Biologically inspired AI, part 2

Tournament selection, crossover, and mutation

- For each individual in population:
  - Choose 8 other individuals at random
  - Rank them in order of fitness
  - Choose one of the 9 probabilistically as a function of rank
  - Crossover the winner with another individual randomly chosen from the tournament. Mutate the resulting child.
  - Put the resulting child in the new generation.

Single-point crossover

Choose a single locus at random

Parents:
0010001 × 1000100101111000101001101110000...
1111100 0100101010000000111000100101O1...

Offspring:
0010000010010101000000011100010010101...
1111100 100010010111100010100010111000...

Choose one of the offspring to go into the next population

Uniform crossover

At each locus, decide at random whether to take bit from parent 1 or parent 2.

Parents:
00100011000100101111000101001101110000...
111110001001010100000000111000100101O1...

Offspring:

011100110001010110000001110011011 0001...

Put offspring into the next population.

GA code for evolving cellular automata

- Demo
- params
- ga.c
- Homework 7

Coevolutionary Learning

Candidate solutions and training examples coevolve. Inspired by host-parasite coevolution in biology.

Ideas on how to apply this to CA example?

Coevolutionary Learning

Candidate solutions and training examples coevolve.
Inspired by host-parasite coevolution in biology.

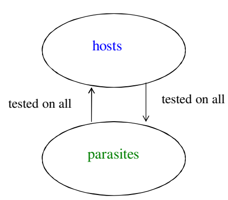
– **Fitness of CA (host):** how well it performs on training examples.

– **Fitness of initial configuration (parasite):** how well it defeats candidate solutions.

	Cellular Automata
Traditional GA (with crossover)	20%
Traditional GA (mutation only)	0%

Percentage of successful runs

Coevolution



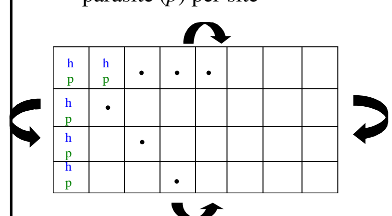
	Cellular Automata
Traditional GA (with crossover)	20%
Traditional GA (mutation only)	0%
Coevolution	0%

Percentage of successful runs

Spatial Coevolution

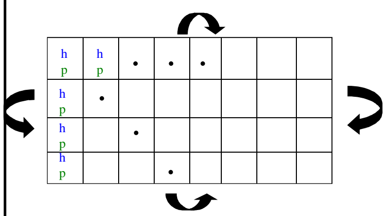
Spatial Coevolution

- 2D toroidal lattice with one host (*h*) and one parasite (*p*) per site



Spatial Coevolution

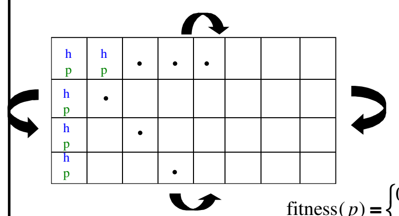
- 2D toroidal lattice with one host (h) and one parasite (p) per site



$\text{fitness}(\mathit{host}) = \text{fraction of 9 nearest-neighbor parasites answered correctly}$

Spatial Coevolution

- 2D toroidal lattice with one host (h) and one parasite (p) per site



$$\text{fitness}(p) = \begin{cases} 0 & \text{if } h(p) \text{ is correct} \\ > 0 & \text{if } h(p) \text{ is not correct} \end{cases}$$

$\text{fitness}(\mathit{host}) = \text{fraction of 9 nearest-neighbor parasites answered correctly}$

	Cellular Automata
Traditional GA (with crossover)	20%
Traditional GA (mutation only)	0%
Coevolution	0%
Spatial Coevolution	67%

Percentage of successful runs

What types of applications might coevolutionary learning be good for?

Computer Immunlogy

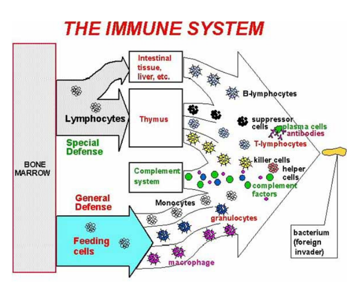


Figure 20. Immune system block diagram.

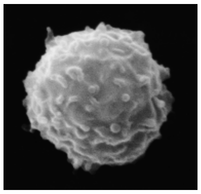
http://www.cheniere.org/images/rife/rife20.jpg

- **Primary response:** Antigen has not been “seen” before by the immune system.
- **Secondary response:** Antigen has been seen; immune system has antibodies well-matched to this specific antigen.

- **Primary response**
  - Need to respond to invaders that are present while still monitoring for new invaders
  - Not much prior knowledge about what invaders will look like.

Detection of Pathogens via Affinity Maturation

- Trillions of lymphocytes continually circulate in blood and tissues, with continual turnover of lymphocytes.



B-lymphocyte

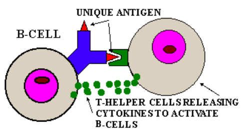
http://www.miltenyibiotec.com/macs/products/human/b/911-51.htm

- A lymphocyte’s surface is covered with identical receptors that bind to a range of molecular shapes with a range of affinities.
- Each individual lymphocyte is born with a set of unique receptors, due to random shuffling of variable gene libraries.

Need pictures to illustrate the two bullet points

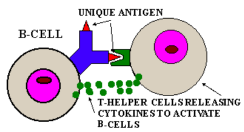
- Continual random variation in receptors and individual receptor’s range of affinities:

→ Good coverage of huge space of possible pathogen shapes.



http://www.slic2.wsu.edu:82/hurlbert/micro101/images/101ThwithBcell16.gif

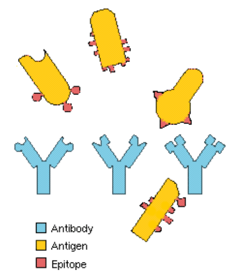
- Lymphocyte binding with antigen; range of possible affinities.
- Receptors can even self-adjust to make an existing bond stronger.



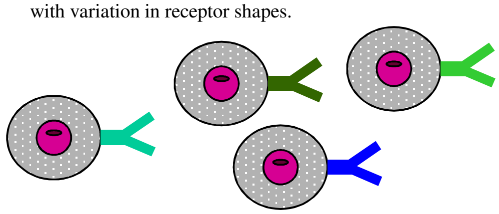
http://www.slic2.wsu.edu:82/hurlbert/micro101/images/101ThwithBcell16.gif

If number of strongly bound receptors exceeds a threshold, and lymphocyte gets "go-ahead" signal from helper T-cells with similarly bound receptors, then lymphocyte is "activated".

- Activated lymphocytes secrete antibodies, which bind to pathogens, neutralize them, and mark them for destruction by other cells.



- Activated lymphocyte migrates to lymph node, and divides rapidly, producing large numbers of daughter lymphocytes with variation in receptor shapes.



- Daughters are subject to selection, depending on affinity for antigens. Best-matching new lymphocytes themselves produce the most offspring cells.

- How to avoid recognizing and destroying self?
  - Lymphocytes mature in isolated settings (thymus or bone marrow).
  - The ones that match self are killed off.
    - "Negative selection"
  - This is a statistical sampling technique.
  - Lymphocytes that survive are allowed to enter the blood stream.

Immune-system inspired change detection algorithm

(Forrest, Perleson, et al.)

- Generate set of detectors (e.g., bit strings) at random
  - enough to have "coverage"
- Let them encounter "self"
  - Fragments of operating system, other code.
  - Kill off those detectors that match self sufficiently strongly.
- Use surviving detectors to monitor protected code.
  - If match is found at later time, assume something has changed.

m	r	l	P_M
2	8	32	0.0502023
2	8	64	0.108697
2	8	128	0.2151
2	8	256	0.391316
2	16	32	0.000137329
2	16	64	0.000381437
2	16	128	0.000869474
2	16	256	0.00184483
128	8	32	$3.33067 * 10^{-16}$
128	8	64	$7.77156 * 10^{-16}$
128	8	128	$1.66533 * 10^{-15}$
128	8	256	$3.44169 * 10^{-15}$
128	16	32	~ 0.0
128	16	64	~ 0.0
128	16	128	~ 0.0
128	16	256	~ 0.0

Table 1: Example values of P_M for varying values of m (alphabet size), r (number of contiguous matches required for a *match*), and l (string length).

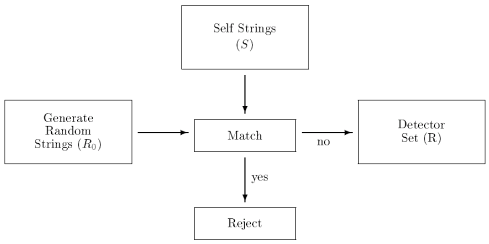


Figure 1: Generation of Valid Detector Set (Censoring) .

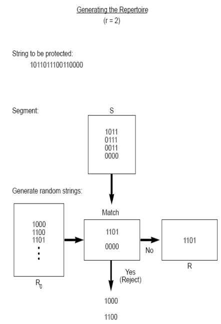


Figure 4: Generating the repertoire. The string to be protected is logically segmented into four equal-length "self" strings (stored in S). To generate the repertoire, random strings are produced in the box labeled R_0 and matched against each of the self strings. The first two strings, 1000 and 1100, are eliminated because they both match self string 0000 at at least two contiguous positions. The string 1101 fails to match any string in self at at least two contiguous positions, so it is accepted into the repertoire (box labeled "R").

r	N_R	N_{R_0}	P_f
9	2	1435(1150.53)	0.270(0.044)
9	5	3229(1104.72)	0.111(0.074)
9	8	5910(1864.25)	0.010(0.010)
9	10	7274(2580.88)	0.000(0.000)
10	5	182 (71.67)	0.150(0.036)
10	8	315 (126.09)	0.040(0.020)
10	10	382 (111.66)	0.020(0.014)
10	15	598 (161.29)	0.020(0.014)
10	25	996 (211.11)	0.000(0.000)
13	25	54 (7.18)	0.140(0.035)
13	50	86 (8.36)	0.110(0.031)
13	100	170 (12.35)	0.010(0.010)
13	125	205 (0.00)	0.000(0.000)

Table 4: Probability of failing to detect modification to more.com when infected by timid virus. $N_S = 655$. String Length = 32. r is match threshold. N_R is the number of detectors. N_{R_0} is the Initial repertoire size. N_M is the number of non-self strings. $N_M \approx 76$. P_f is the observed probability of failing to detect the virus.

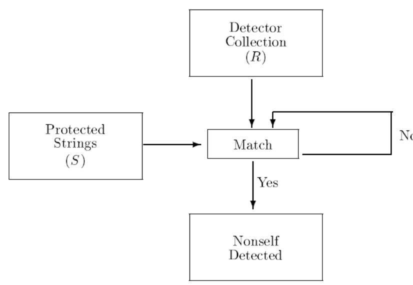


Figure 2: Monitor Protected Strings for Changes.

Intrusion Detection

(Forrest, Hofmeyr, et al.)

Example of "normal system calls" for a given program:

open, read, mmap, mmap, open, read, mmap

Create database of trigrams ("self"):

open, read, mmap
read, mmap, mmap
mmap, mmap, open
mmap, open, read

Monitor trigrams of ongoing system calls:

if enough mismatches with "self", suspect intrusion

Example

Self:

open, read, mmap
read, mmap, mmap
mmap, mmap, open
mmap, open, read

Observed system calls:

open, read, mmap, mmap, open, mmap, mmap

Observed trigrams:

Program	Number Mismatches	% Mismatches	$\hat{S}_A$
ls	42	75	0.6
ls -l	134	91	1.0
ls -a	44	76	0.6
ps	539	97	0.6
ps -ux	1123	99	0.6
finger	67	83	0.6
ping	41	57	0.6
ftp	271	90	0.7
pine	430	77	1.0

Table 3. Distinguishing `sendmail` from other programs. Each column reports results for a single anomalous measures: Mismatches (column 2), percentage of mismatches over a trace (column 3), and $\hat{S}_A$ (column 4). The results shown are for a sequence length of $k = 10$. There are no mismatches against `sendmail` itself because the database includes all variations.

Network intrusion detection and other "anomalies" (e.g., distributed "denial of service" attacks):

- Local-area broadcast network:
  - Every computer on the network sees every packet passing through the network
  - Communication:
    - Pair-wise connection between computers in the LAN or between outside computer and computer in the LAN
  - Pair-wise connection represented by triple:
    (source-I, destination-IP, communication port)
  - Encoded as 49-bit string that unambiguously defines the connection

From Forrest & Hofmeyr, Immunology as Information Processing

Stephanie Forrest and Steven A. Hofmeyr 377

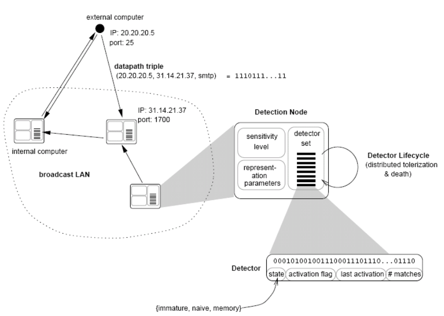


FIGURE 3 Architecture of the artificial immune system.

- Artificial immune system:
  - "Self":
    - Normal pair-wise connections between computers
  - Non-self":
    - All other all other possible pair wise connections
  - Detectors:
    - 49-bit strings with small amount of local state
    - States: immature cell, mature but naive cell, memory cell