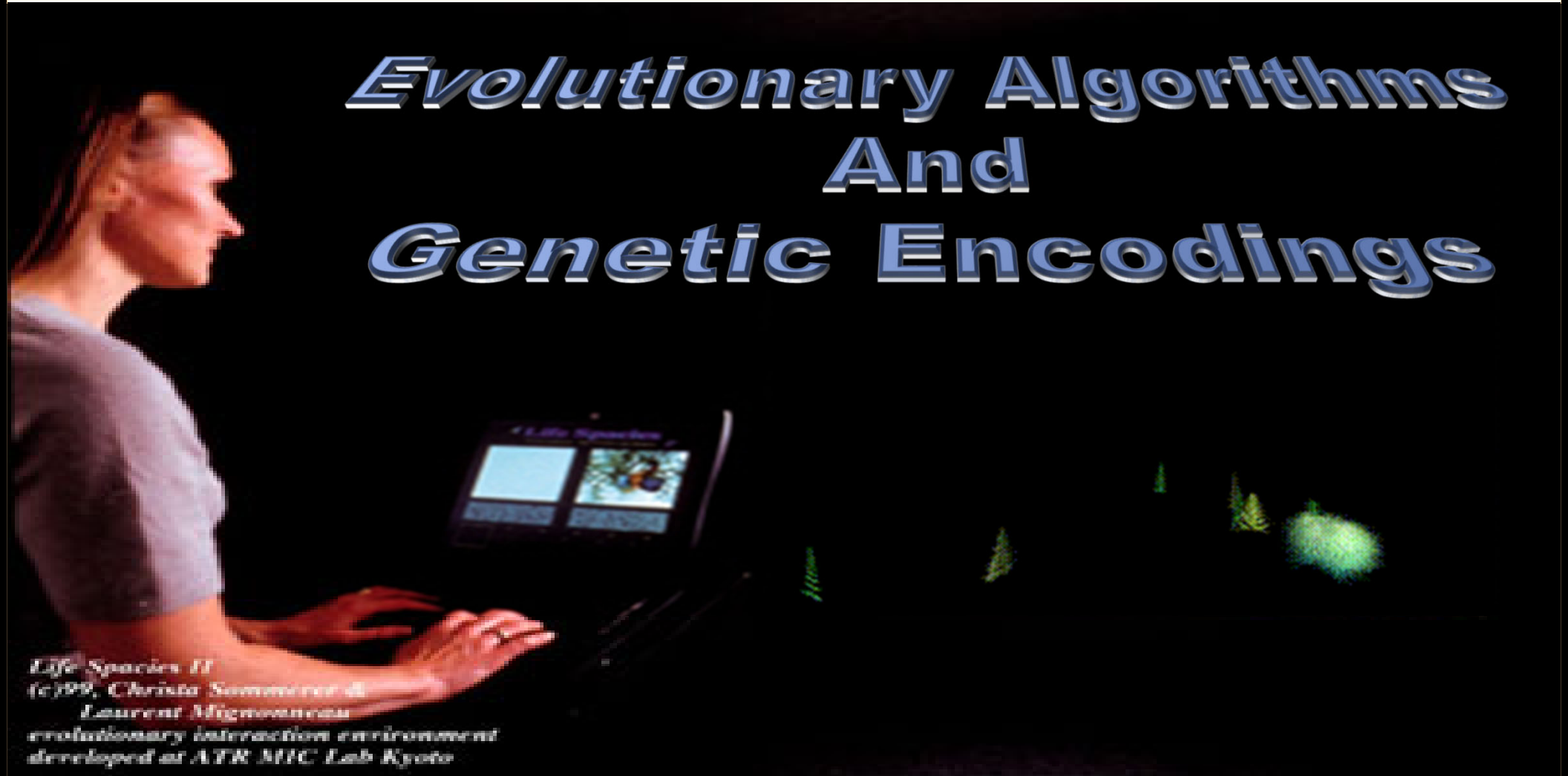


Evolutionary Algorithms And Genetic Encodings



Life Species II
(c)199, Christa Sommeret &
Laurent Mignonneau
evolutionary interaction environment
developed at ATR MIC Lab Kyoto

key events coming up

- Labs: 35% (ISE-483)
 - Complete 5 (best 4 graded) assignments based on algorithms presented in class
 - Lab 4 : April 8th
 - Evolutionary Algorithms, (Assignment 4)
 - Delivered by SSIE583 Group 2
 - Due April 22nd
 - Lab 5: April 29th
 - Ant Clustering Algorithm, (Assignment 5)
 - Delivered by Group 1
 - Due May 6th
- SSIE – 583 -Presentation and Discussion: 25%
 - Present and lead the discussion of an article related to the class materials
 - Enginet students post/send video or join by Zoom
 - April 25th or April 29th
 - Conrad, M. [1990]. "The geometry of evolution." *Biosystems* 24: 61-81.
 - Mario Franco
 - Stanley, Kenneth O., Jeff Clune, Joel Lehman, and Risto Miikkulainen. "Designing Neural Networks through Neuroevolution." *Nature Machine Intelligence* 1, no. 1 (January 2019): 24–35.
 - Jessica Lasebikan
 - Lindgren, K. [1991]. "Evolutionary Phenomena in Simple Dynamics." In: *Artificial Life II*. Langton et al (Eds). Addison-wesley, pp. 295-312.
 - Akshay Gangadhar
 - Salahshour, Mohammad. "Interaction between Games Give Rise to the Evolution of Moral Norms of Cooperation." *PLOS Computational Biology* 18, no. 9 (September 29, 2022): e1010429
 - Srikanth Iyer
 - Discussion by all



bit.ly/atBIC

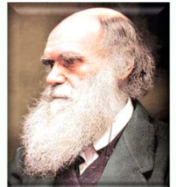
until now

■ Class Book

- Floreano, D. and C. Mattiussi [2008]. *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*. MIT Press. Preface, **Chapters 1 and 4**.

■ 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
- Chapter 6: Von Neumann and Natural Selection
 - posted online @ <http://informatics.indiana.edu/rocha/i-bic>



■ Papers and other materials

- Optional
 - Nunes de Castro, Leandro [2006]. *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. Chapman & Hall.
 - Chapter 2, 7, 8
 - **Chapter 3, sections 3.1 to 3.5**



bit.ly/atBIC

■ Projects

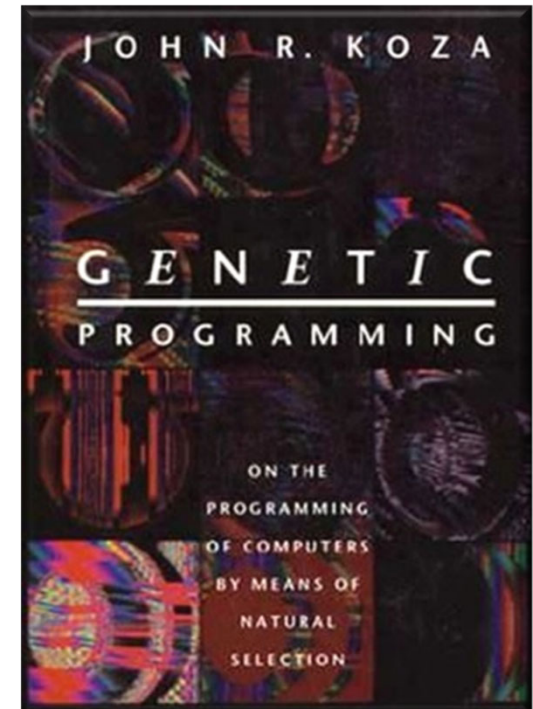
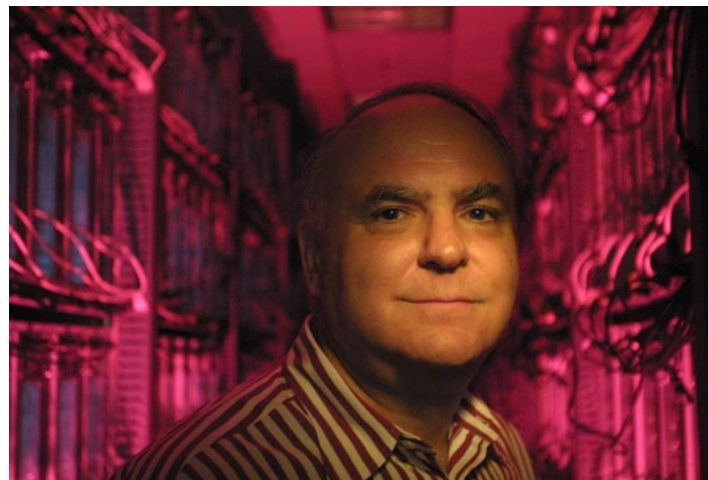
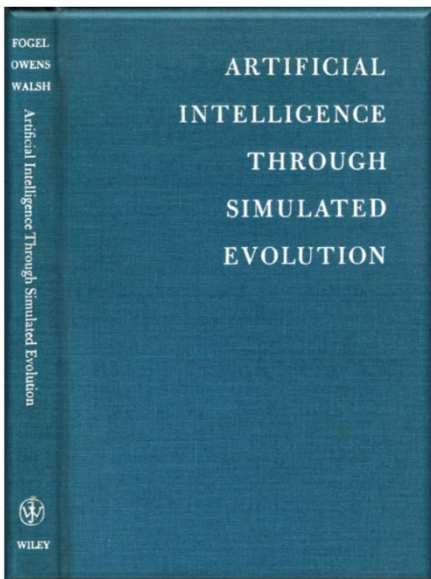
- Due by **May 8th** in Brightspace, “Final Project Paper” assignment
 - ALIFE 2023
 - Not to submit to actual conference due date (April 3rd , 2024)
 - <https://2024.alife.org/>
 - 8 pages, author guidelines:
 - https://2024.alife.org/call_paper.html
 - MS Word and Latex/Overleaf templates
 - Preliminary ideas **by March 15**
 - Submit to “Project Idea” assignment in Brightspace.
- Individual or group
 - With very definite tasks assigned per member of group

ALIFE 2024

Tackle a real problem using bio-inspired algorithms, such as those used in the labs.



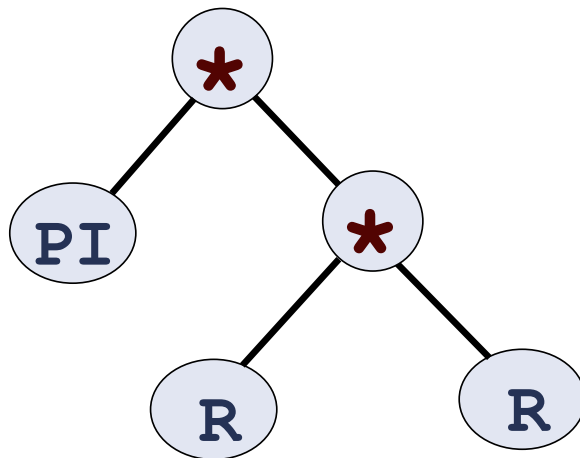
- Fogel, Owens and Walsh (1966)
 - *Artificial Intelligence through simulated evolution*. Wiley.
 - Evolution of finite-state machines
- John Koza (1992) at Stanford University
 - *Genetic Programming: On the programming of computers by means of Natural Selection*. MIT Press.



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))
```

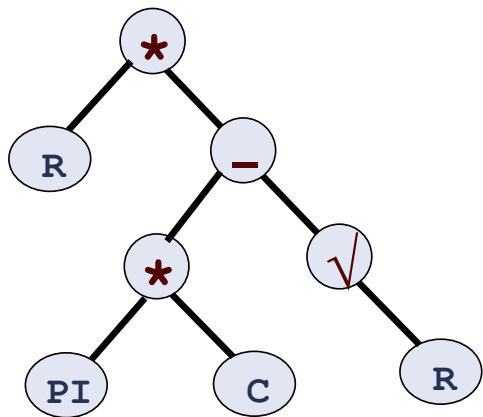
functions

terminals

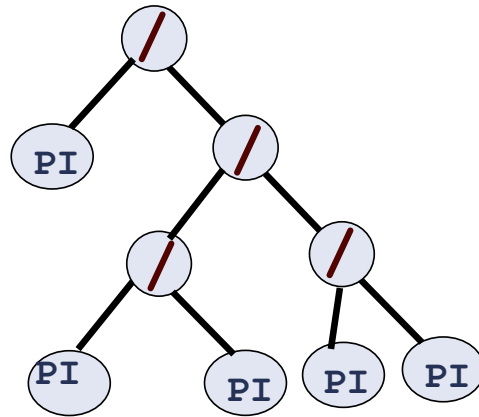
the workings

- 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)

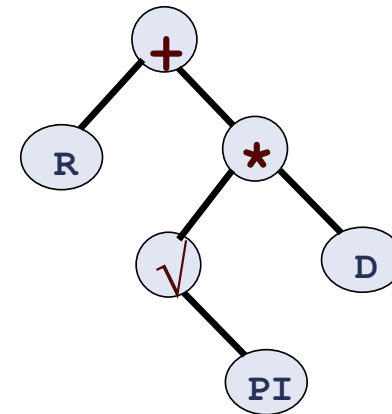
- / + == √ ABS ...
 PI C R B ...



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



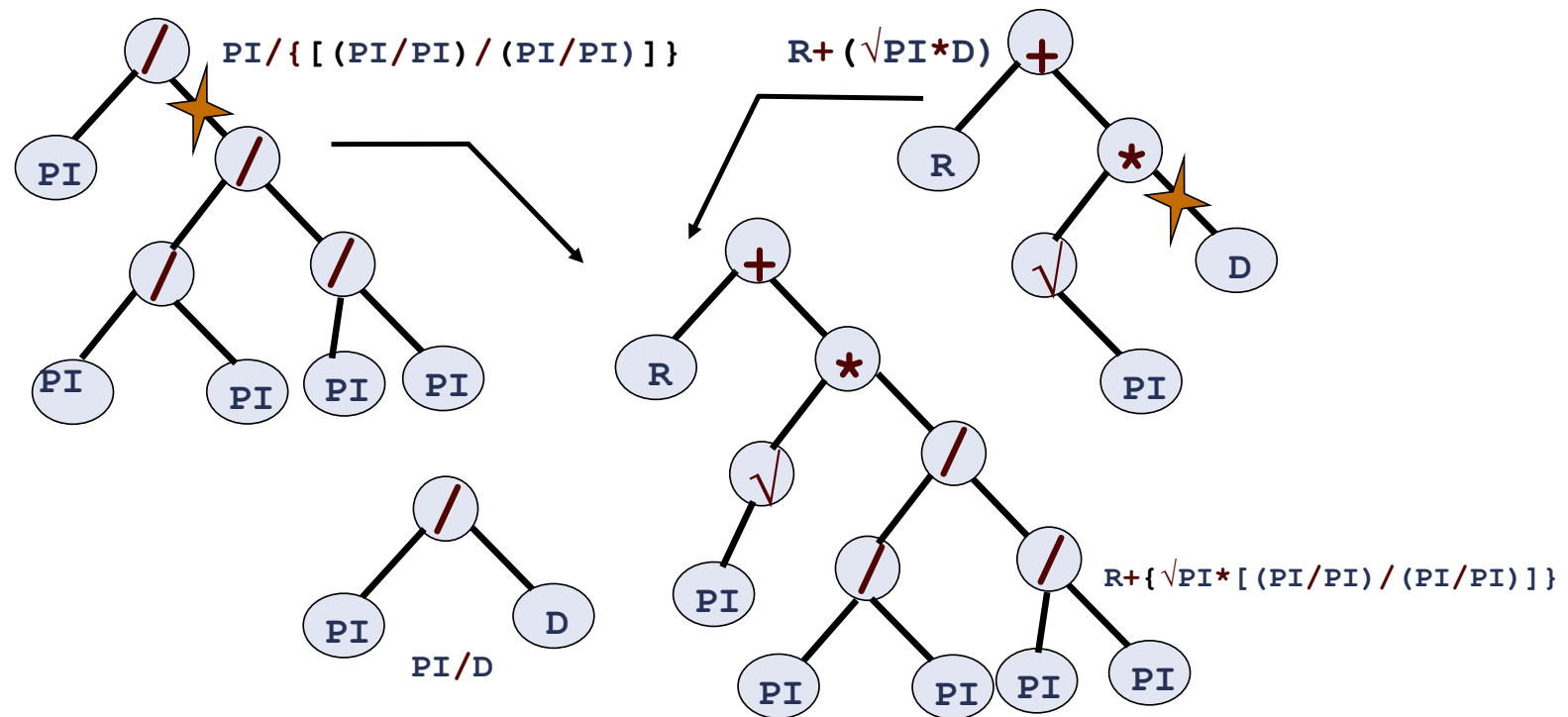
$$PI / \{ [(PI / PI) / (PI / PI)] \}$$



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

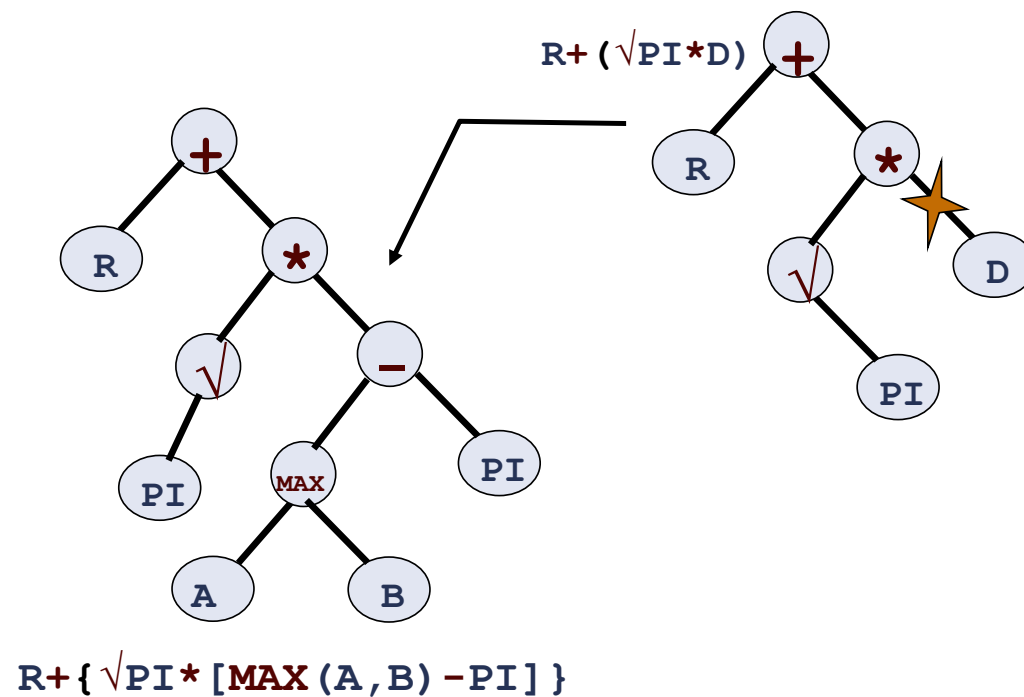
crossover

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

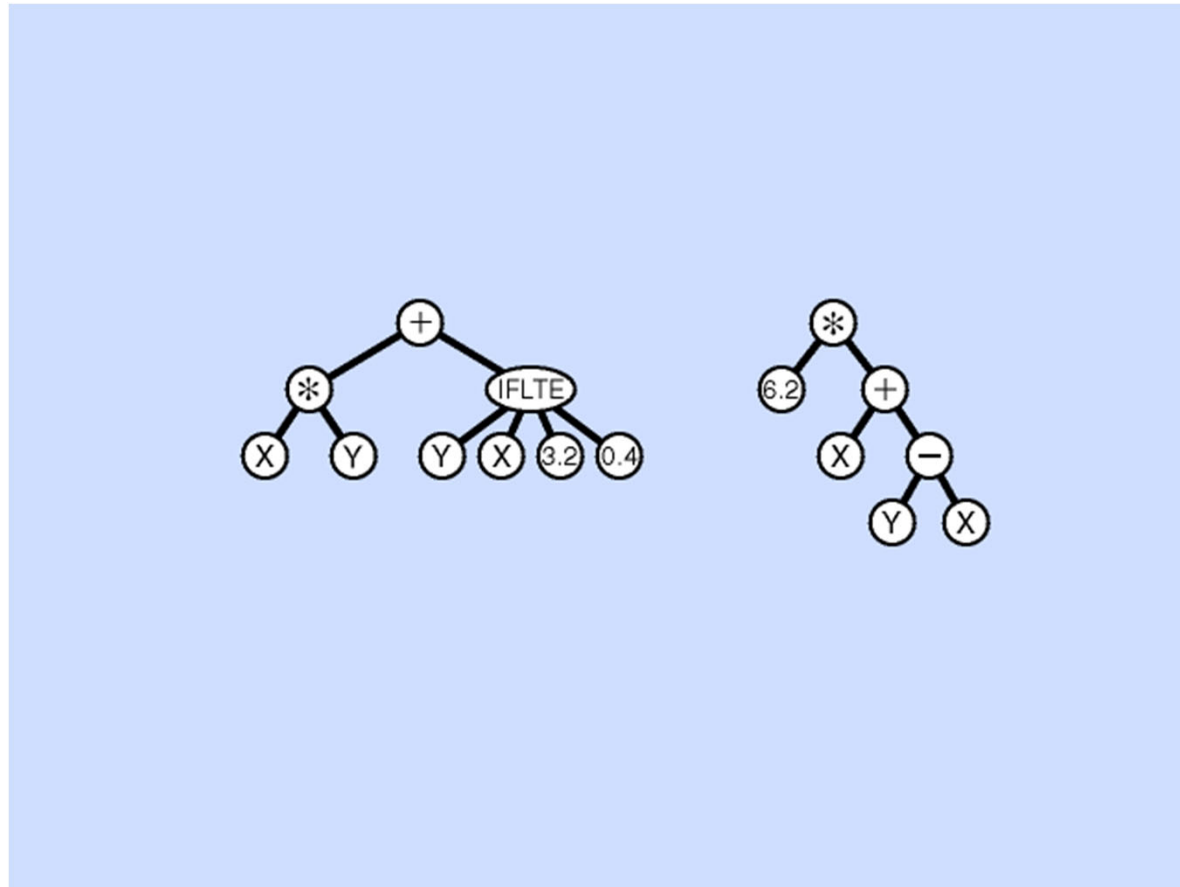


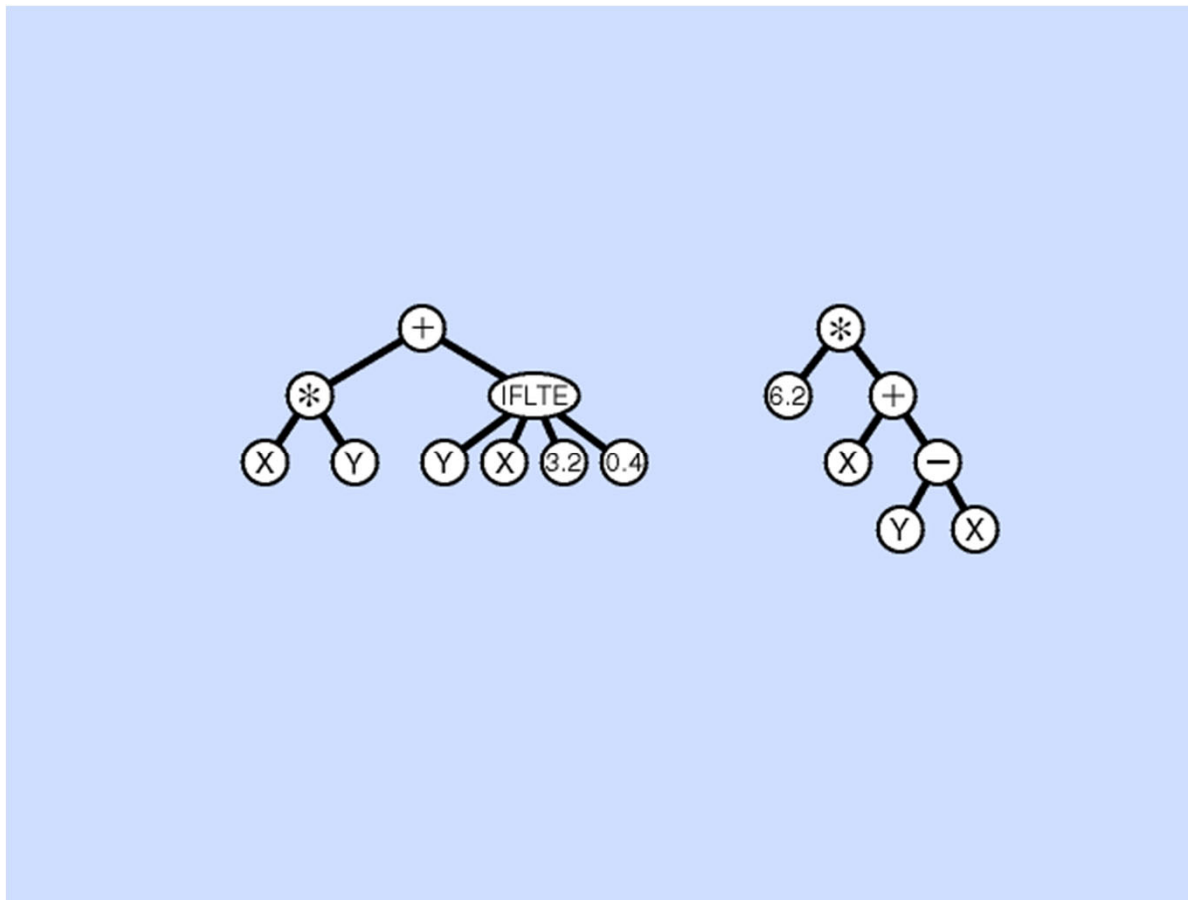
mutation

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

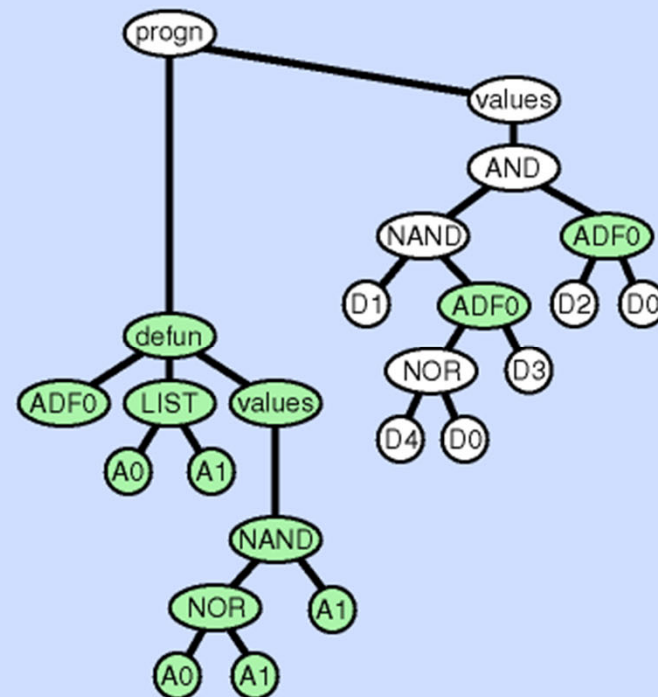




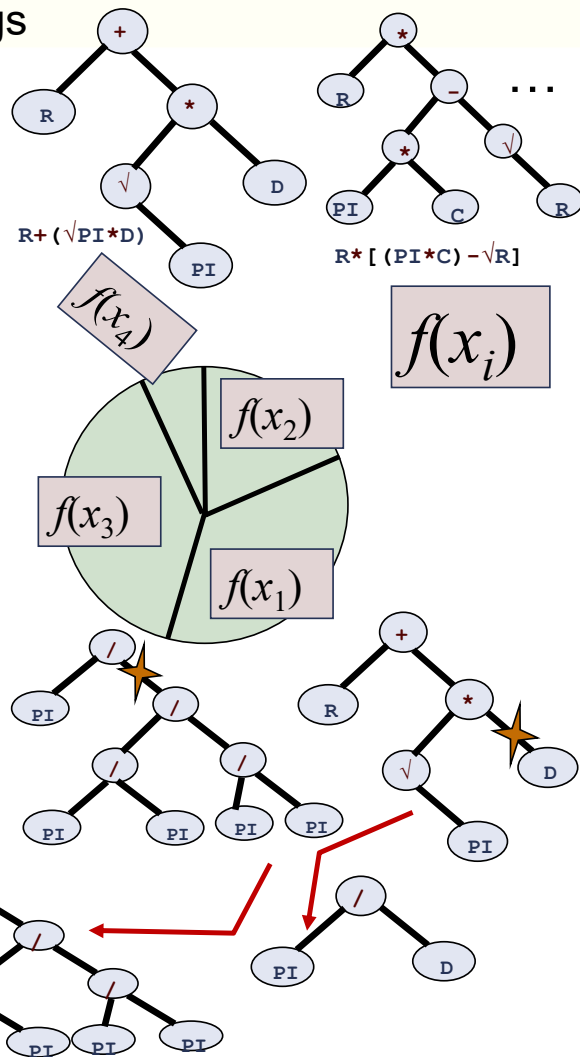




Architecture-altering operations



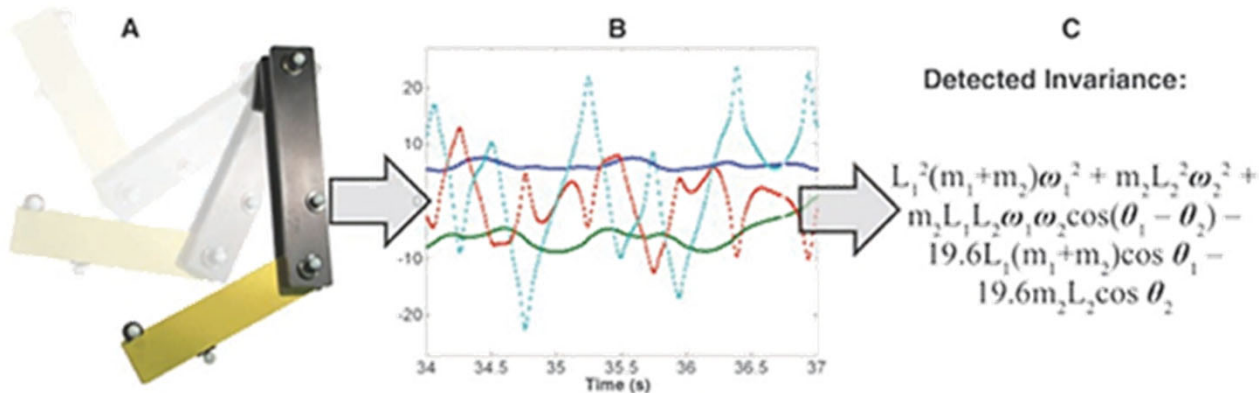
The workings



- 1) Generate Random population of trees/programs
- 2) Evaluate Fitness Function for each program
 - Desired I/O, simplicity, speed
- 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


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: <https://www.creativemachineslab.com/eureka.html>
- Robot strategies
 - Robocup
- Evolvable hardware



applications

- Optimal control
- Planning
- Symbolic regression
 - Fit real data
 - Example: Uncover laws of physics
 - Schmidt M., Lipson H. (2009)
 - Binary Classification
 - Software Tool
 - Eureka: <https://www.creativecomputing.com/eureka/>

SoftSea.com 

Eureka



- ⊞ Rating: ★★★★★
- ⊞ Version: 0.99.8 Beta
- ⊞ Publisher: creativemachines.cornell.edu
- ⊞ File Size: 9.12 MB
- ⊞ Date: Apr 03, 2014
- ⊞ Price: \$2499.00
- ⊞ License: Free Trial Software
- ⊞ Category: Calculator Office

DataRobot
DOCS

[UI docs](#)
[API docs](#)
[Platform](#)
[Learn more](#)
[Releases](#)

[Modeling](#) > [Model insights](#) > [Describe](#) > Eureka Models

Eureka Models

The **Eureka Models** tab provides access to model blueprints for Eureka generalized additive models (Eureka GAM), Eureka regression, and Eureka classification models. These blueprints use a proprietary Eureka machine learning algorithm to construct models that balance predictive accuracy against complexity.

Fundamentals of modeling

DataRobot workflow overview

- + Build models
- Model insights
- + Evaluate
- + Understand
- Describe
- Blueprint
- Coefficients (preprocessing)
- Monotonic constraints
- Data Quality Handling Report
- Eureka Models
- Model log
- Model Info
- Rating Tables
- GA2M output (from Rating Tables)
- + Predict

Eureka Model

Complexity: 5 Error: 55.163 (Surrogate Mean Squared Error) Export

Grad_Rate = High Cardinality and Text features Modeling +13.3117095487359 + 4.37348652143869*STANDARDIZED_Top2Spere + 5.82993537024461*min(STANDARDIZED_Outstate, 1) - 11.9767048941044*(STANDARDIZED_Apps, 0.848189085633554, 1.43843144277699)

Models by Error vs. Complexity

These are the most accurate (lowest error) models with the least complexity (a measure of the size and mathematical complexity of the analytical model)

Selected Model Detail

Complexity: 5 Error: 55.163

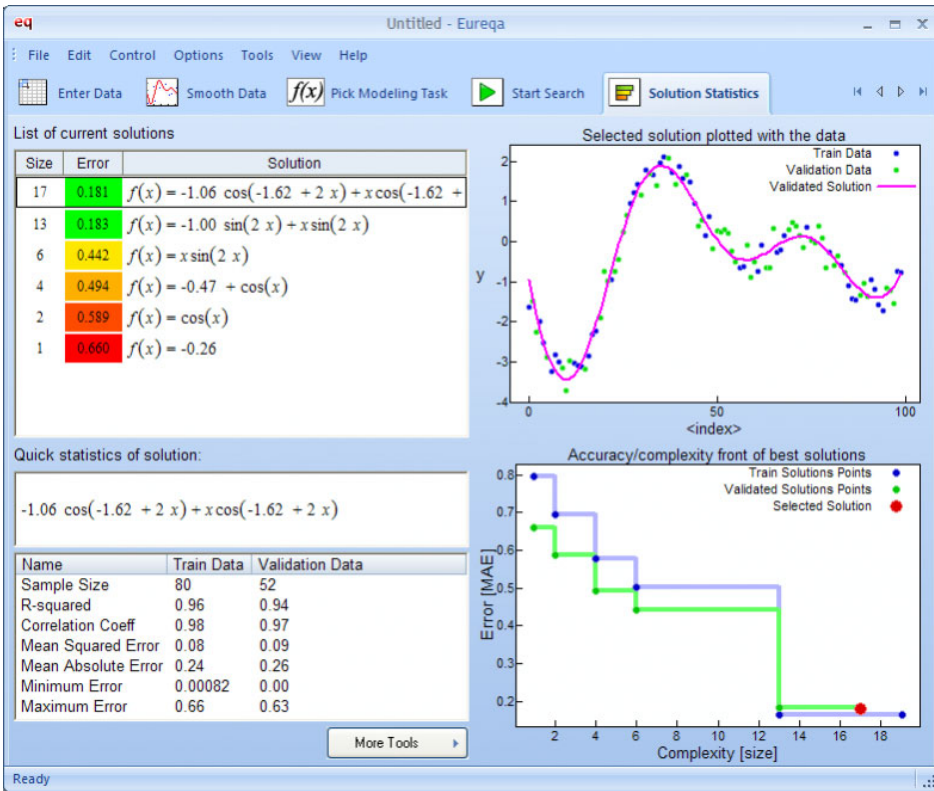
Grad_Rate = High Cardinality and Text features Modeling +13.3117095487359 + 4.37348652143869*STANDARDIZED_Top2Spere + 5.82993537024461*min(STANDARDIZED_Outstate, 1) - 11.9767048941044*(STANDARDIZED_Apps, 0.848189085633554, 1.43843144277699)

BINGHAMTON
UNIVERSITY
STATE UNIVERSITY OF NEW YORK

rocha@binghamton.edu
casci.binghamton.edu/academics/i-bic

A symbolic regression tool

Eureqa: <https://www.creativemachineslab.com/eureqa.html>

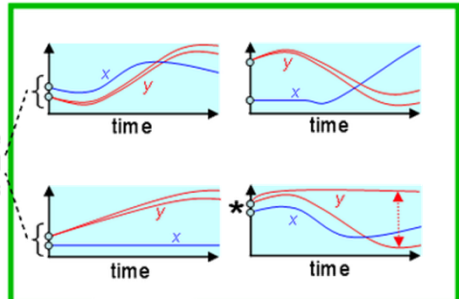


Candidate models

$$\begin{cases} \frac{dx}{dt} = -2y^2 + \log x \\ \frac{dy}{dt} = -x + \frac{y}{6} \end{cases} \quad ? \quad \begin{cases} \frac{dx}{dt} = -\sqrt{y} + \frac{x}{5} \\ \frac{dy}{dt} = -\sin y \end{cases}$$

$$\begin{cases} \frac{dx}{dt} = -3 \frac{y+1}{y-1} \\ \frac{dy}{dt} = -\frac{y}{x^2+1} \end{cases} \quad \begin{cases} \frac{dx}{dt} = -y^{18} + \log x \\ \frac{dy}{dt} = -x + \frac{y}{4x} \end{cases}$$

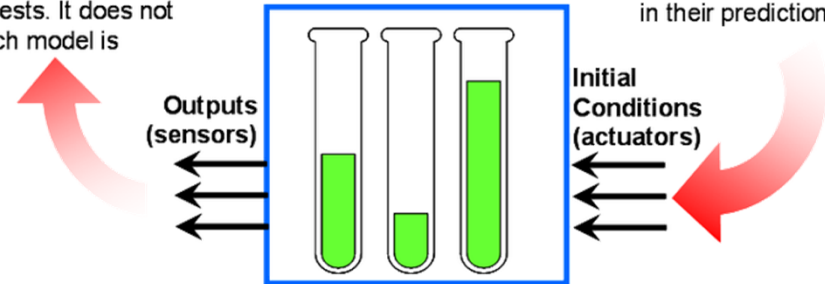
Candidate tests



b The inference process generates several *different* candidate symbolic models that match sensor data collected while performing previous tests. It does not know which model is correct.

Inference Process

c The inference process generates several possible new candidate tests that disambiguate competing models (make them disagree in their predictions).



a The inference process physically performs an experiment by setting initial conditions, perturbing the hidden system and recording time series of its behavior. Initially, this experiment is random; subsequently, it is the best test generated in step c.

rocha@binghamton.edu
casci.binghamton.edu/academics/i-bic

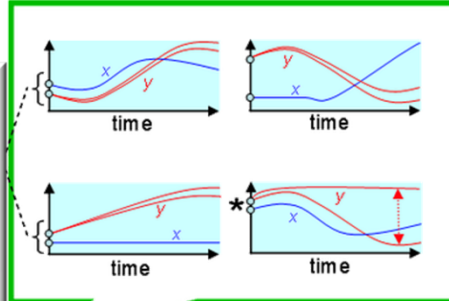
A symbolic regression tool

Eureqa: <https://www.creativemachineslab.com/eureqa.html>

Candidate models

$$\left[\frac{dx}{dt} = -2v^2 + \log x \quad \left[\frac{dx}{dt} = -\sqrt{y} + \frac{x}{y} \right. \right.$$

Candidate tests



c The inference process generates several possible new candidate tests that disambiguate competing models (make them disagree in their predictions).

Initial Conditions (actuators)



a The inference process physically performs an experiment by setting initial conditions, perturbing the hidden system and recording time series of its behavior. Initially, this experiment is random; subsequently, it is the best test generated in step c.

Size	Error	Solut
17	0.181	$f(x) = -1.06 \cos(-1.62$
13	0.183	$f(x) = -1.00 \sin(2 x) +$
6	0.442	$f(x) = x \sin(2 x)$
4	0.494	$f(x) = -0.47 + \cos(x)$
2	0.589	$f(x) = \cos(x)$
1	0.660	$f(x) = -0.26$

Name	Train Data	Valida
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



Genetic Programming in Python,
with a scikit-learn inspired API:

gp learn

<https://gplearn.readthedocs.io/>

*One general law, leading to the advancement of all organic beings, namely,
multiply, vary, let the strongest live and the weakest die.*

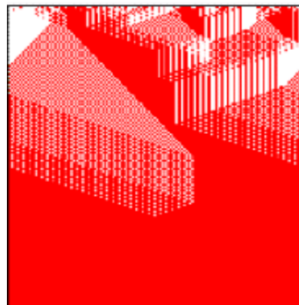
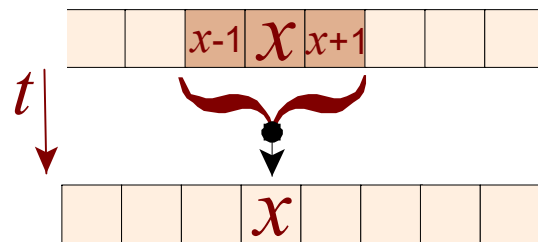
— Charles Darwin, *On the Origin of Species* (1859)

rocha@binghamton.edu
casci.binghamton.edu/academics/i-bic

- **Binary encodings**
 - Typically fixed-length
- **Many-letter encoding**
 - Larger alphabet (e.g. graph-generation grammars)
- **Real-valued encodings**
 - Genes take real values
- **Tree Encodings**
 - Genetic programming
- **Indirect Encodings**
 - Modeling Phenotype development or **post-transcription** processes
 - L-Systems, Dynamical systems, evolutionary robotics

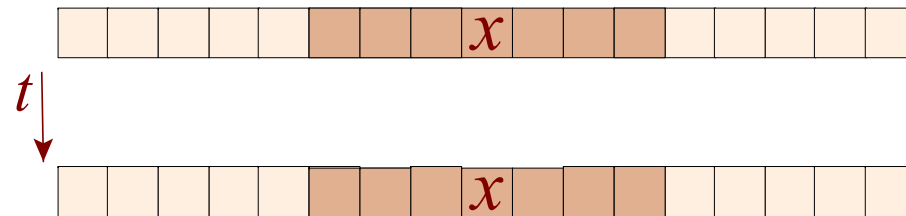
homogenous lattice of state-determined systems

Cellular Automata



Density Task (a.k.a majority classification problem)

- Lattices of 149 Binary Cells (599, 999)
- Rules of Radius 3 (7 Cells in Neighborhood)
- Task: Organize to
 - All 1's if Initial Configuration (IC) has more 1 Cells
 - All 0's if IC has more 0 Cells



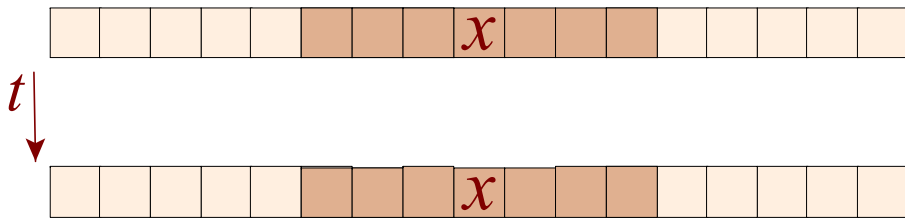
Possible neighborhood states

$$K^{|N|} = 2^7 = 128$$

Possible CA transition functions

$$K^{K^{|N|}} = 2^{128} \approx 3.4 \times 10^{38}$$

encoding in GA with binary encoding

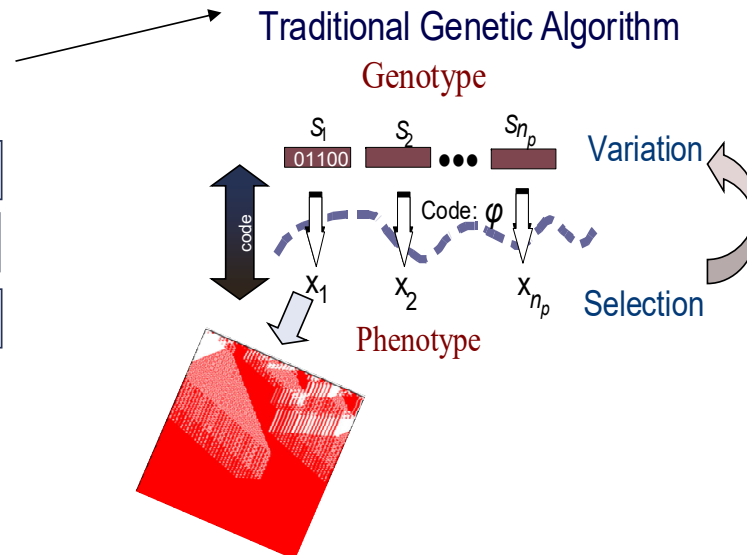


Possible neighborhood states $K^{|N|} = 2^7 = 128$

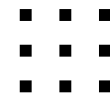
Pop of rules

- 010010101100100
- 010010101100100
- 010010101100100
- 010010101100100
- 010010101100100

Used in the evolutionary search by GA (elite selection)



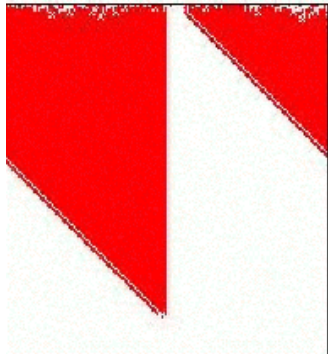
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	1	1	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	1	0
0	0	0	0	0	1	1	0	0
0	0	0	0	0	1	1	1	0
0	0	0	0	1	0	0	0	1
0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	1	0	1
0	0	0	0	1	0	1	1	1
0	0	0	0	1	1	0	0	0
0	0	0	0	1	1	0	1	0



With genetic algorithms

- Das, Mitchell and Crutchfield
 - Used Genetic Algorithm to evolve rules for this task

Typical Result:
Block Expansion



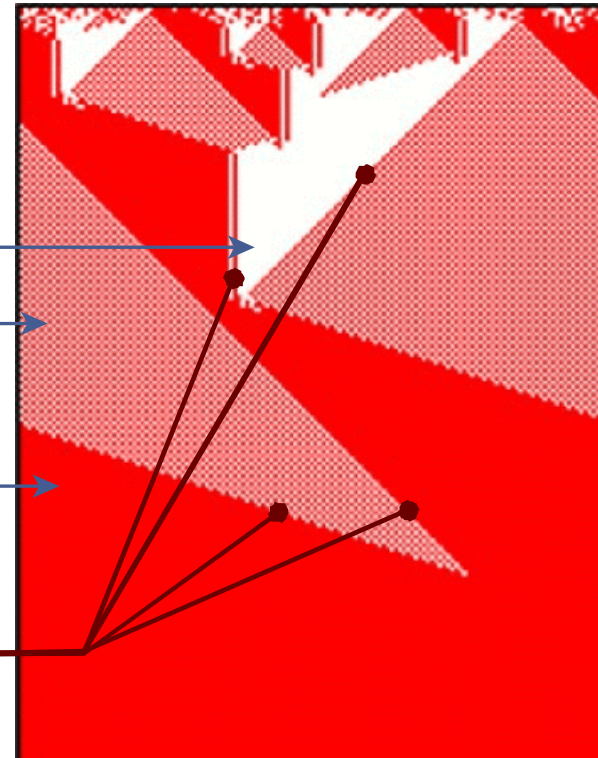
Regular domains

{0+}

{10+}

{1+}

Particles



Das,R., Mitchell,M., Crutchfield,J.P., [1994]. "A genetic algorithm discovers particle-based computation in cellular automata". In: *Parallel Problem Solving from Nature - PPSN III*. Davidor,Y., Schwefel,H.-P., Manner,R. (Eds.), Springer-Verlag, pp. 344-353.

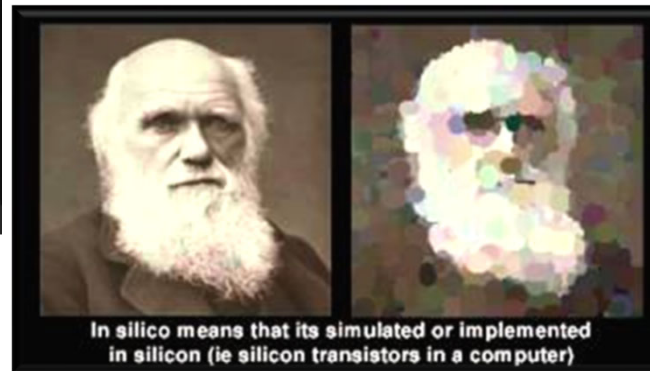
to evolve photos with numerical encodings



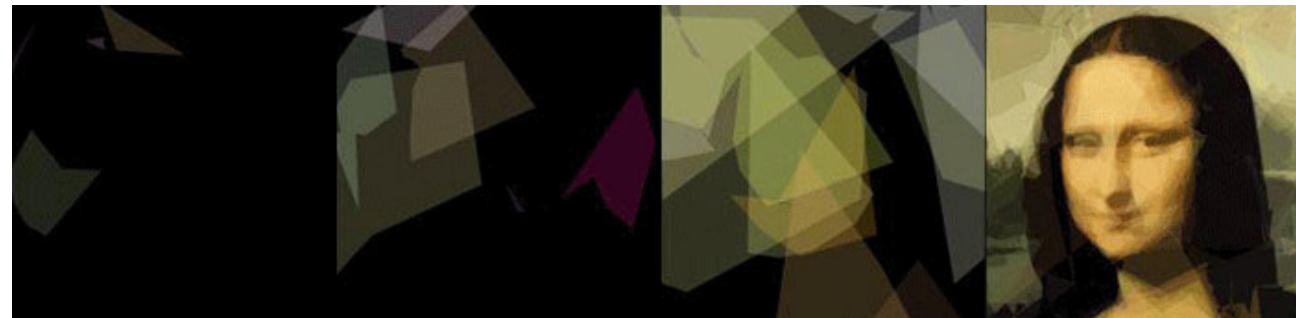
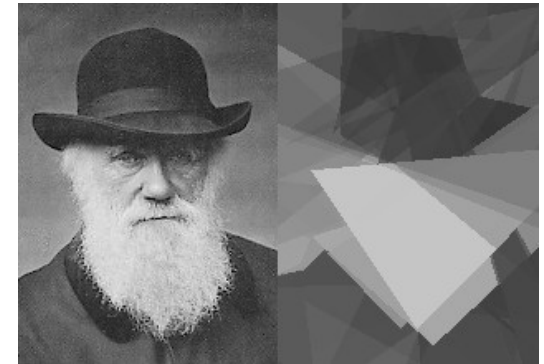
Original

128 circles

256 circles



In silico means that its simulated or implemented in silicon (ie silicon transistors in a computer)



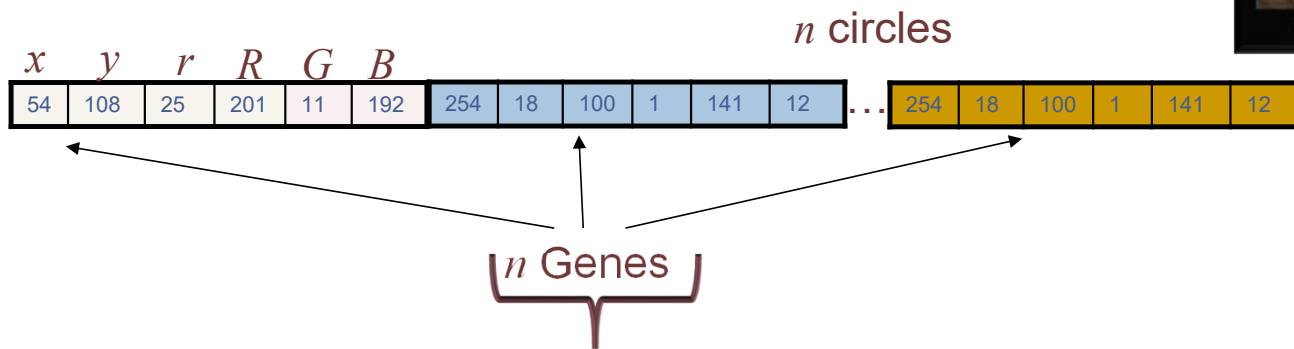
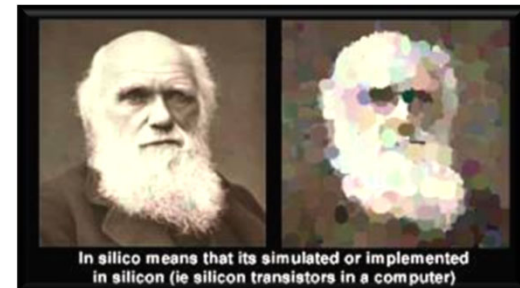
In genetic algorithms

- 1) **Genotypes contain real or integer values**
 - 1) Crossover is performed in the same way
 - 2) Mutation assigns a random number in a given interval
- 2) **More computationally demanding for Reals**
- 3) **Attention to crossover points**
 - 1) Conversion to binary avoids crossover issues, but longer genotypes

0.3	1.7	3.8	1.7	6	1.2	3.2	6.4	2.8	0
-----	-----	-----	-----	---	-----	-----	-----	-----	---



0.3	1.7	3.8	1.7	2.9	1.2	3.2	6.4	2.8	0
-----	-----	-----	-----	-----	-----	-----	-----	-----	---



Agent Chromosome/Genotype (Population of p agents)

Evolutionary algorithms to optimize neural networks

- **Capabilities (not in gradient-based ANN)**
 - Generation of ANN building blocks
 - Activation functions
 - Hyperparameter optimization
 - Architecture and learning algorithm search and optimization
- **Massive Paralellism**
 - Population of solutions

nature machine intelligence

[Explore content](#) ▾
 [About the journal](#) ▾
 [Publish with us](#) ▾
 [Subscribe](#)

[nature](#) >
 [nature machine intelligence](#) >
 [review articles](#) >
 [article](#)

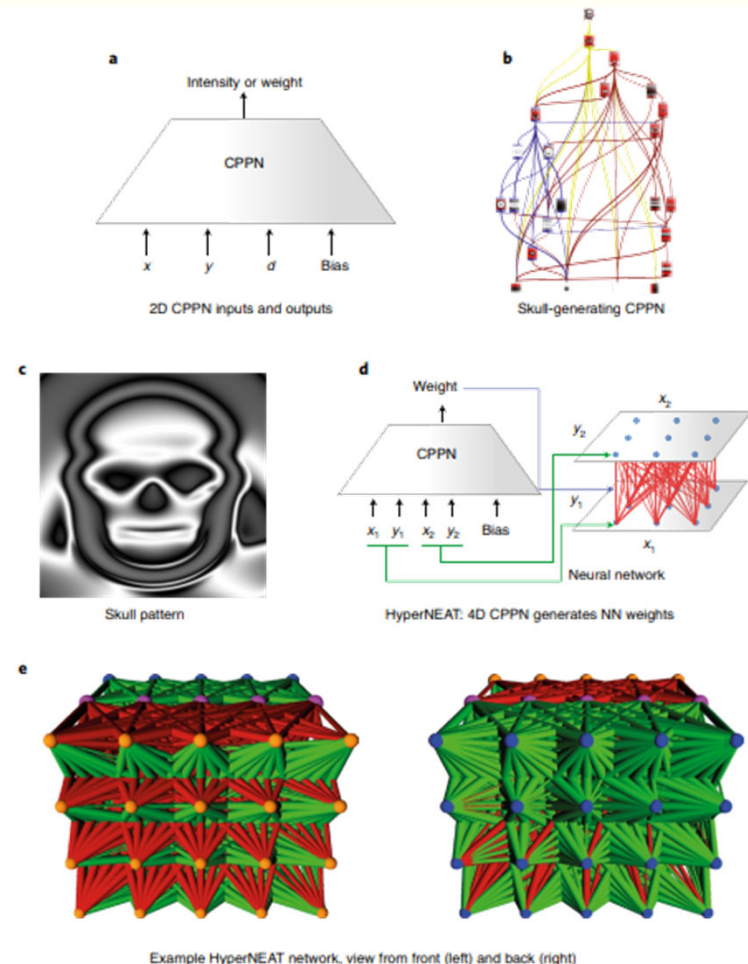
Review Article | [Published: 07 January 2019](#)

Designing neural networks through neuroevolution

[Kenneth O. Stanley](#) ✉,
 [Jeff Clune](#) ✉,
 [Joel Lehman](#) ✉ &
 [Risto Miikkulainen](#) ✉

[Nature Machine Intelligence](#) 1, 24–35 (2019) | [Cite this article](#)

Stanley, K.O., Clune, J., Lehman, J. et al (2019). Designing neural networks through neuroevolution. *Nat Mach Intell*, 24–35.

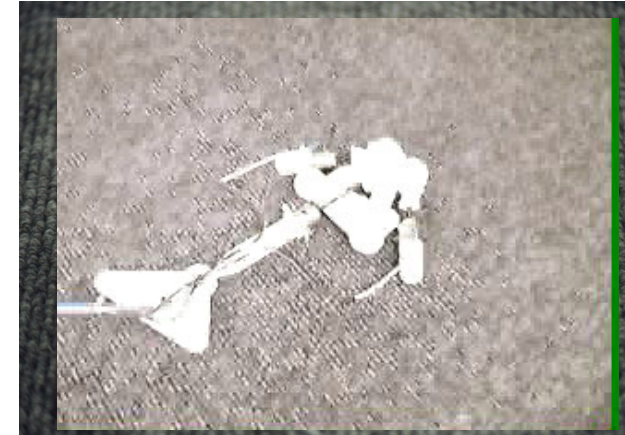
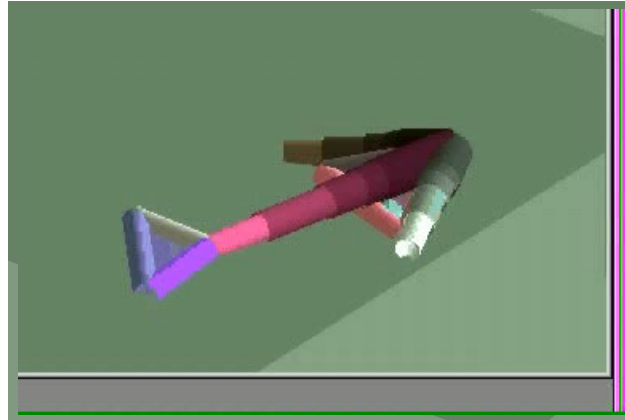
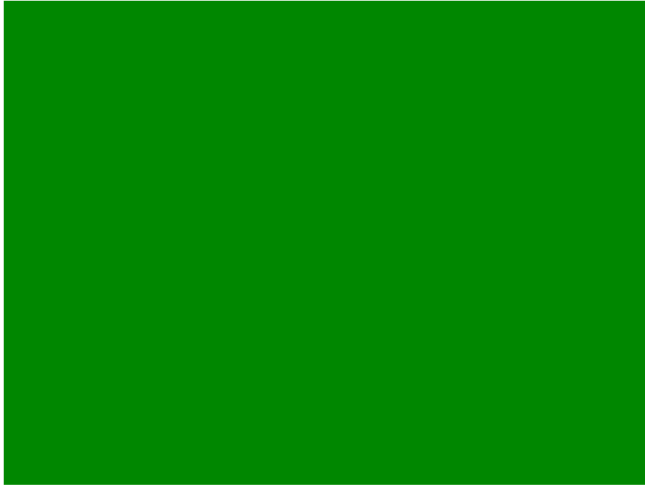


Example HyperNEAT network, view from front (left) and back (right)

evolving morphologies and robots with indirect encodings

Karl Sims' simulations, Genobots, and the Golem project

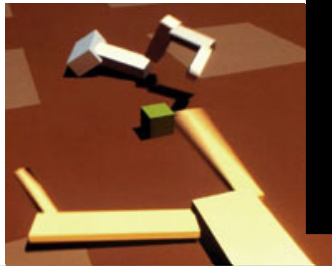
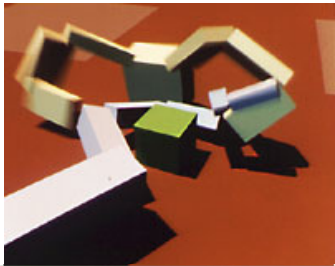
Sims, Karl. "Evolving virtual creatures."
In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pp. 699-706. 2023.



<http://demo.cs.brandeis.edu/golem/>

Generative Representations for
Evolutionary Design Automation
3D Genobots

The Golem
Project



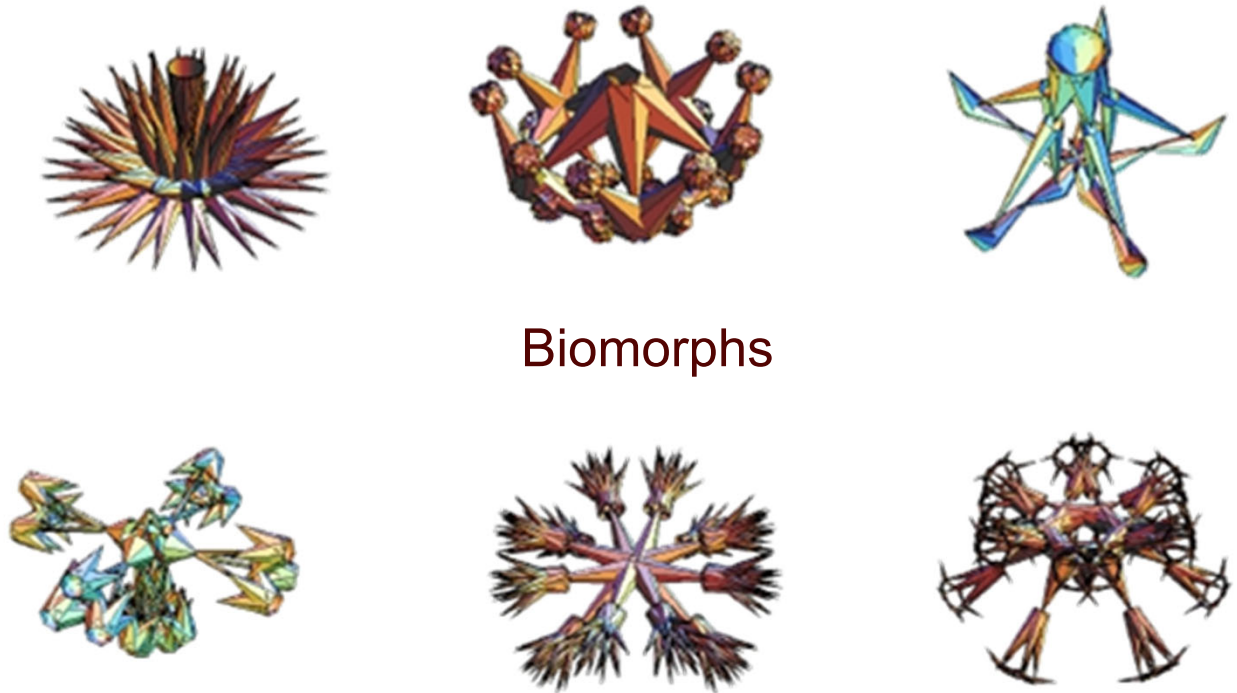
<https://www.karlsims.com/evolved-virtual-creatures.html>

rocha@binghamton.edu
casci.binghamton.edu/academics/i-bic

http://www.demo.cs.brandeis.edu/pr/evo_design/evo_design.html

objective function may be subjective

"Once a Darwinian process gets going in a world, it has an open-ended power to generate surprising consequences: us, for example" Richard Dawkins



Biomorphs

readings

■ Class Book

- Floreano, D. and C. Mattiussi [2008]. *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*. MIT Press.
 - Chapter 7

■ 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
- Chapter 6: Von Neumann and Natural Selection
 - posted online @ casci.binghamton.edu/academics/i-bic

■ Papers and other materials

● Optional

- Nunes de Castro, Leandro [2006]. *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. Chapman & Hall.
 - Chapter 5, 7.7, 8.3.1, 8.3.6,

