

Genetic Programming



biologically
Inspired
computing

Sections I485/H400

- **Assignments: 35%**
 - Students will complete 4/5 assignments based on algorithms presented in class
- **Lab meets in I1 (West) 109 on Lab Wednesdays**
 - Lab 0 : January 14th (completed)
 - *Introduction to Python* (No Assignment)
 - Lab 1 : January 28th
 - *Measuring Information* (Assignment 1)
 - Graded
 - Lab 2 : February 11th
 - *L-Systems* (Assignment 2)
 - Graded
 - Lab 3: March 25th
 - Cellular Automata & Boolean Networks (Assignment 3)
 - Due: April 1st
 - Lab 4: April 8th
 - Genetic Algorithms (Assignment 4)
 - Due: April 22nd



■ Class Book

- Nunes de Castro, Leandro [2006]. *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. Chapman & Hall.
 - Chapters 1, 2, 3.1-3.4, 7.1-7.5, 8.1-8.2, 8.3.10

■ Lecture notes

- Chapter 1: “What is Life?”
- Chapter 2: “The Logical Mechanisms of Life”
- Chapter 3: “Formalizing and Modeling the World”
- Chapter 4: “Self-Organization and Emergent Complex Behavior”
 - posted online @ <http://informatics.indiana.edu/rocha/i-bic>

■ Other materials

- Flake’s [1998], *The Computational Beauty of Nature*. MIT Press
 - Chapters 20

The International Society for
ArtificialLife

ALIFE 15

■ Projects

- Due by May 4th in Oncourse

- ALIFE 15 (14)

- Actual conference due date: 2016

- <http://blogs.cornell.edu/alife14nyc/>

- 8 pages (LNCS proceedings format)

- <http://www.springer.com/computer/lncs?SGWD=0-164-6-793341-0>

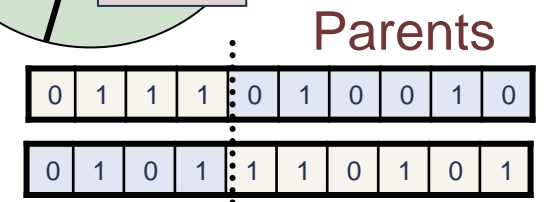
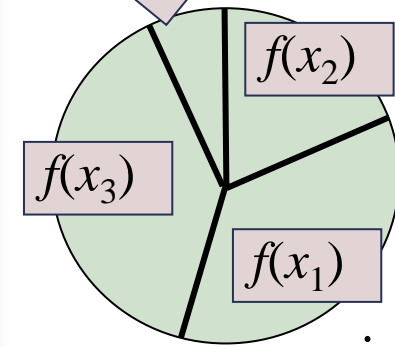
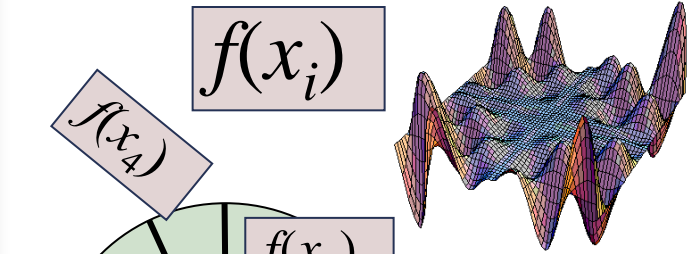
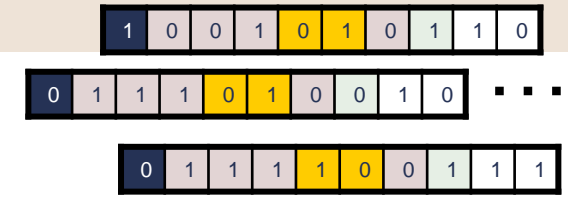
- Preliminary ideas **overdue!**

- Individual or group

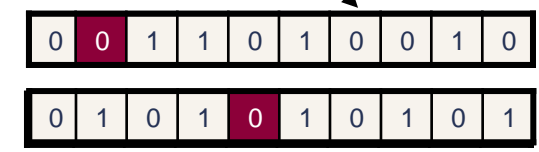
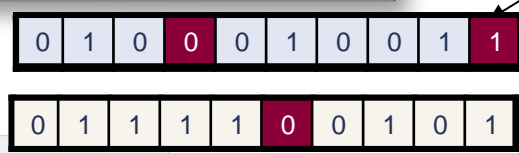
- With very definite tasks assigned per member of group

The workings

- 1) Generate Random population of bit-strings
- 2) Evaluate Fitness Function for each decoded solution
- 3) Reproduce next generation
 - Selection by fitness
 - Variation
 - crossover and mutation
 - Fill new population
- 4) Go back to 2) until stop criteria is met
 - Desired fitness
 - Specified number of generations
 - Convergence
 - Lack of variability in population and/or fitness
 - Tends to a peak

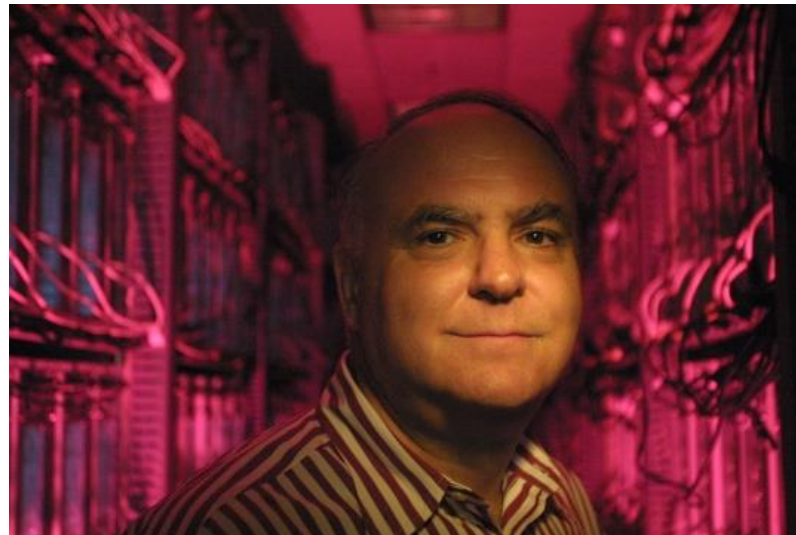


crossover no crossover



biologically
Inspired
computing

- Fogel, Owens and Walsh (1966)
 - *Artificial Intelligence through simulated evolution*. Wiley.
 - Evolution of finite-state machines
- John Koza (1992) at Stanford University
 - *Genetic Programming: On the programming of computers by means of Natural Selection*. MIT Press.



biologically
Inspired
computing

tree encodings

- Evolving computer programs to perform a task
 - No strict genotype-phenotype mapping
 - LISP programs
 - Can be expressed in the form of parse trees

```
(DEFUN AREA-OF-CIRCLE ())
```

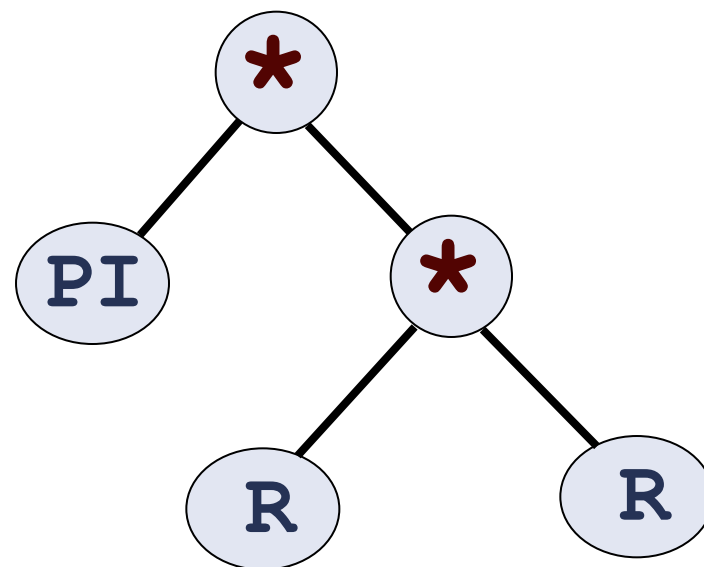
```
  (SETF R 45)
```

```
  (SETF PI 3.1415)
```

```
  (* PI (* R R)))
```

functions

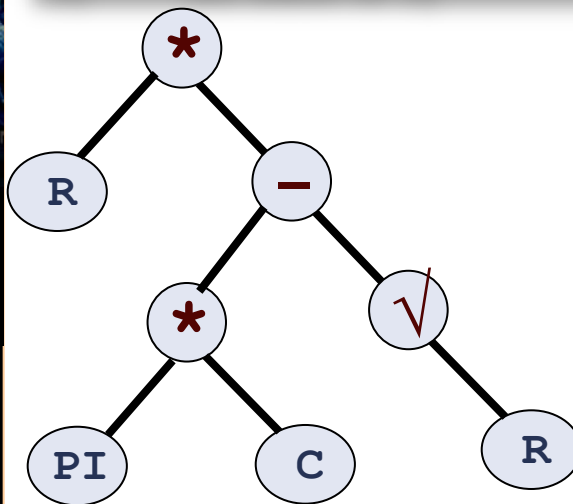
terminals



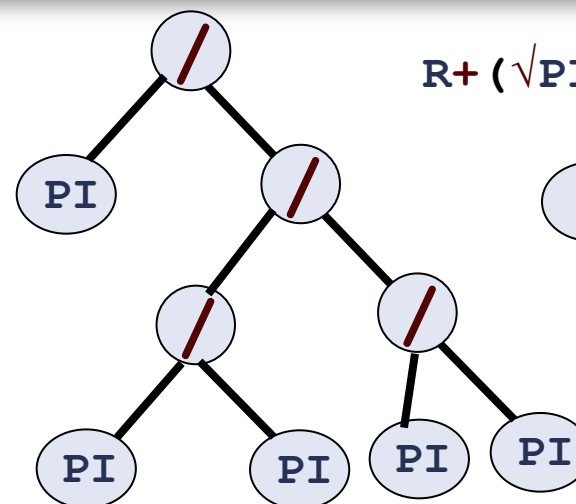
the workings

- 1) Choose a pool of possible functions and terminals
 - Setting up a language of description
- 2) Generate Random population of trees (programs)
 - Must be syntactically correct (parsing)
 - Size is usually restricted
- 3) Evaluate Fitness Function for each tree
 - Desired I/O
 - Simplicity, speed
- 4) Reproduce next generation with variation
 - Trees with higher fitness value reproduce with higher probability
- 5) Go back to 3)

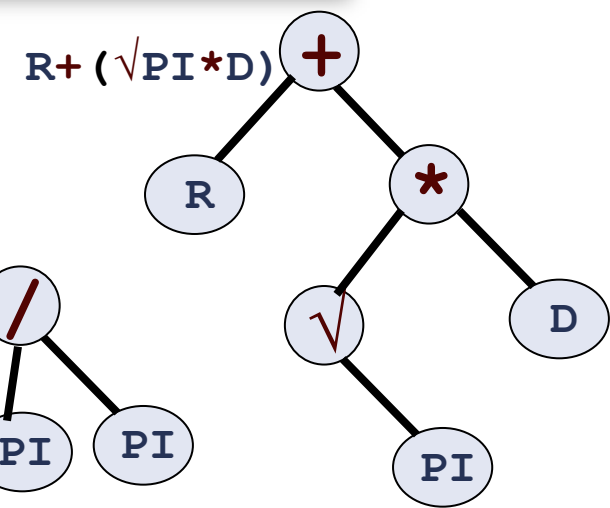
- / + == √ ABS ...
PI C R B ...



$$R * [(PI * C) - \sqrt{R}]$$

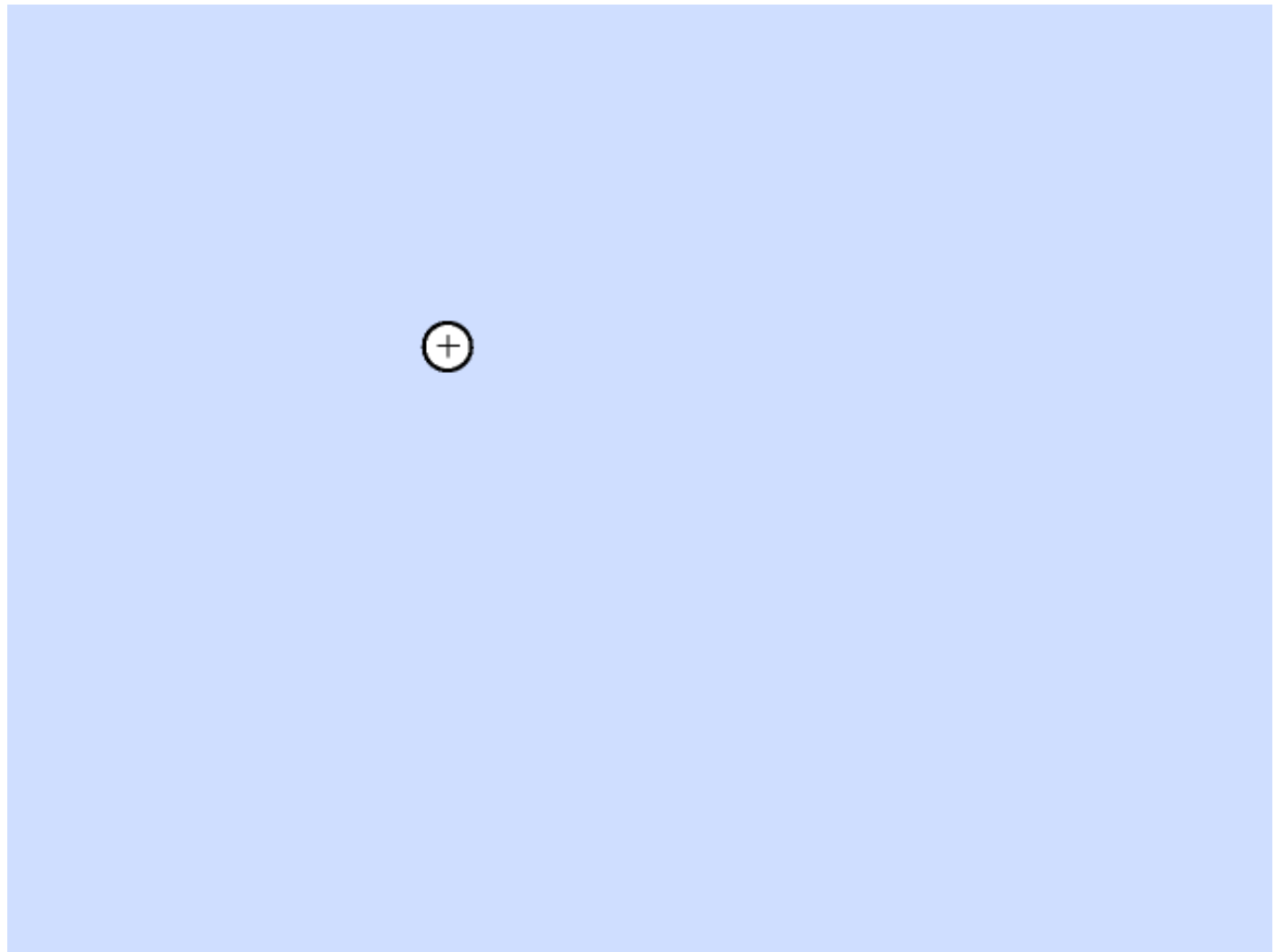


$$PI / \{ [(PI / PI) / (PI / PI)] \}$$



$$R + (\sqrt{PI * D})$$

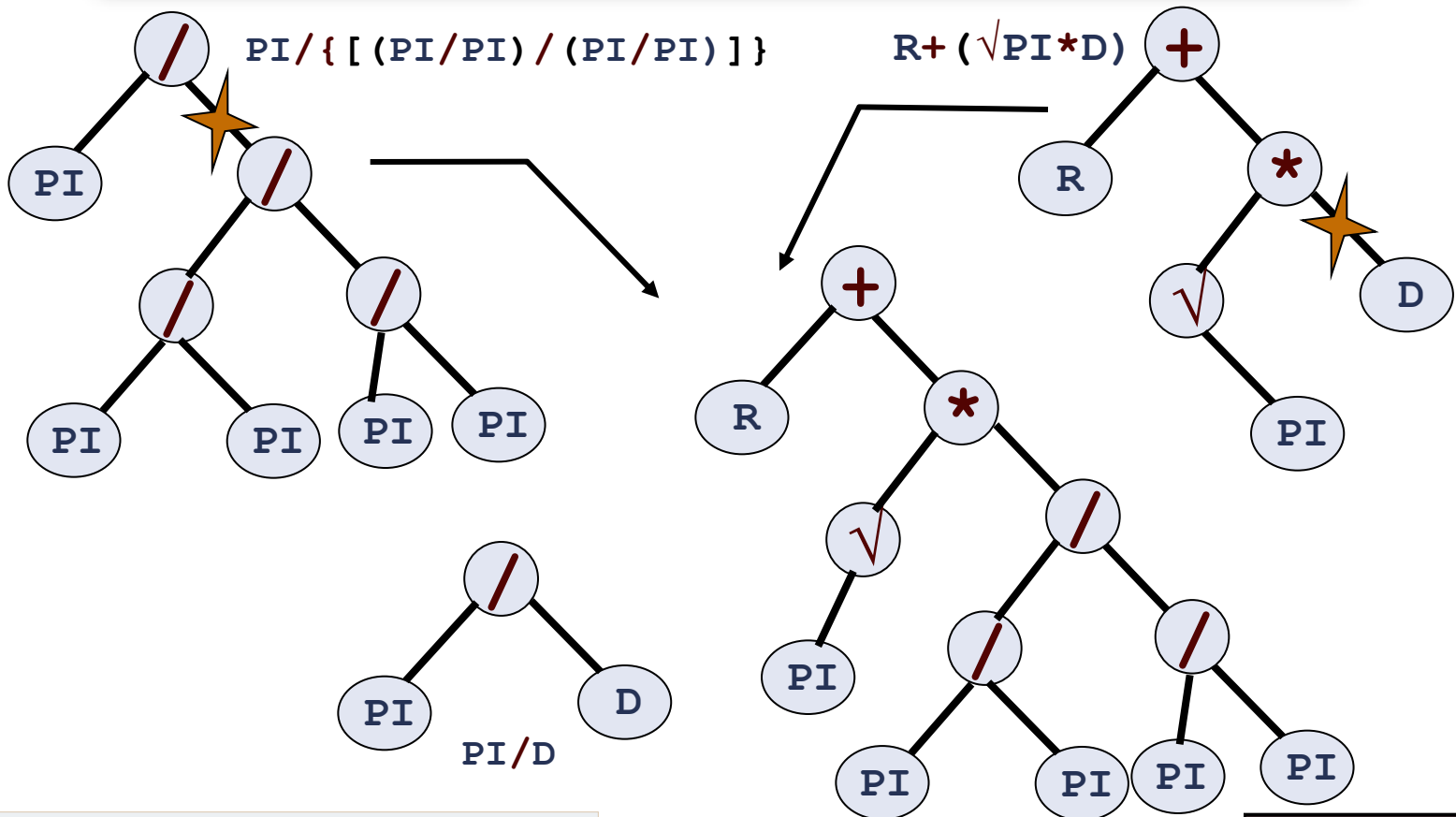
biologically
Inspired
computing

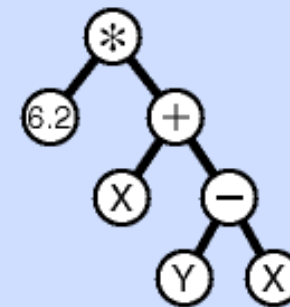
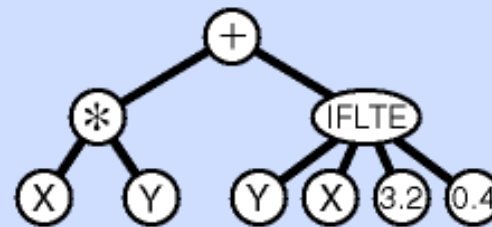


biologically
Inspired
computing

crossover

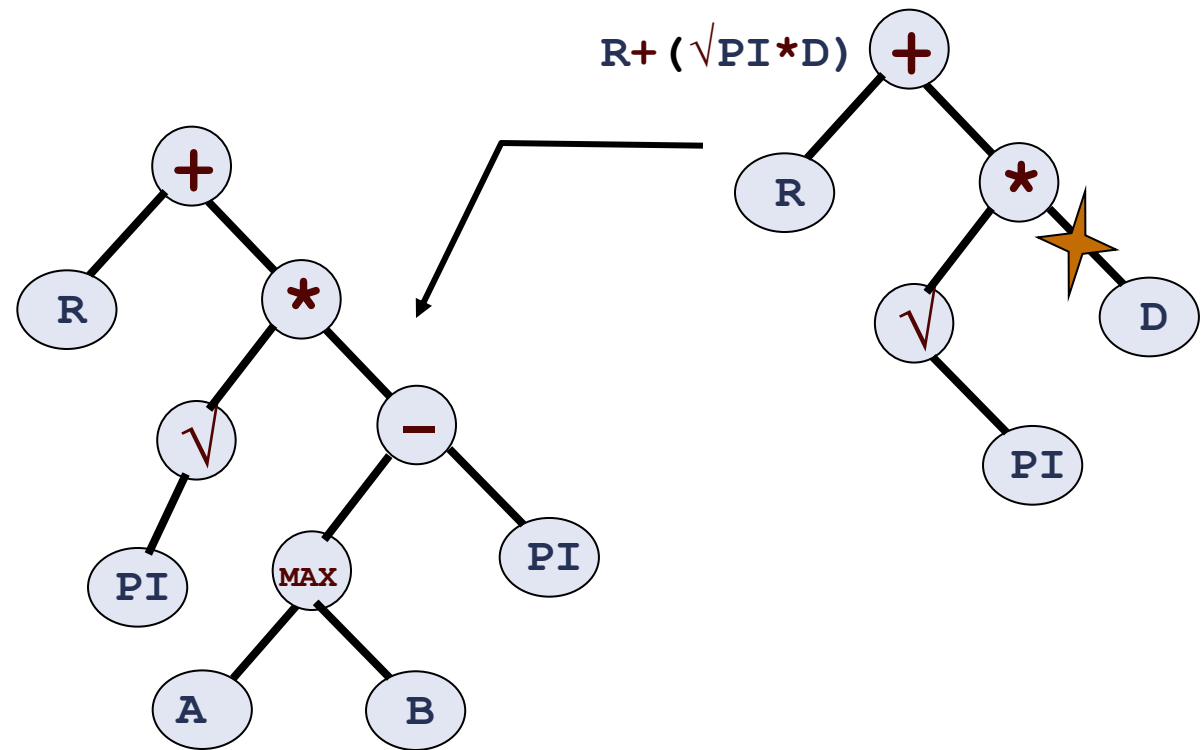
- Choose random point in each parent's tree
- Exchange subtrees beneath to produce offspring
 - Allows size of program to increase or decrease



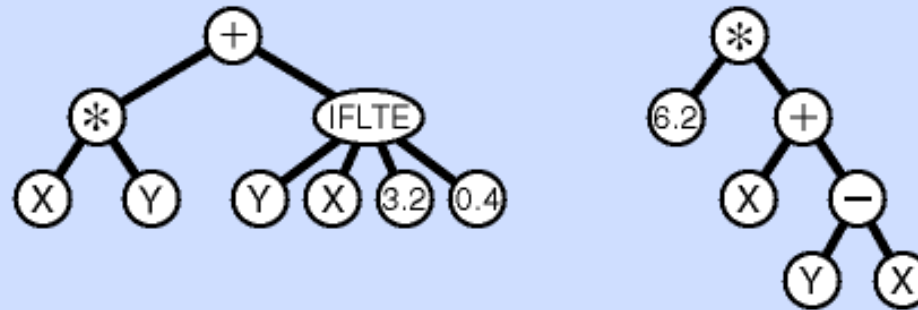


mutation

- Choose random point in a tree
- Replace subtree beneath with random tree



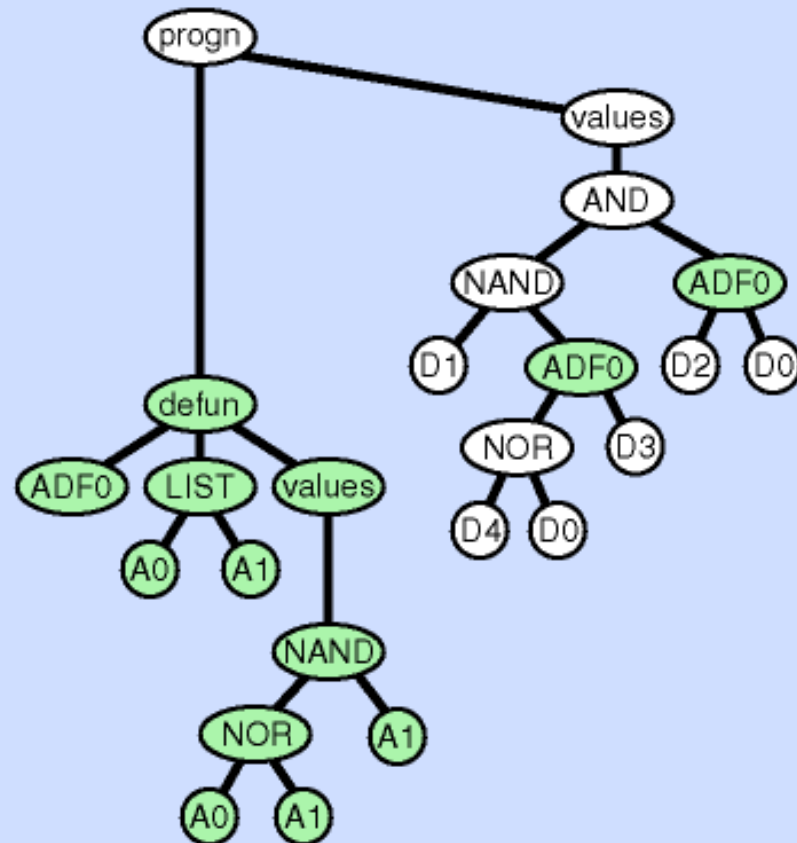
$$R + \{ \sqrt{PI} * [MAX(A, B) - PI] \}$$



```
return ["no more bottles",  
        "of beer on the wall", "beer"]  
except KeyError: return "no more beer"  
  
for i in range(99, 0, -1):  
    b1, b2 = bottle(i), bottle(i-1)  
    print "%d is on the wall, %d is, %d"  
    " bottles down, pass it around."  
    print "%d bottles on the wall." % locals()
```

biologically
Inspired
computing

Architecture-altering operations



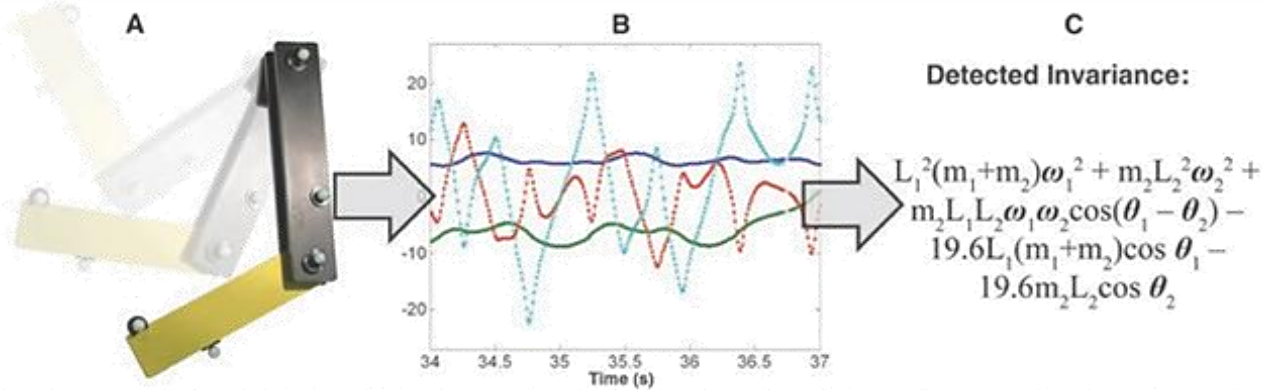
biologically
Inspired
computing

applications

- Optimal control
- Planning
- Symbolic regression
 - Fit real data
 - Example: Uncover laws of physics
 - Schmidt M., Lipson H. (2009) "Distilling Free-Form Natural Laws from Experimental Data," *Science*, **324** (5923): 81 - 85.
 - Binary Classification
 - Software Tool
 - Eureka: <http://ccsl.mae.cornell.edu/eureka>
- Robot strategies
 - Robocup
- Evolvable hardware



biologically
Inspired
computing



A symbolic regression tool

Eureqa: <http://creativemachines.cornell.edu/Eureqa>

The screenshot shows the Eureqa software interface with the following components:

- List of current solutions:**

Size	Error	Solution
17	0.181	$f(x) = -1.06 \cos(-1.62 + 2x) + x \cos(-1.62 +$
13	0.183	$f(x) = -1.00 \sin(2x) + x \sin(2x)$
6	0.442	$f(x) = x \sin(2x)$
4	0.494	$f(x) = -0.47 + \cos(x)$
2	0.589	$f(x) = \cos(x)$
1	0.560	$f(x) = -0.26$
- Selected solution plotted with the data:** A scatter plot showing Train Data (blue dots), Validation Data (green dots), and the Validated Solution (magenta line) against index (0 to 100) on the x-axis and y (-4 to 2) on the y-axis.
- Quick statistics of solution:**

-1.06 cos(-1.62 + 2 x) + x cos(-1.62 + 2 x)

Name	Train Data	Validation Data
Sample Size	80	52
R-squared	0.96	0.94
Correlation Coeff	0.98	0.97
Mean Squared Error	0.08	0.09
Mean Absolute Error	0.24	0.26
Minimum Error	0.00082	0.00
Maximum Error	0.66	0.63
- Accuracy/complexity front of best solutions:** A step plot showing Error [MAE] on the y-axis (0.2 to 0.8) versus Complexity [size] on the x-axis (2 to 18). It includes Train Solutions Points (blue), Validated Solutions Points (green), and the Selected Solution (red).

rocha@indiana.edu
<http://informatics.indiana.edu/rocha/i-bic>



readings

■ Class Book

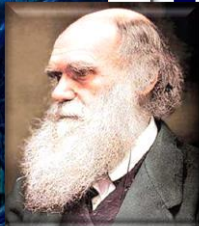
- Nunes de Castro, Leandro [2006]. *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. Chapman & Hall.
 - Chapter 2, 7, 8
 - Appendix B.3.2-3 - Turing Machines, Computational complexity
 - **Chapter 3, all sections**
 - **Sections 7.8, 8.3.2, 8.3.10**

■ Lecture notes

- Chapter 1: “What is Life?”
- Chapter 2: “The logical Mechanisms of Life”
- Chapter 3: Formalizing and Modeling the World
- Chapter 4: “Self-Organization and Emergent Complex Behavior”
- Chapter 5: “Reality is Stranger than Fiction”
 - posted online @ <http://informatics.indiana.edu/rocha/i-bic>

Optional materials

- Flake’s [1998], *The Computational Beauty of Life*. MIT Press
 - Chapter 20
- *Scientific American*: Special Issue on ***the evolution of Evolution***, January 2009.



biologically
Inspired
computing