The Emergence of Pathological Constructors during von Neumann Self Reproduction

Declan Baugh
The Rince Institute, Dublin City University

NCFrontiers Workshop, 10th September 2012
The EvoSym Project

- Emergence and Evolution of Biological Symbol Systems
  - Evolution of a Genotype-Phenotype mapping

- Theory of Self-Reproducing Automata
  - John von Neumann & Arthur Burks, 1966
Von Neumann Self Reproducing Architecture

- An arbitrarily complex machine $M$, must be composed of two components, $P + G$.
  - $P = (A + B + C + D)$
    - $A =$ Programmable Constructor
    - $B =$ Copier
    - $C =$ Control
    - $D =$ “Ancillary” Machinery
  - $G = \Phi(A + B + C + D)$
    - Description Tape
Von Neumann Self Reproducing Architecture

M: An “arbitrarily complex” machine

D: “Ancillary” Machinery
A: Programmable Constructor
C: Control
B: Copier

G: “Description” Tape

M: \[ P + \Phi(P) = (A+B+C+D) + \Phi(A+B+C+D) \]
Implementation in Cellular Automata
The Tierra Platform

- Tom Ray – 1991
  - 1D circular core memory
  - Small assembler language instruction set
  - Template addressing
  - Memory protection
  - Random perturbations
The Tierra Platform
The Tierra Platform
Implementation within Tierra

**Standard Tierra self copier:** \((B + C)\)

<table>
<thead>
<tr>
<th>Address Template</th>
<th>(Phenotype/Genotype)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C) Self Inspection and Offspring Allocation</td>
<td>(B) Phenotype/Genotype Copying and Offspring Construction</td>
</tr>
</tbody>
</table>

**Von Neumann style self reproducer:** \((A + B + C) + \Phi(A + B + C)\)

<table>
<thead>
<tr>
<th>Address Template</th>
<th>(Phenotype)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C) Self Inspection and Offspring Allocation</td>
<td>(A) Genotype-Encoding and Offspring Phenotype Construction</td>
</tr>
<tr>
<td>(B) Genotype-Copying and Offspring Genotype Construction</td>
<td></td>
</tr>
<tr>
<td>(A) Look-Up Table</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address Template</th>
<th>(Genotype)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine Description</td>
<td></td>
</tr>
</tbody>
</table>

| Address Template | |
|------------------||
Results

- Catastrophic ecosystem collapse
  - Emergence of pathological constructors
Results

- Catastrophic ecosystem collapse
  - Emergence of pathological constructors

| $X$   | $\Phi(X)$ |
Results

- Catastrophic ecosystem collapse
  - Emergence of pathological constructors

\[
\begin{array}{c|c}
X & \Phi(X) \\
\hline
X & \Phi(X') \\
\end{array}
\]
Results

- Catastrophic ecosystem collapse
  - Emergence of pathological constructors

\[
\begin{array}{c|c}
X & \Phi(X) \\
\hline
X' & \Phi(X') \\
\hline
X' & \Phi(X') \\
X' & \Phi(X') \\
X' & \Phi(X') \\
\end{array}
\]
Results

• Catastrophic ecosystem collapse
  - Emergence of pathological constructors

<table>
<thead>
<tr>
<th>X</th>
<th>Φ(X)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X</th>
<th>Φ(X')</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X'</th>
<th>Φ(X')</th>
<th>X'</th>
<th>Φ(X')</th>
<th>X'</th>
<th>Φ(X')</th>
<th>X'</th>
<th>Φ(X')</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- But that shouldn't be a problem!
Results

- Features which effect the emergence of pathological constructors.
  - Frequency of segment deletions.
  - Separation of genotype/phenotype.
  - Bug in Tierra code!
Future work

- Fix bug in Tierra
- Actually investigate our research problem
References

- Baugh D. (2012). The Emergence of Pathological Constructors when Implementing the Von Neumann Architecture for Self-Reproduction in Tierra
- Golly version 2.3  http://golly.sourceforge.net/
Thank You

Questions?