  
Ádám Szabó, *The chicken or the egg* (2005)  
http://www.szaboadam.hu
Morphogenesis couples **assembly** and **patterning**

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

geno*type*
Morphogenesis couples mechanics and regulation

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

- **Genetic regulation**

  after Carroll, S. B. (2005)
  *Endless Forms Most Beautiful*, p117
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
Morphogenesis couples **mechanics and regulation**

- **Cellular mechanics**
  - modification of cell size and shape
  - mechanical stress, mechano-sensitivity
  - growth, division, apoptosis
  - change of cell-to-cell contacts
  - change of signals, chemical messengers

- **Genetic regulation**
  - gene regulation
  - diffusion gradients ("morphogens")
Morphogenesis couples **motion** and **patterns**

Collective motion **regionalized into patterns**

Pattern formation that triggers **motion**

Nadine Peyriéras, Paul Bourgine, Thierry Savy, Benoît Lombardot, Emmanuel Faure et al.

Embryomics & BioEmergences

Hiroki Sayama (Swarm Chemistry)
[http://bingeweb.binghamton.edu/~sayama/](http://bingeweb.binghamton.edu/~sayama/)

Doursat
1. Toward self-organized *and* architectured systems

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

3. Embryomorphic engineering
   Morphogenesis 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 embryomorphemic system

Recursive morphogenesis

genotype

Grad1

Div1

Grad2

Div2

Grad3

Div3

Patt1

Patt2

Patt3
$G_{SA}: \ r_c < r_e = 1 << r_0 \ 
p = 0.05$
\( G_{PF} : \{ w \} \)
$$G_{SA}: r_c < r_e = 1 << r_0$$
$$p = 0.05$$

$$G_{PF}: \{w\}$$
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 $k = 1\ldots9$
c) gene regulatory network used by each agent to calculate its expression levels, here: $B_1 = \sigma(1/3 - X)$, $B_3 = \sigma(2/3 - Y)$, $I_4 = B_1 B_3 (1 - B_4)$, etc.
Hierarchical morphogenesis

➤ Morphological refinement by iterative growth

✓ details are not created in one shot, but gradually added. . .

✓ . . . while, at the same time, the canvas grows

from Coen, E. (2000) 
The Art of Genes, pp131-135
Hierarchical embryogenesis

\[ r_c = 0.8, r_e = 1, r_0 = \infty \]
\[ G_{SA} = r'_e = r'_0 = 1, p = 0.01 \]
Hierarchical embryogenesis

all cells have same GRN, but execute different expression paths $\rightarrow$ determination / differentiation

microscopic (cell) randomness, but mesoscopic (region) predictability

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

$G_{SA} r' = r'_0 = 1, p = .01$
1. Toward self-organized *and* architectured systems

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

3. Embryomorphic engineering
   Morphogenesis 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 \[ \approx \text{“Transformation”} \approx \text{Phenotype} \]

- more or less direct representation
- generic elementary rules of self-assembly
- macroscopic, emergent level
- microscopic, componential level
Multi-agent evolutionary development (evo-devo)

Quantitative mutations: limb thickness

(a) (b) (c)

Wild type thin-limb thick-limb

Module

Limb module

Body plan module

$G_{PF}$ $G_{SA}$

$1 \times 1$

$3 \times 3$

$G_{PF}$ $G_{SA}$

tip $p' = 0.05$

g' = 15

disc $p = 0.05$

g = 15

$p = 0.05$

g = 15

GPF GSA 3

1, 1

4 6

GPF GSA 3

0.5, 1

4 6

GPF GSA 3

2, 1

4 6

GPF GSA 3

1, 1

4 6

GPF GSA 3

1, 1

4 6

GPF GSA 3

1, 1

4 6
Multi-agent evolutionary development (evo-devo)

Quantitative mutations: body size and limb length
Multi-agent evolutionary development (evo-devo)

- Qualitative mutations: limb position and differentiation

(a) antennapedia

(b) homology by duplication

divergence of the homology

(c) divergence (short & long-limb)

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

\[
\begin{array}{|c|c|}
\hline
G_{PF} & 3 \times 3 \\
G_{SA} & \text{disc } p = .05 \\
\hline
\end{array}
\]

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

\[
\begin{array}{|c|c|}
\hline
G_{PF} & 3 \times 3 \\
G_{SA} & \text{disc } p = .05 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
PF & 1 \times 1 \\
SA & \text{tip } p' = .05 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
PF & 1 \times 1 \\
SA & \text{tip } p' = .03 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
PF & 1 \times 1 \\
SA & \text{tip } p' = .1 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
G_{PF} & 3 \times 3 \\
G_{SA} & \text{disc } p = .05 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
G_{PF} & 3 \times 3 \\
G_{SA} & \text{disc } p = .05 \\
\hline
\end{array}
\]
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
   Morphogenesis 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 morphogenesis in *n* dimensions
✓ the node routines are the "genotype" of the network
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 \textit{and} adapt dynamically to their environment
Order influenced (not imposed) by the environment
Polymorphism

Evolution

“wildtype” ruleset A

ruleset A

(b) ruleset A’

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)
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
Details: an 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 ($P$)

Ports can be "occupied" or "free", "open" or "closed"

```
port X 0 0 port X' x x'
```

```
$t = 0$
```

```
$t = 1$
```

```
$t = 2$
```

```
$t = 3$
```

```
$t = 4$
```

"slower" link creation

"fast" gradient update

```
t = 0.0
```

```
t = 2.0
```

```
t = 2.1
```

```
t = 2.2
```

```
t = 2.3
```

```
t = 3.0
```

```
t = 3.0
```

```
t = 4.0
```

```
t = 4.0
```

```
t = 4.0
```
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

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

  $G \rightarrow P \rightarrow L$

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

  each node executes $G$, $P$, $L$ in a loop
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

\[
\begin{align*}
\text{if (} x == 0 \text{ or } \\
( x > 0 \text{ and } Y'(x-1, y) \text{ is occupied}) \\
\{ \text{open } X' \} \\
\text{else } \{ \text{close } X' \}
\end{align*}
\]

- lattice growing in waves
Abstract model of self-made network

- **Cluster chains and lattices**
  - several nodes per location: reintroducing randomness but only within the constraints of a specific structure

![Diagram of cluster chains and lattices](image)
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, \((X_a, X'_a), (X_b, X'_b)\), etc., and links cannot be created different tags

---

Diagram showing the connections between nodes labeled with \(X_a\), \(X'_a\), \(X_b\), and \(X'_b\) with arrows indicating the direction of connectivity.
Abstract model of self-made network

- Modular structures by local gradients

✓ the node routines are the "genotype" of the network

```
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 }
```

![Diagram of a network with nodes and edges labeled with numbers and arrows indicating the flow of information.]
Morphogenetic Engineering, ANTS 2010, Brussels

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

+ Springer book in preparation

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
   Morphogenesis as a multi-agent self-assembly process

4. Evo-devo engineering
   Evolutionary innovation by development

5. Extension to self-knitting network topologies