Evolutionary Robotics

Jeff Clune
Assistant Professor
Evolving Artificial Intelligence Laboratory

University of Wyoming
Update

• Doing great on the homeworks
  • 86% got 100! Rest 95 or 90
  • and people say my classes are hard!
  • (remember that when it does get hard!)

• Warning:
  • homeworks start to get harder.
  • start early

• Seize the Opportunity!
  • You may not realize it now, but this is a golden opportunity
Outline

• Background

• Experiments
  • Part I: regularity
    - generative exploits problem regularity to outperform direct
      - produces regular behaviors and brains
      - more evolvable
    - bias towards regularity can be harmful
      - on problems with irregularity
      - combining generative and direct offers path forward

• Conclusion
Bias Towards Regularity Can Hurt an Indirect Encoding

HyperNEAT’s performance decreases as irregularity increases

Clune et al. PPSN. 2008
Would generative encodings perform better if they could produce more irregularity?

Test
  combine generative and direct
Clune et al. ECAL 2009

Hybrid

Generative

Switch

Direct

HyperNEAT

exploit regularity

generations

FT-NEAT

handle irregularity

fitness
HybrID > HyperNEAT & FT-NEAT on Target Weights

HybrID > HyperNEAT & FT-NEAT at generation 250 on 70%, 80%, and 90% (*p* < .01)

Clune et al. ECAL 2009
HybrID $\geq$ HyperNEAT on Bit Mirroring

![Graph showing comparison between HybrID and HyperNEAT on Bit Mirroring](image)

Clune et al. ECAL 2009
HybrID > HyperNEAT on Quadruped Controller

HybrID > HyperNEAT on all ($p < .001$)

Clune et al. CEC. 2009
Clune et al. IEEE TEC. 2011
Quadruped Controller Problem
0 faulty joints

HyperNEAT
Quadruped Controller Problem
0 faulty joints
Quadruped Controller Problem

HybrID
Quadruped Controller Problem
1 faulty joint

HyperNEAT
Quadruped Controller Problem
1 faulty joint

HyperNEAT

HybrID
HybrID Changes

After HyperNEAT Phase

After FT-NEAT Phase

Clune et al. IEEE TEC. 2011
Future Work:
Alternate Instantiations

- Results from Switch-Hybrid
- Automate switch point
- Offset-Hybrid

\[
\begin{array}{cccc}
2 & 2 & 1 & 1 \\
2 & 2 & 1 & 1 \\
2 & 2 & 1 & 1 \\
2 & 2 & 1 & 1 \\
\end{array}
\quad + \quad
\begin{array}{cccc}
-2 & 0 & 0 & 3 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 3 \\
\end{array}
\quad = \quad
\begin{array}{cccc}
0 & 2 & 1 & 4 \\
2 & 2 & 1 & 1 \\
2 & 2 & 1 & 1 \\
2 & 2 & 1 & 4 \\
\end{array}
\]
HybrID implications

• HybrID \geq \text{HyperNEAT} on all problems

• Suggests generative encodings have difficulty adjusting patterns in irregular ways

• HybrID offers path forward: a process of refinement
  - generative encoding + direct encoding
  - generative encoding + lifetime learning
Conclusions I

• more comprehensive picture of generative encodings
  • continuum of regularity
Conclusions I

• more comprehensive picture of generative encodings
  • continuum of regularity

• generative encodings exploit problem regularity
  • increasingly outperform direct encodings as problem regularity increases
Conclusions I

- more comprehensive picture of generative encodings
  - continuum of regularity
- generative encodings exploit problem regularity
  - increasingly outperform direct encodings as problem regularity increases
  - regular brains (visually complex regularities)
Conclusions I

• more comprehensive picture of generative encodings
  • continuum of regularity

• generative encodings exploit problem regularity
  • increasingly outperform direct encodings as problem regularity increases
  • regular brains (visually complex regularities)
  • regular behaviors
Conclusions II

- generative encodings struggle with irregularity
- HybrID offers path forward
- not stand-alone algorithms: combined with a process of refinement
HyperNEAT: Geometrically Aware


(a) Robot

(b) Concentric

(c) Parallel
Geometry in Checkers
Example:
HyperNEAT with Checkers
(Neural Computation journal 2010)

- Similar though not identical principles extrapolate across the board geometry
- Over 4,000 connections
  - Connectivity computed as function of geometry

Ken Stanley, MSU Invited Talk, 2013
The Sensitivity of HyperNEAT to Different Geometric Representations of a Problem

Proc. Genetic & Evolutionary Computation Conference 2009

Jeff Clune  Charles Ofria  Robert T. Pennock
Motivation

- HyperNEAT adds geometry...
  - do we have to manage it well?
  - how sensitive is HyperNEAT to different geometric representations of a problem?
Tic-Tac-Toe Example

1D

row: 1 2 3 4 5 6 7 8 9
column: 1 2 3 4 5 6 7 8 9

2D

y: 1 2 3
x: 1 2 3
row: y=1 1 2 3
row: y=2 1 2 3
row: y=3 1 2 3
column: x=1 1 2 3
column: x=2 1 2 3
column: x=3 1 2 3
Tic-Tac-Toe Example

1D

```
7  8  9
4  5  6
1  2  3
```

2D

```
1,1
1,2
1,3
2,1
2,2
2,3
3,1
3,2
3,3
```

Diagram representation of Tic-Tac-Toe in 1D and 2D.
Problem Domain

- Evolving gaits for simulated quadruped
- Chosen because
  - HyperNEAT previously did well with one geometric representation (Clune et al. CEC 2008, Clune et al. 2011)
  - Problem has many symmetries
Engineered Representation

ANN Controller

Clune et al. CEC 2008
• Each of the 50 trials had a different randomized representation for the entire run
- Engineered rep. beats Randomized rep. (p < .05)
  - Representations can make a difference
  - Human intuitions help

- But Randomized rep. beats direct encoding control (p < .001)
  - HyperNEAT can outperform direct encoding without an ‘intelligent’ rep.
  - Could be b/c of its generative nature, or via exploiting randomly generated regularity