Reverse-engineering and Exploiting Radars with **EVOLUTIONARY COMPUTING**

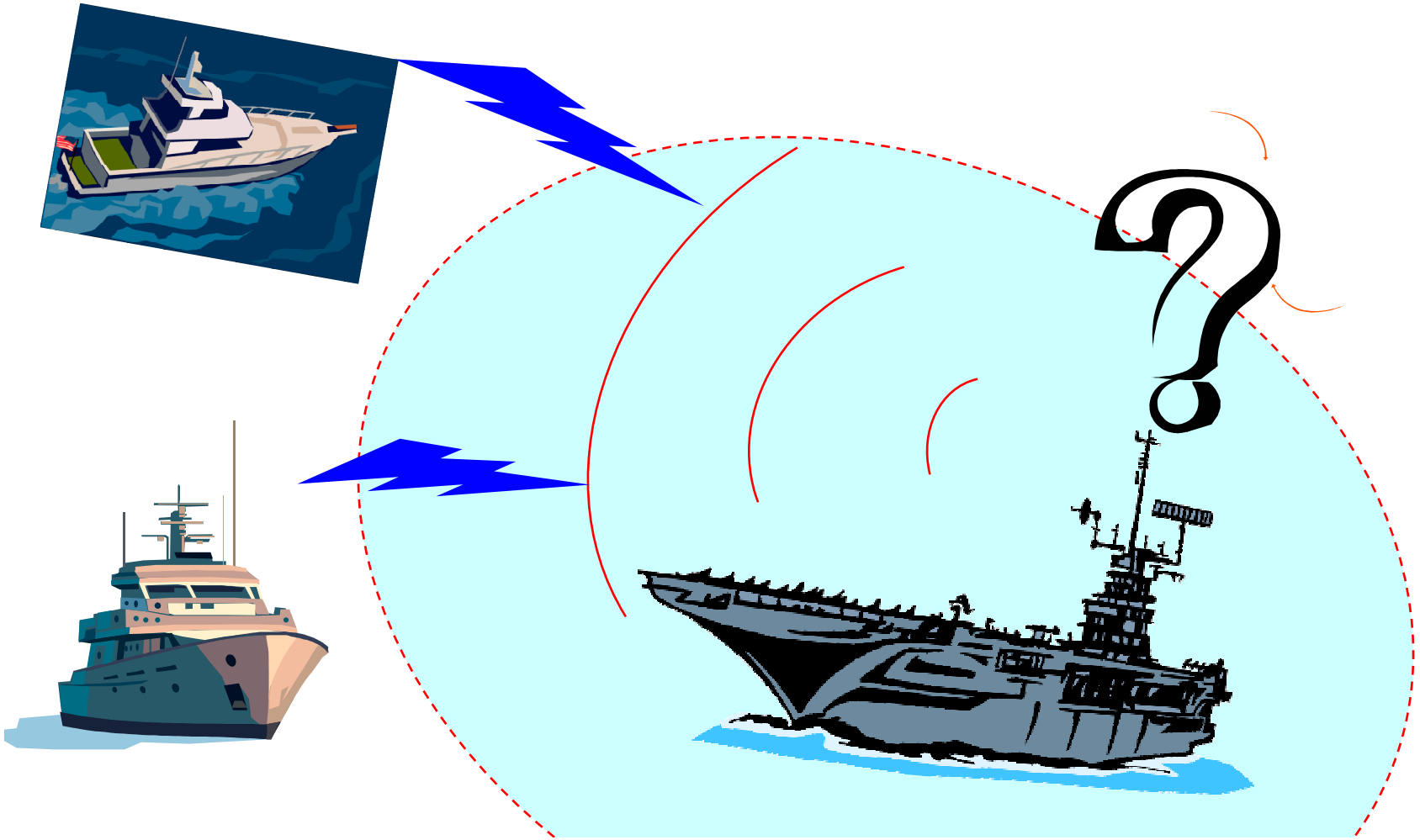
CompSci Graduate Co-Op

ThanhVu H. Nguyen

Supervisor: ***Dr. James F. Smith, III***

Naval Research Laboratory,
Washington, DC

GOAL: confuse the radar to approach the platform without being detected



Problem Statement

- Determine design specifications and design flaws for a system for which

i) there are no design specifications present

ii) invasive study is difficult or impossible

iii) the system may not be disassembled

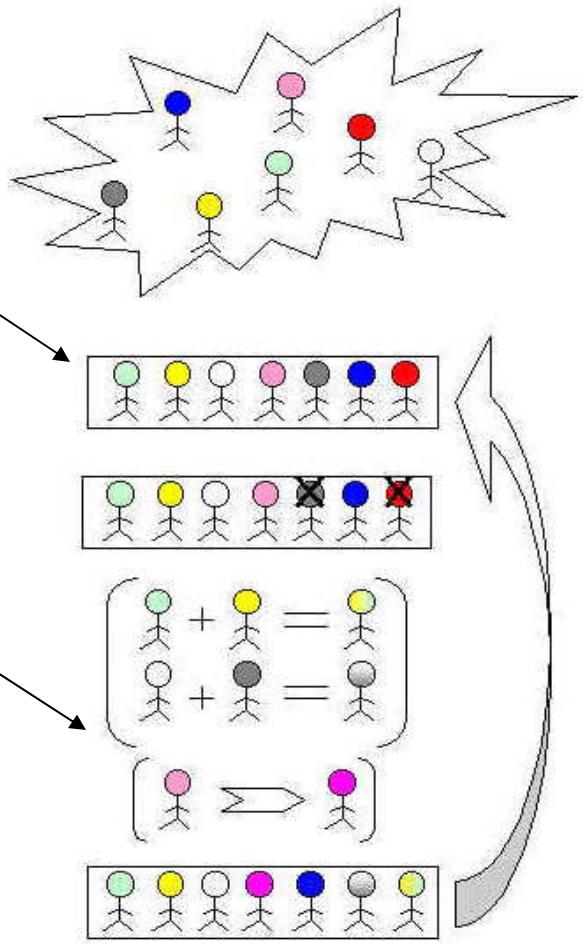
Approach

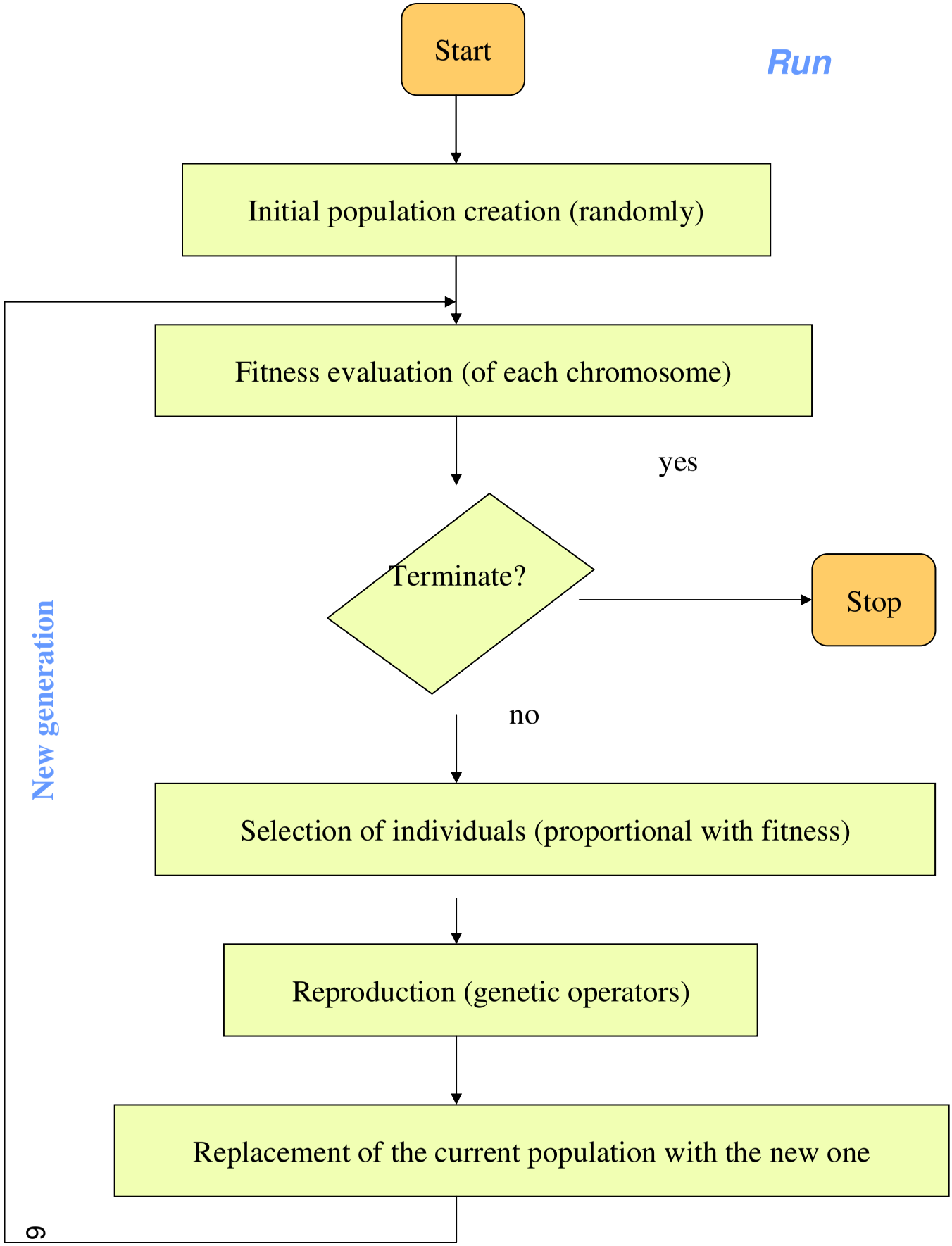
- Reverse-engineer the unknown system (digital logic) with **Genetic Programming**
- Exploit the system with **Genetic Algorithm**

*To **Defeat** the enemy, you must **Know** the enemy*

Evolutionary Algorithms

- **Darwin's theory of evolution**
  - Natural selection/survival of the fittest [Selection]
  - Reproduction by [recombination/Crossover] and [mutation]
- Formulated as the basis of almost all evolutionary algorithms: **GA**, **GP**

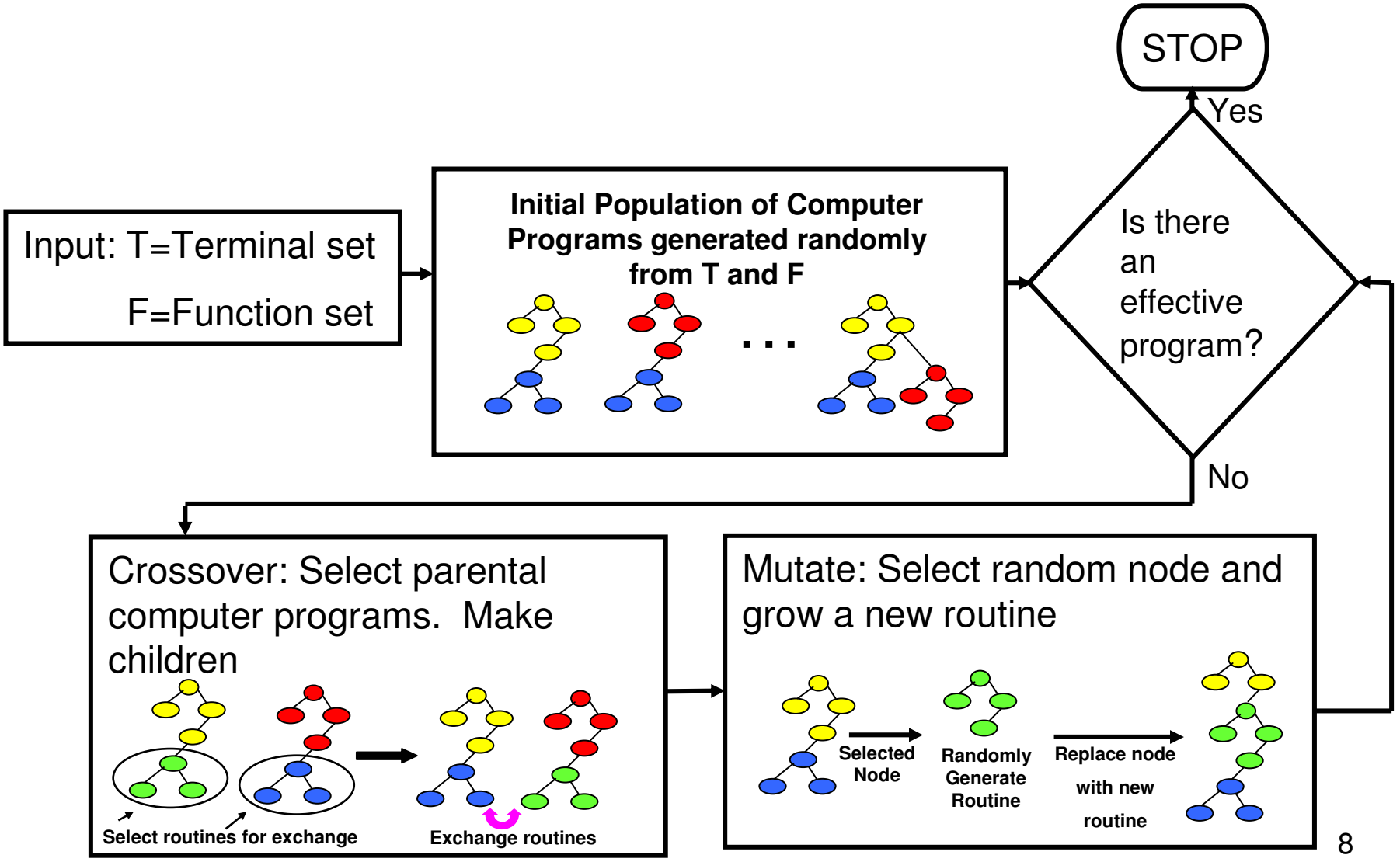



Start
Run
Initial population creation (randomly)
Fitness evaluation (of each chromosome)
yes
Terminate?
Stop
no
New generation
Selection of individuals (proportional with fitness)
Reproduction (genetic operators)
Replacement of the current population with the new one

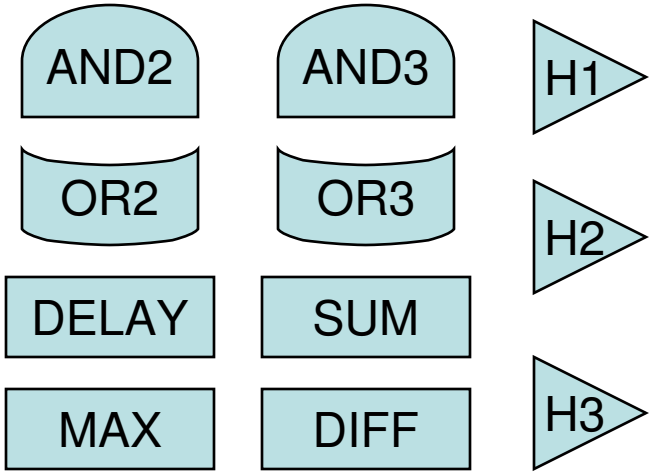
Reverse-Engineering with Genetic Programming

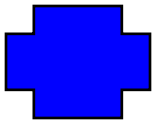
- Data mine the design specification of the system using a data base consisting of
  - Recorded system inputs and output measurements
- Incorporate rules provided by experts into the data mining process
- Use a genetic program (**GP**) as a symbolic data mining function to data mine the digital logic

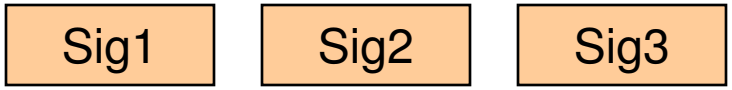
Genetic Program Overview



functions

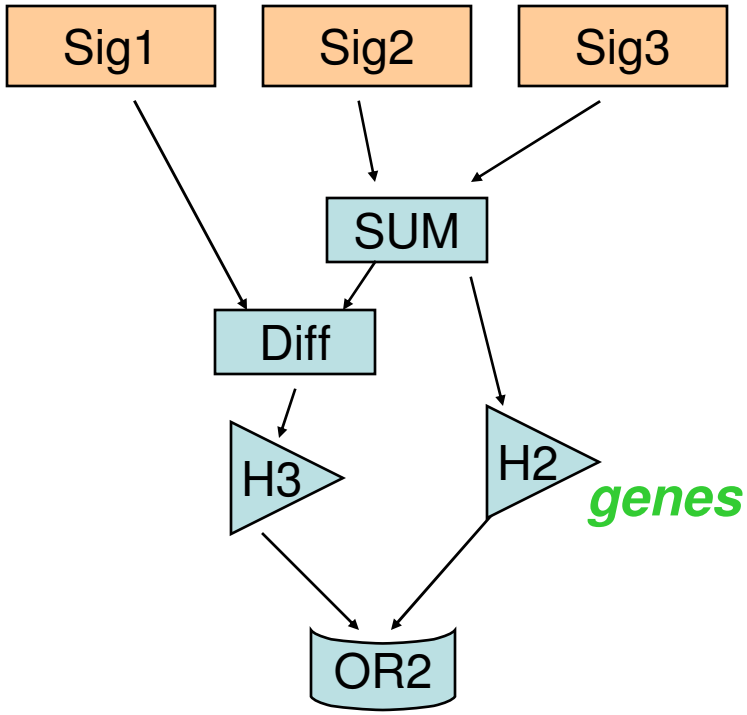






terminals

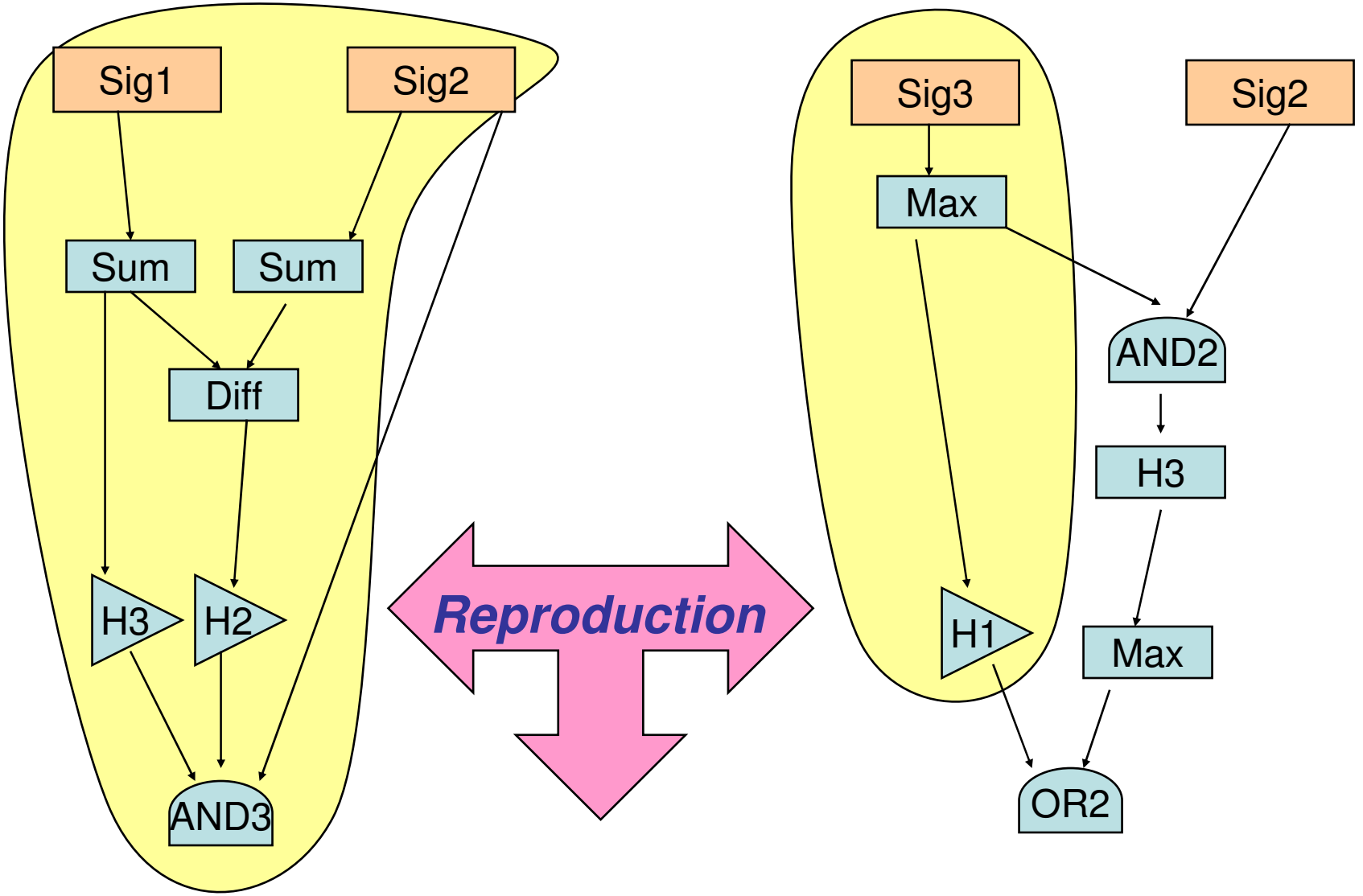
chromosome (tree scheme)



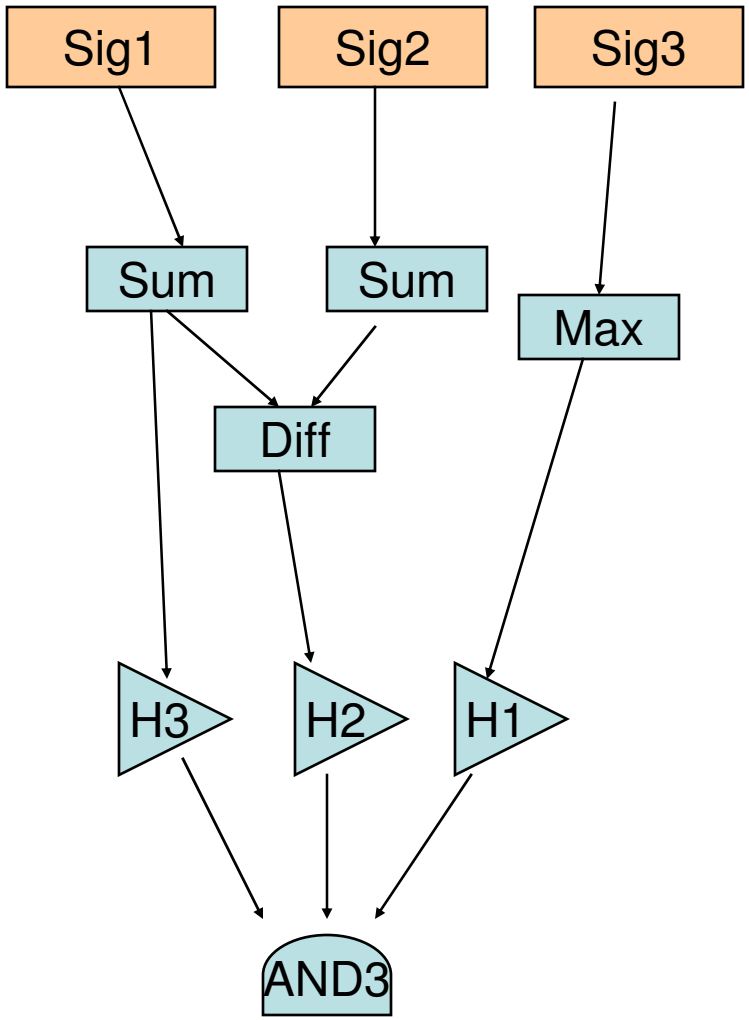
genes

chromosome (prefix scheme)

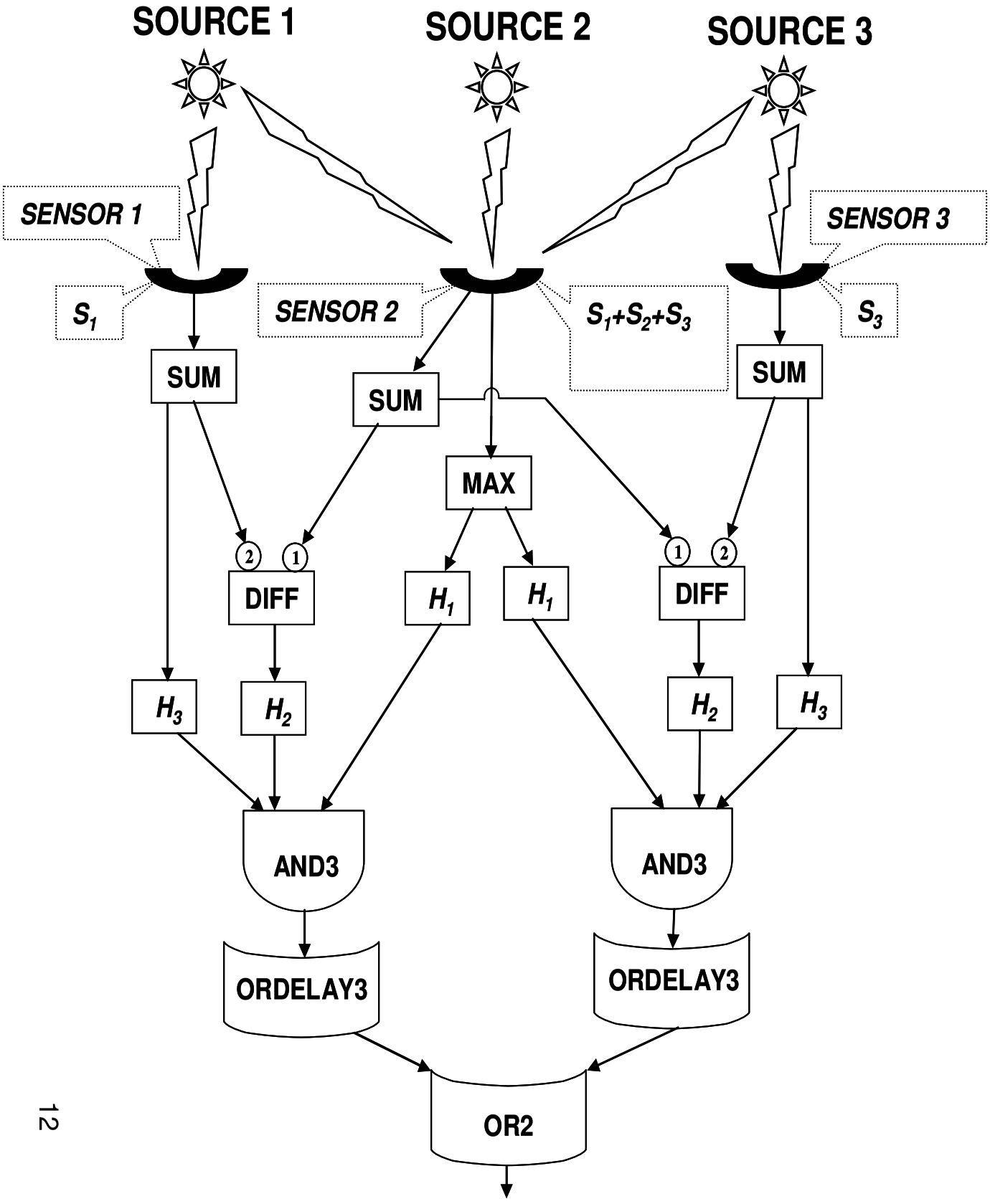
❖ ***OR2 H3 DIFF SIG1 SUM SIG2 SIG3 H2 SUM SIG2 SIG3***


Sig1
Sig2
Sum
Sum
Diff
H3
H2
AND3
Reproduction
Sig3
Sig2
Max
AND2
H3
H1
Max
OR2

- ***Children***
  - ***Contains 'good' genes from parents***

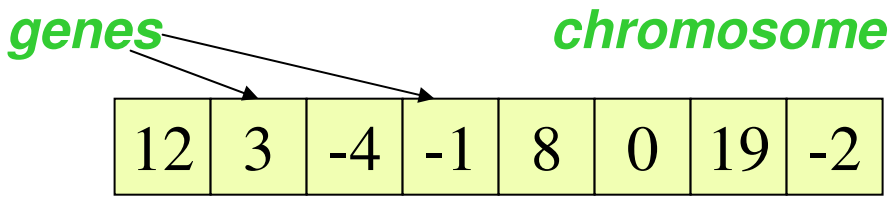


Digital Logic discovered by GP



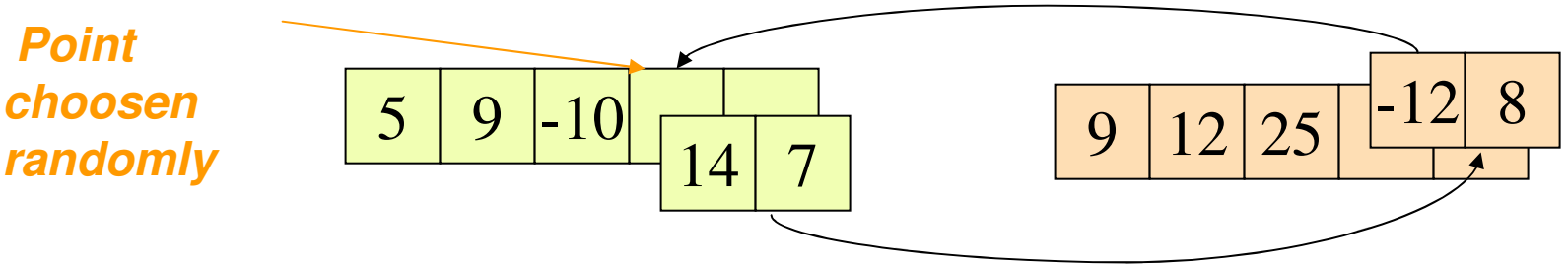
Genetic Algorithm Overview

❖ ***Encoding***

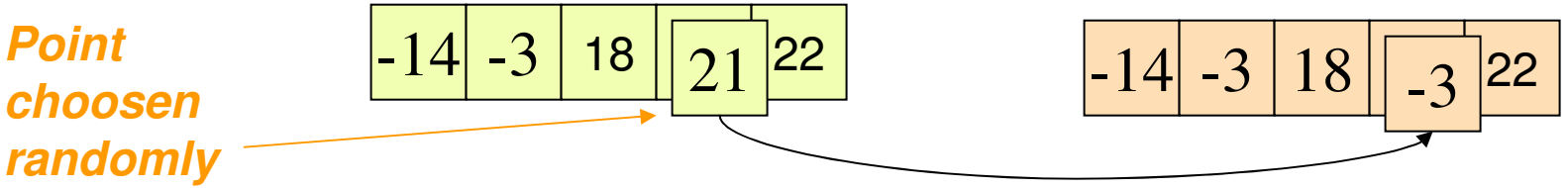


❖ ***Reproduction***

Recombination (crossover) – exchanges parts of two chromosomes



Mutation – changes the gene value



GA Based Automatic Defect Discovery

- A **GA** used the digital logic data mined by the **GP** to automatically discover defects.
- The DL plus expert rules were used to construct a fitness function for the **GA**
- A significant design flaw in the digital logic was discovered within 300 **GA** generations.

Summary

- A **genetic program** has been used as a data mining function to reverse engineer digital logic
- A **genetic algorithm** has been successfully used to automate the discovery of design defects
- **References**
  - James F. Smith, III and ThanhVu H. Nguyen "Data-mining-based automated reverse engineering and defect discovery", proc. SPIE vol. 5812, p. 232-242, data mining, intrusion detection, information assurance, and data networks security, 2005.
  - James F. Smith, III and ThanhVu H. Nguyen "Genetic program based data mining to reverse engineer digital logic", to be submitted