Next-Generation IoT & Intelligent Edge

Ch. 3-1 - Artificial Intelligence (Part I)

Yuan-Yao Shih (施淵耀)

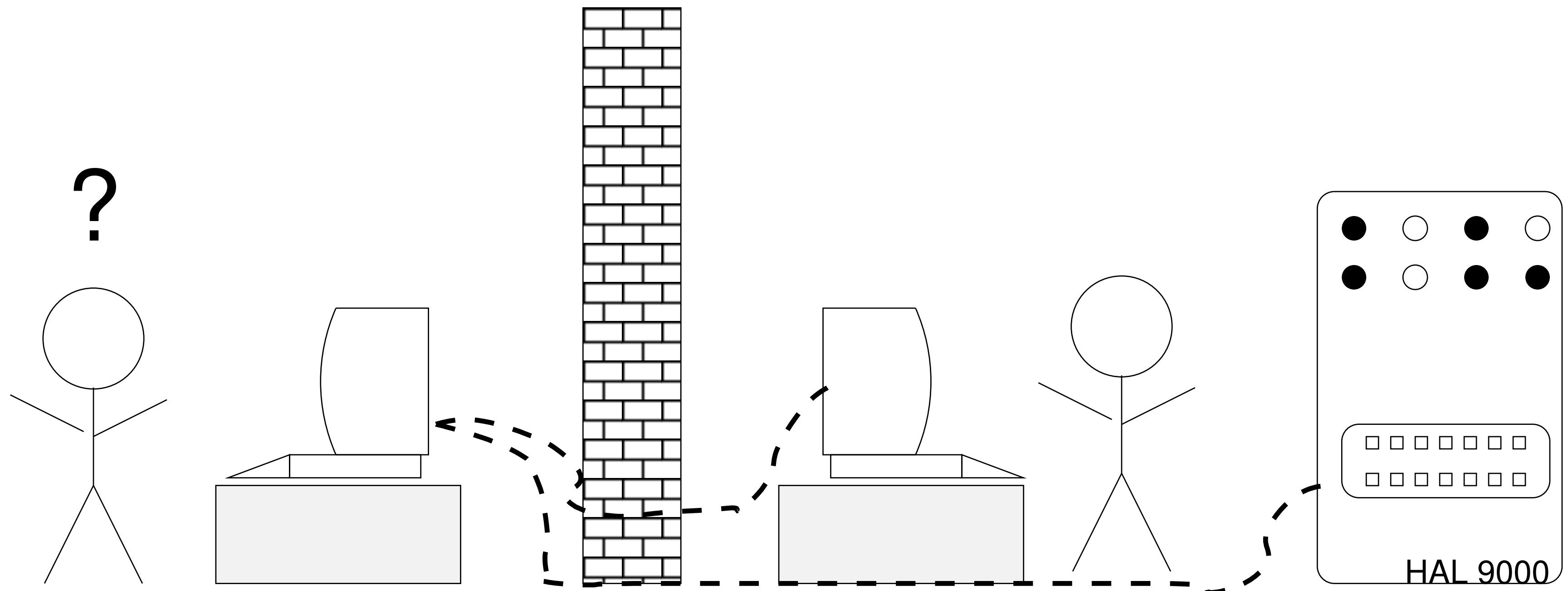
What is Artificial Intelligence?

- The attempt to make computers more “**intelligent**”
- The attempt to **better understand human intelligence**
- Four approaches:
  - - Is it about thought **thinking** . . .
  - - ... or **acting**?
  - - Oriented towards a **human** model (with all its defects) . . .
  - - . . . or **normative** (how should a rational being think/act)?

What is Artificial Intelligence?

	HUMAN	RATIONAL
THOUGHT	**Systems that think like humans**	**Systems that think rationally**
BEHAVIOUR	**Systems that act like humans** • “The art of creating machines that perform functions that require intelligence when performed by people.” (Kurzweil) • “The study of how to make computers do things at which, at the moment, people are better.” (Rich and Knight)	**Systems that act rationally**

The Turing Test



- You enter a room which has a computer terminal. You have a fixed period of time to type what you want into the terminal, and study the replies. At the other end of the line is either a human being or a computer system.

- If it is a computer system, and at the end of the period you cannot reliably determine whether it is a system or a human, then the system is deemed to be intelligent.

Systems that Act like Humans

- **The Turing Test approach**
  - A human questioner cannot tell if
    - there is a computer or a human answering his question, via teletype (remote communication)
  - The computer must behave intelligently
- **Intelligent behavior**
  - to achieve human-level performance in all cognitive tasks

Systems that Act like Humans

- These cognitive tasks include:
  - ***Natural language processing***
    - for communication with human
  - ***Knowledge representation***
    - to store information effectively & efficiently
  - ***Automated reasoning***
    - to retrieve & answer questions using the stored information
  - ***Machine learning***
    - to adapt to new circumstances

What is Artificial Intelligence?

	HUMAN	RATIONAL
THOUGHT	**Systems that think like humans** “The exciting new effort to make computers think ...machines with minds, in the full and literal sense.” (Haugeland, 1985) “[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning . . . ” (Bellman, 1978)	**Systems that think rationally**
BEHAVIOUR	**Systems that act like humans**	**Systems that act rationally**

Systems that Think Like Humans: Cognitive Modeling

- Humans as observed from ‘inside’
- How do we know how humans think?
  - Introspection vs. psychological experiments
- Cognitive Science
- Not important: Being able to solve problems correctly
- Important: **Being able to solve problems like a human would**

What is Artificial Intelligence?

THOUGHT	**Systems that think like humans**	**Systems that think rationally** • “The study of mental facilities through the use of computational models” (Charniak and McDermott) • “The study of the computations that make it possible to perceive, reason, and act” (Winston)
BEHAVIOUR	**Systems that act like humans**	**Systems that act rationally**
	HUMAN	**RATIONAL**

Systems that Think ‘Rationally’: ”Laws of Thought"

- Humans are not always ‘rational’
- Rational - defined in terms of logic?
- Problem:
  - Logic cannot express everything (e.g. **uncertainty**)
  - Logical approach is often **not feasible in terms of computation time** (needs ‘guidance’)
- → These are problems that appear regardless of the formalization method

What is Artificial Intelligence?

	HUMAN	RATIONAL
THOUGHT	**Systems that think like humans**	**Systems that think rationally**
BEHAVIOUR	**Systems that act like humans**	**Systems that act rationally** “Computational Intelligence is the study of the design of intelligent agents.” (Poole et al., 1998) “AI . . . is concerned with intelligent behavior in artifacts.” (Nilsson, 1998)

Systems that Act Rationally: “Rational Agent”

- Rational behavior: doing the right thing
- The right thing: that which is expected to **maximize goal achievement**, given the available information
- Giving answers to questions is ‘acting’.
- Do not care whether a system:
  - replicates human thought processes
  - makes the same decisions as humans
  - uses purely logical reasoning

Systems that Act Rationally: “Rational Agent”

- Rational agents (or rational actors)
  - A rational agent acts so as to achieve its given goals, under the assumption that its impressions of the world and its convictions are correct
  - **Rational acting -> Rational thinking** (<-X-)
- What to do, for example, when we must make a decision faced with **insufficient information**?

The Origins of AI

- Since the beginning, Philosophy, Mathematics, Psychology, Linguistics, and Computer Science have all
  - asked similar questions
  - developed methods and produced results for AI
- **The origins of AI (1943–1956)**: With the development of the first computing systems, people began to wonder, “Can computers copy the human mind? (Turing Test)”

The ‘von Neuman’ Architecture

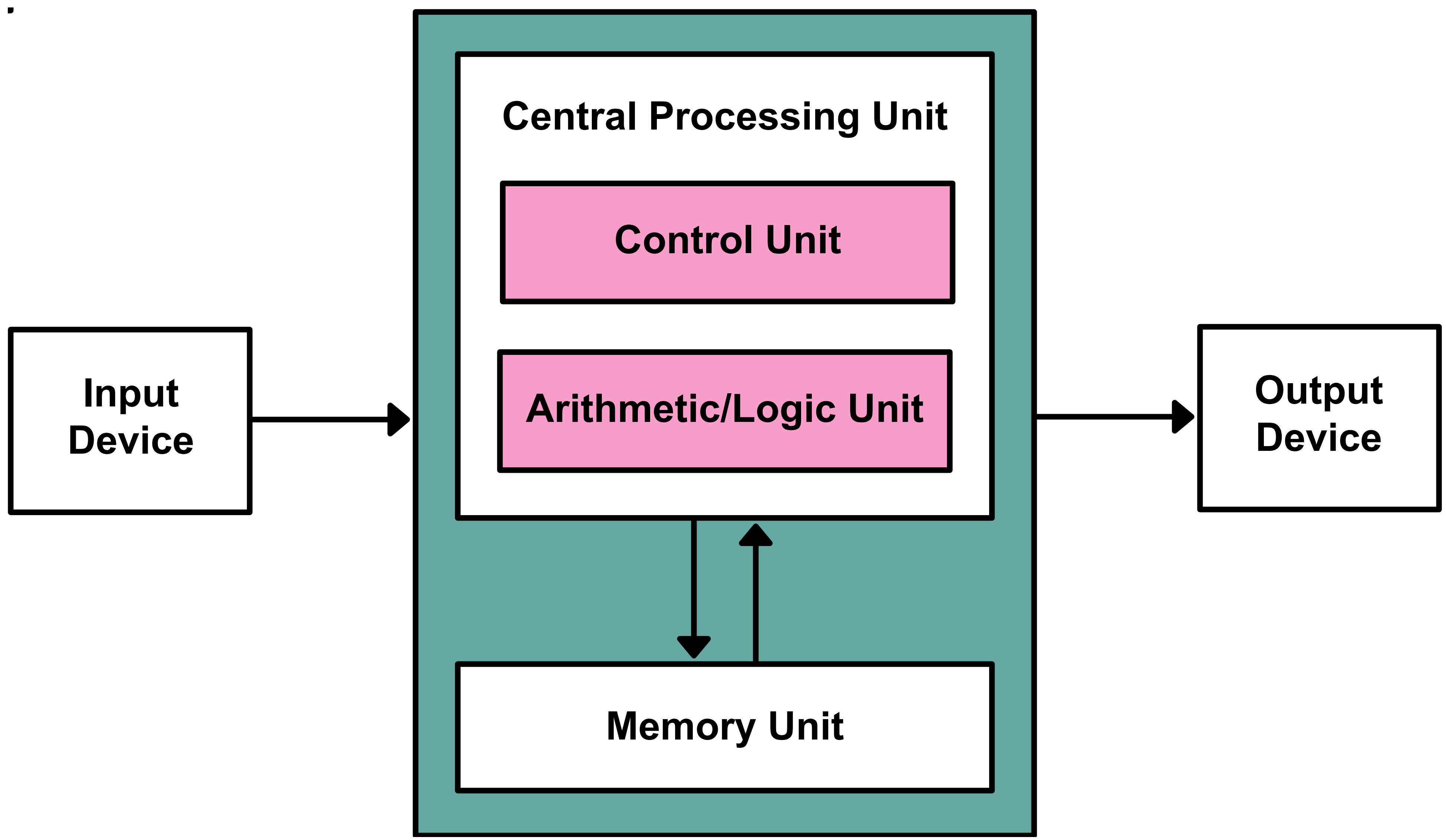


Figure Credit: Kapooht@Wikipedia

Periods in AI

- **Early period** - 1950’s & 60’s
  - Game playing
    - brute force (calculate your way out)
  - Theorem proving
    - symbol manipulation
  - Biological models
    - neural nets
- **Symbolic application period** - 70’s
  - Early expert systems, use of knowledge
- **Commercial period** - 80’s
  - boom in knowledge/ rule bases

Fashions in AI

Progress goes in stages, following funding booms and crises. Some examples:

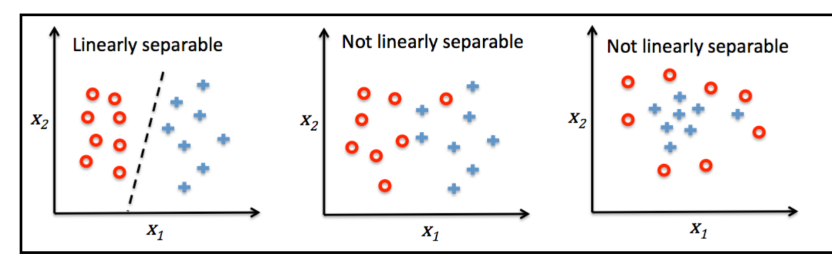
1. Machine translation of languages

1950's to 1966 - Syntactic translators

1966 - all US funding cancelled

1980 - commercial translators available

2. Neural Networks



1943 - first AI work by McCulloch & Pitts

1950's & 60's - Minsky's book on "Perceptrons"   stops nearly all work on nets

1986 - rediscovery of solutions leads to massive growth in neural nets research

Periods in AI - Recent

* **End of the 80's**: Expert systems prove less promising than imagined, (demystification of expert systems), end of the Fifth generation computer systems project, "AI winter"

* **90's**: Inclusion of probabilistic methods, agent-oriented vision techniques, formalization of AI techniques and increased use of mathematics in the field

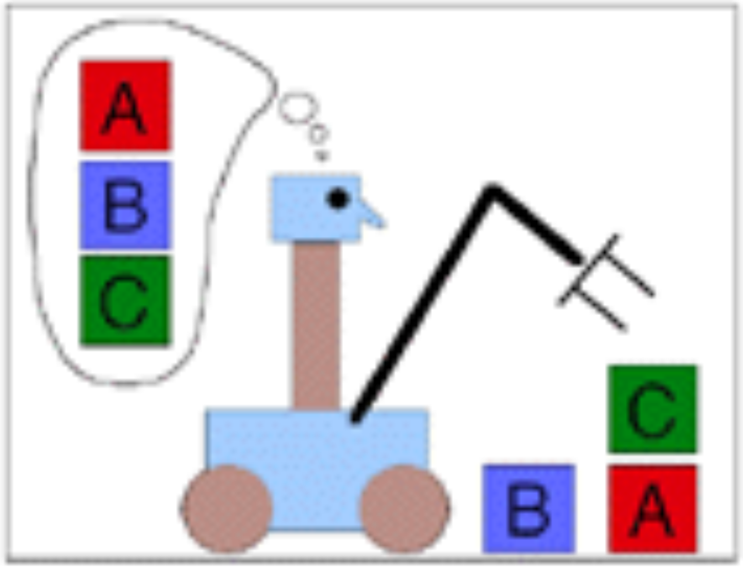
* **Today**, many methods are no longer regarded as pure AI methods. Examples: Board game programs, logic programming (PROLOG), search procedures, . . .

* **Deep learning** has become the new hype . . .

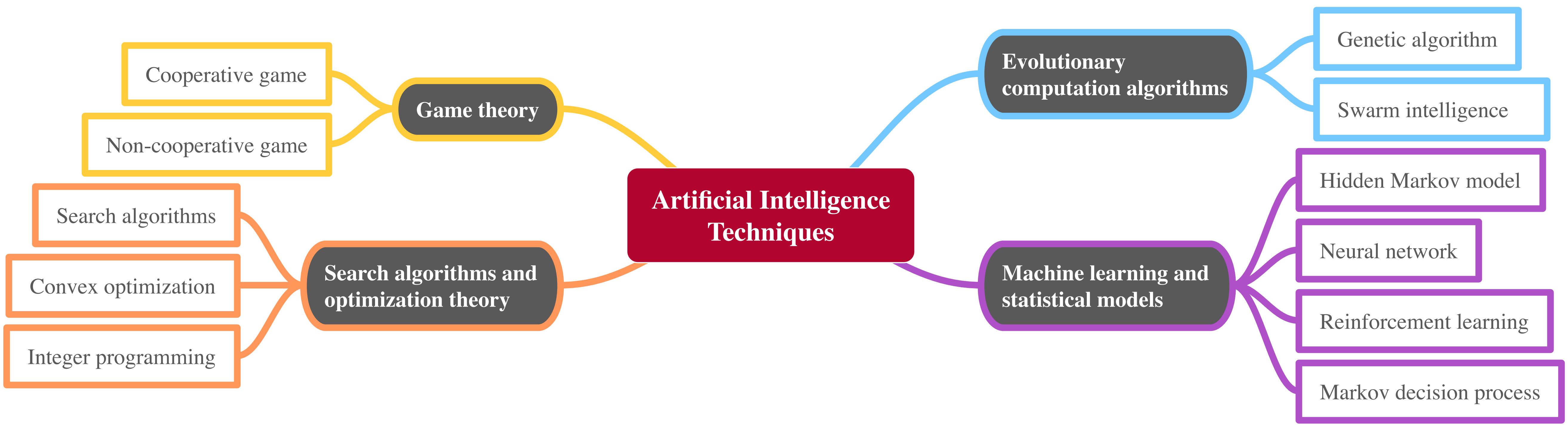
Goals of AI

- To make computers more useful by letting them take over dangerous or tedious tasks from human

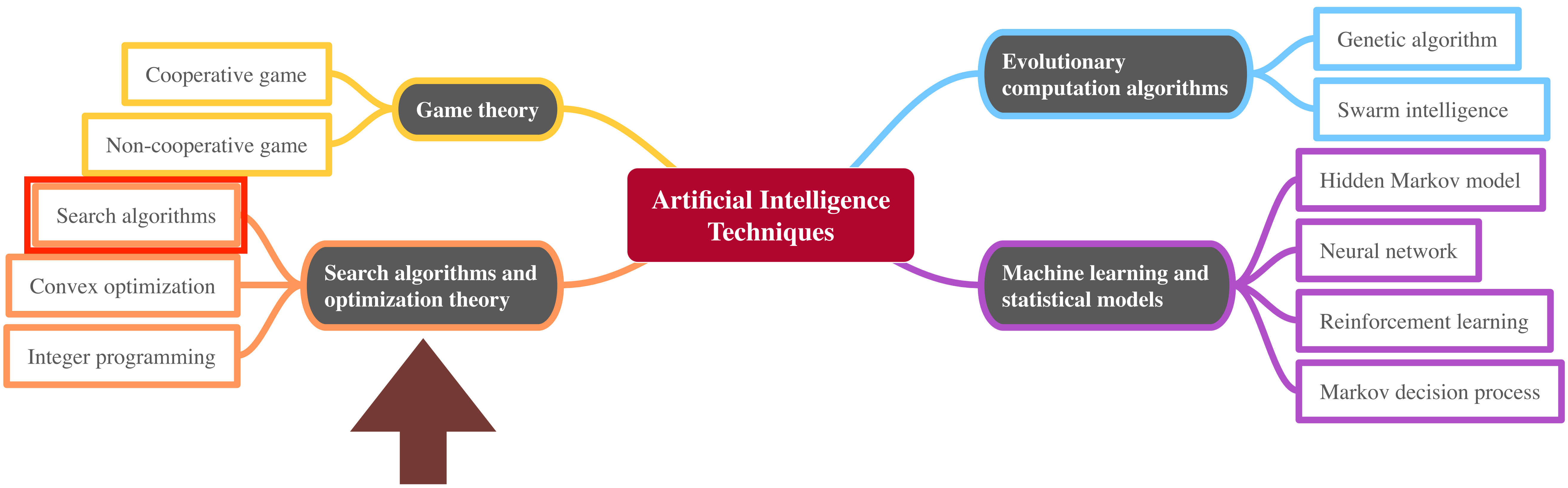
- Understand principles of human intelligence



The Main Topics in AI



The Main Topics in AI



Search Algorithms and Optimization Theory

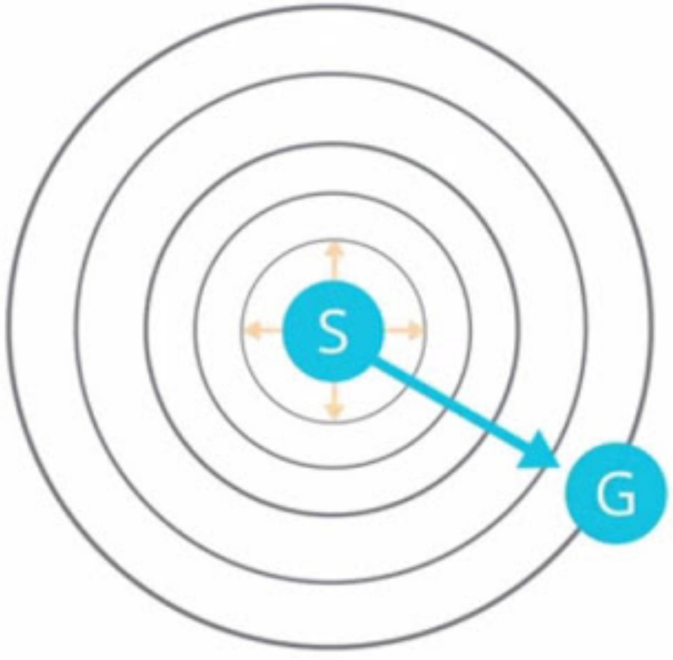
- *Search* is the fundamental technique of AI.
  - Possible answers, decisions or courses of action are structured into an abstract space, which we then search.
- Search is either “uninformed (blind)“ or “informed“:
  - **Uniformed search**
    - have no additional information about the distance from current state to the goal.
  - **Informed Search**
    - have additional information about the estimate distance from the current state to the goal.
- We may want to search for the first answer that satisfies our goal, or we may want to keep searching until we find the best answer.

Search Algorithms - Uninformed Search

* **Exhaustive Search (ES)**
  * A very intuitive problem-solving technique that **enumerates all possible candidates** for the solution which satisfy the problem's statement systematically.
  * Simple to implement, however, **its cost is proportional to the number of possible candidate solutions**, which may tend to proliferate as the size of the problem increases.
  * Typically used when the **problem size is limited**, or there are problem-specific heuristics that can be used to reduce the set of candidate solutions to a manageable size.

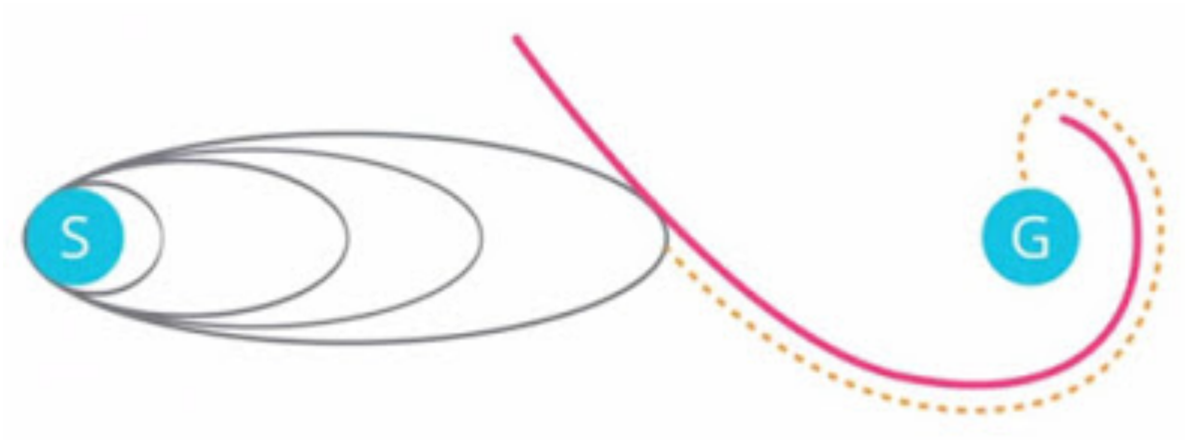
Search Algorithms - Uninformed Search

* **Breadth-First Search (BFS)**
    * An algorithm of traversing or searching strategy in **tree or graph data structures**.
    * can be applied to tackle many problems including finding the shortest path in a graph and solving puzzle games.
    * However, choosing the shallowest solution may not be the optimal one, **BFS can be optimal only if the path cost is a non-decreasing function for the depth of the node**.



Search Algorithms - Uninformed Search

* **Uniform-Cost Search (UCS)**
  * Can be useful if **all the steps in the search graph do not have the same cost**, where BFS may not find an optimal solution.
  * UCS **expands the node with the lowest path cost**, which can be done via storing the frontier as a priority queue ordered by the path cost of each node.
  * UCS is optimal in general; however, since it only cares about finding minimum path cost, instead of total path steps, **it may be stuck in an infinite loop** if there is a path with an infinite sequence of zero-cost actions.



Search Algorithms - Uninformed Search

* **Depth-first Search (DFS)**
    * Manages the list as a **last-in-first-out (LIFO)** stack while BFS manages the list as a first-in-first-out (FIFO) queue.
    * DFS is not guaranteed to find the optimal solution; however, **it provides a linear memory requirement solution with respect to the search graph**.

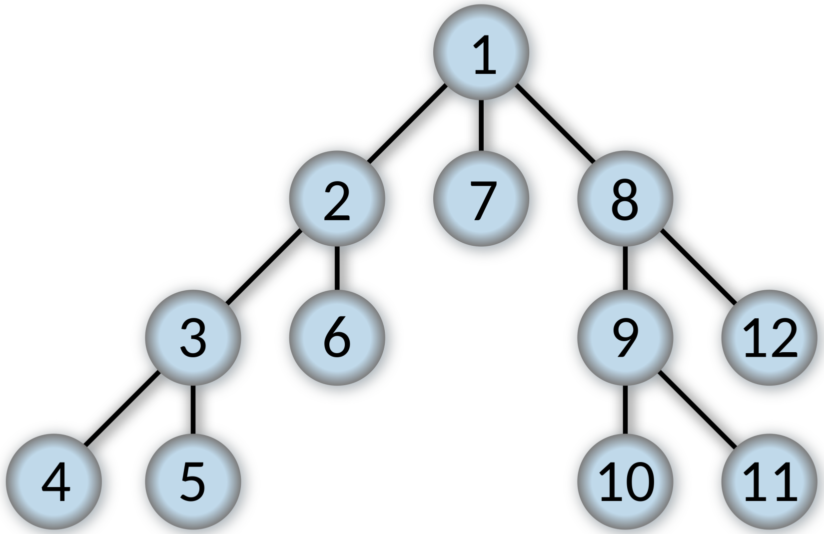


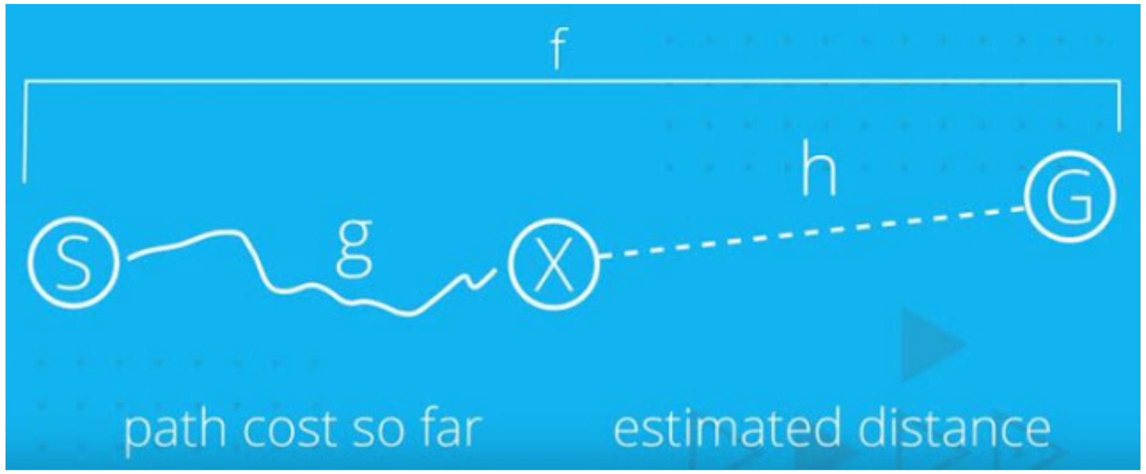
Figure Credit: Alexander Drichel@Wikipedia

Search Algorithms - Informed Search

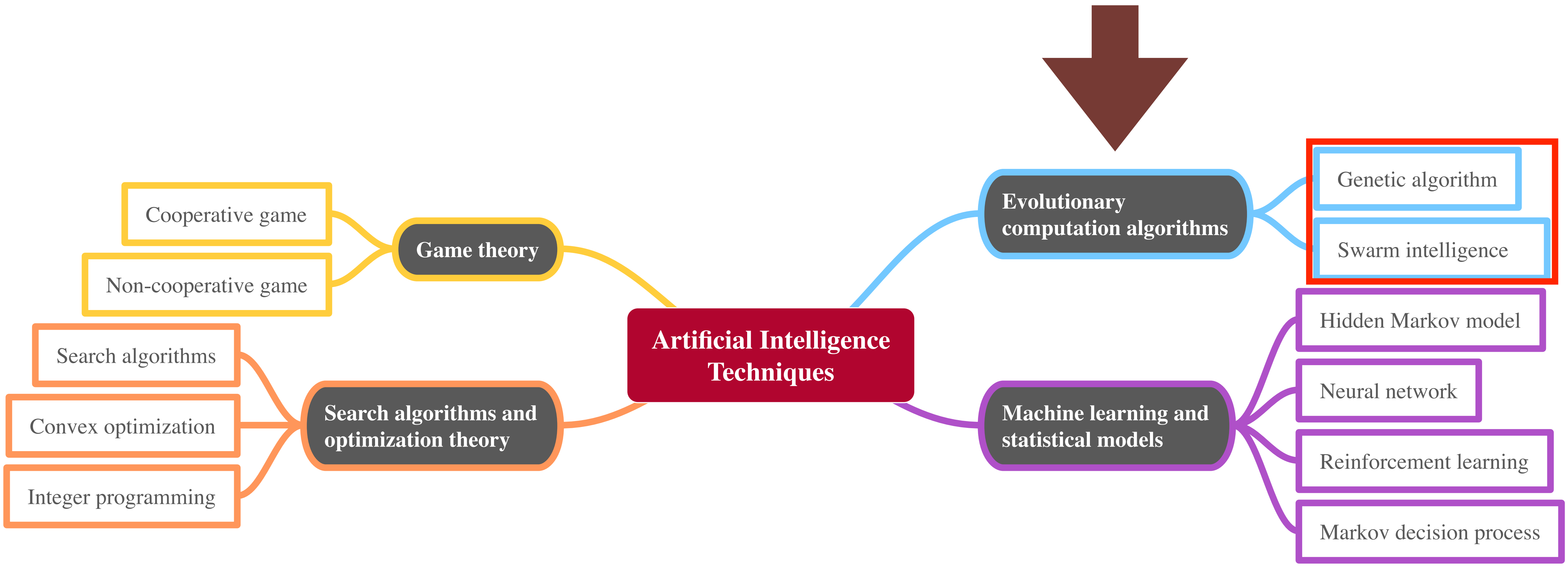
* **Greedy Search**
    * An algorithm that hopes to find a global optimum via **finding local optimum for each step**.
    * A general approach for **tree-search** or **graph-search** in which a node is selected for expansion based on an evaluation function, ***f(n)***, which can determine the search strategy.
    * The implementation of Greedy Search is identical to that for Uniform-Cost Search (UCS), except for the use of *f(n)* in Greedy search instead of path cost function in UCS to order.

Search Algorithms - Informed Search

* **A* Search**
  * Evaluates priority of traversing nodes by combining
    * ***g(n)***, the cost of reaching the node *n* from current state.
    * ***h(n)***, the estimated cheapest path cost of reaching the goal from the node *n* for the estimated cost of the cheapest solution through *n*.
  * choosing the node with the lowest value of *g(n) + h(n)*
  * is both complete and optimal under proper heuristic function *h(n)* and certain conditions.
  * Identical to Uniform-cost Search except that A* Search uses *g(n) + h(n)* instead of *g(n)*.



The Main Topics in AI

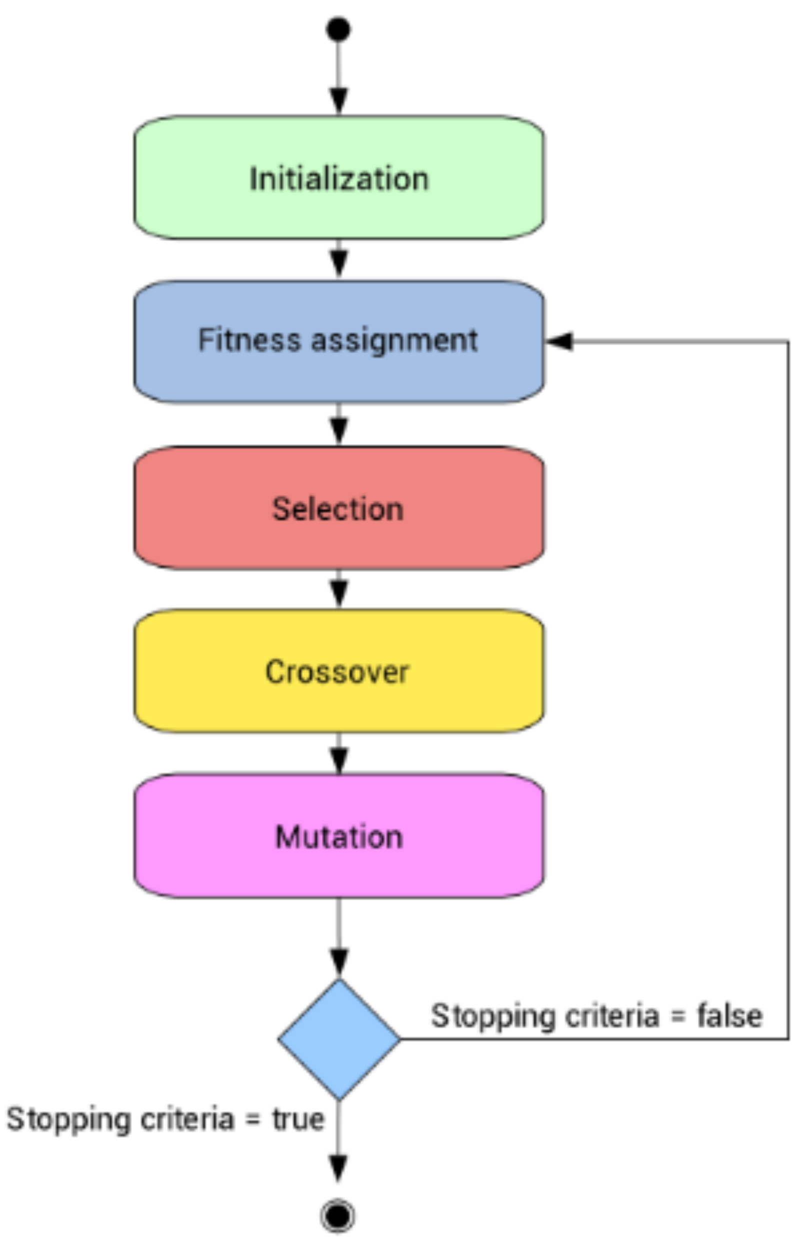


Evolutionary Computation Algorithm

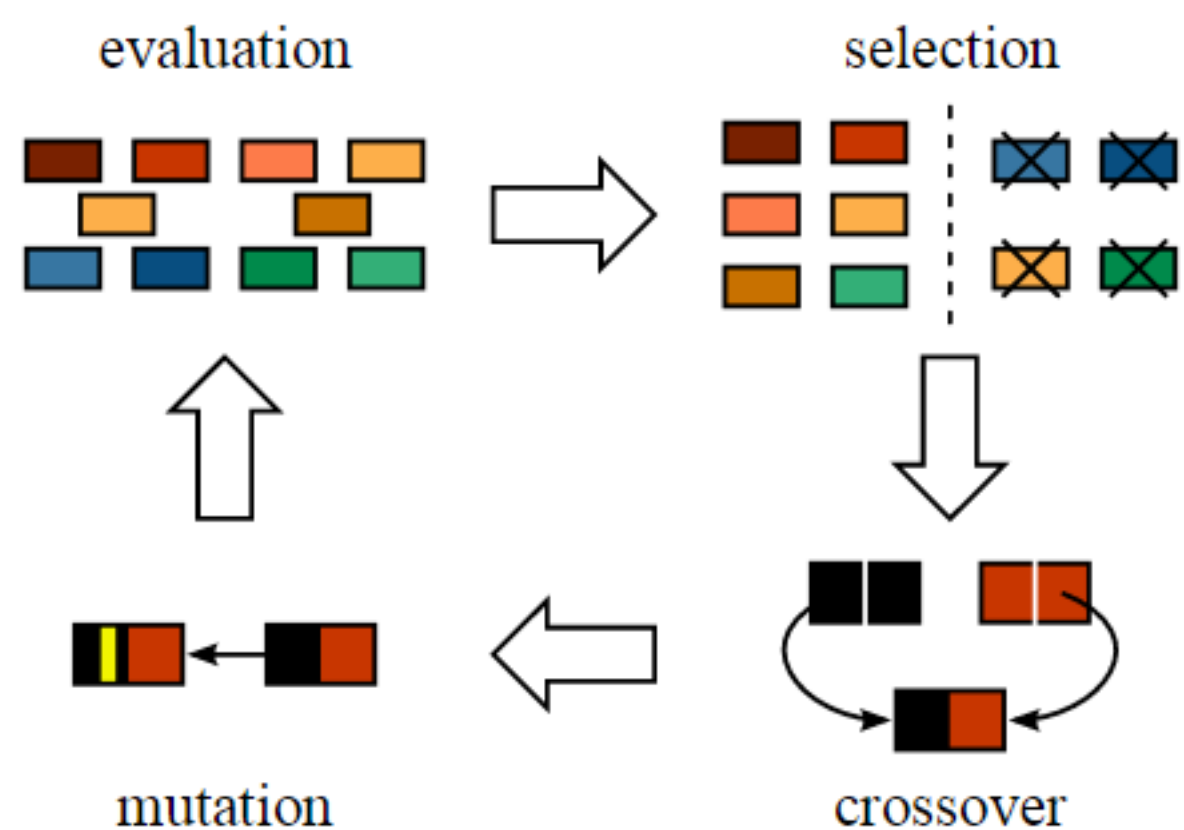
* A family of algorithms for global optimization inspired by **biological evolution**.
* **The main idea of evolution**: given a population of individuals, the environmental pressure causes natural selection and thus the fitness of the population is growing.
* We can easily match this idea on problem-solving: given an objective function to be optimized, randomly creating a set of candidate solutions and using the objective function as an abstract fitness measure.
* Based on this fitness, some of the better candidates are chosen to seed the next generation by applying **recombination** or **mutation** stochastically. This process is an optimization process in searching for solutions, and it will iterate until a solution is found or a previously set time limit is reached.
* Can be considered as alternative methods to previously mentioned search algorithms when the **network size is huge or the information about the network is limited**.

Evolutionary Computation Algorithm

* **Genetic Algorithm(GA)**



* Inspired by the process of natural evolution mechanism.
* Used to generate high-quality solutions to optimization and search problems.
* Tends to optimize the problem toward finding a better solution through this process and terminate when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

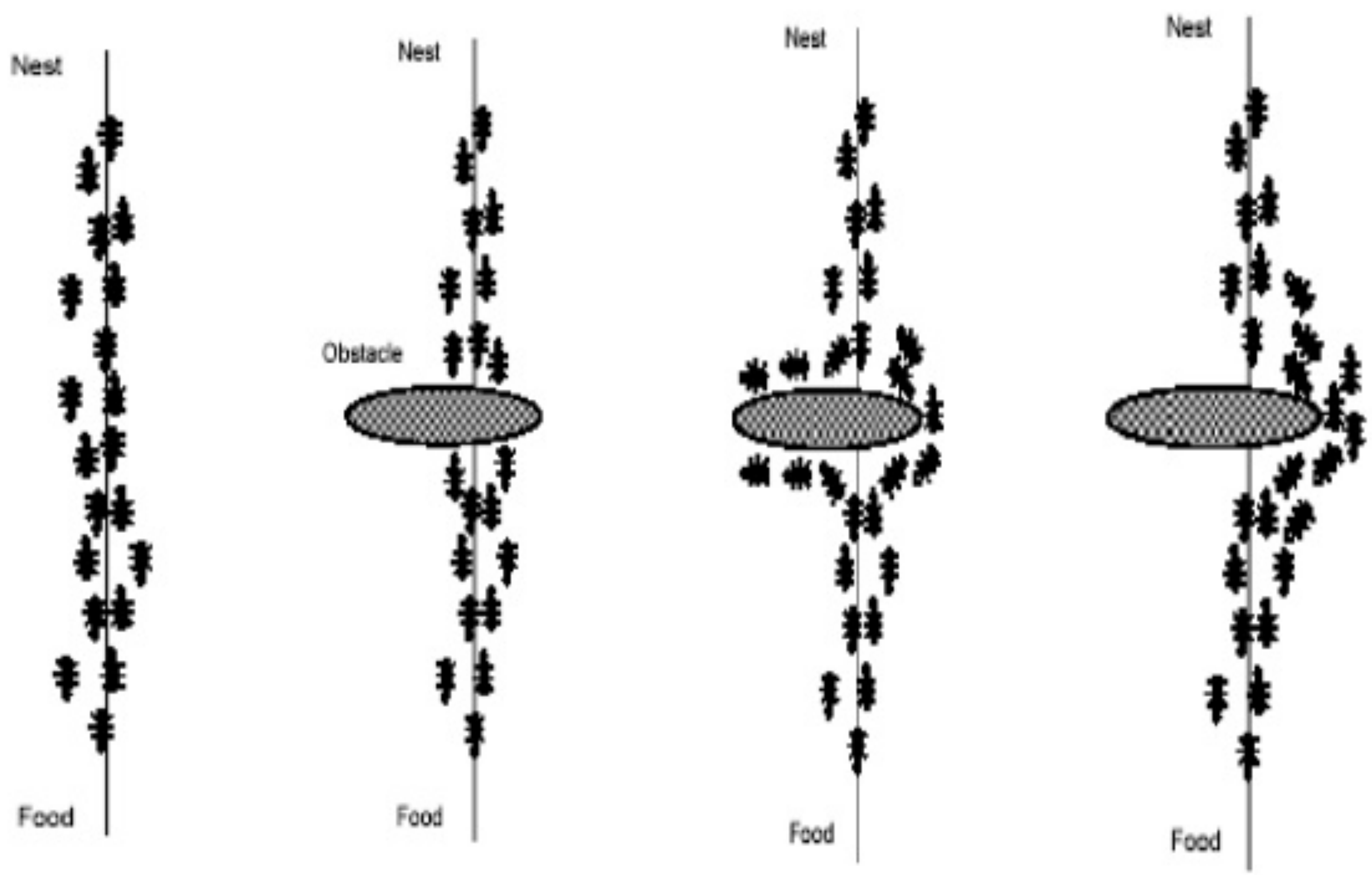


Evolutionary Computation Algorithm - SI

* **Swarm Intelligence (SI)**
    * Inspired from **the collective behavior of social insects** such as ants, termites, bees, and wasps, as well as from other animal societies such as flocks of birds or schools of fish.
    * The developed algorithms need to be **flexible** to internal and external changes, to be **robust** when some individuals fail, to be **decentralized** and **self-organized**.
    * Even though the single members of these colonies are non-sophisticated individuals, they can achieve complex tasks in cooperation.
    * Their coupling can have a wide range of characteristics, but there must be interaction among the units.

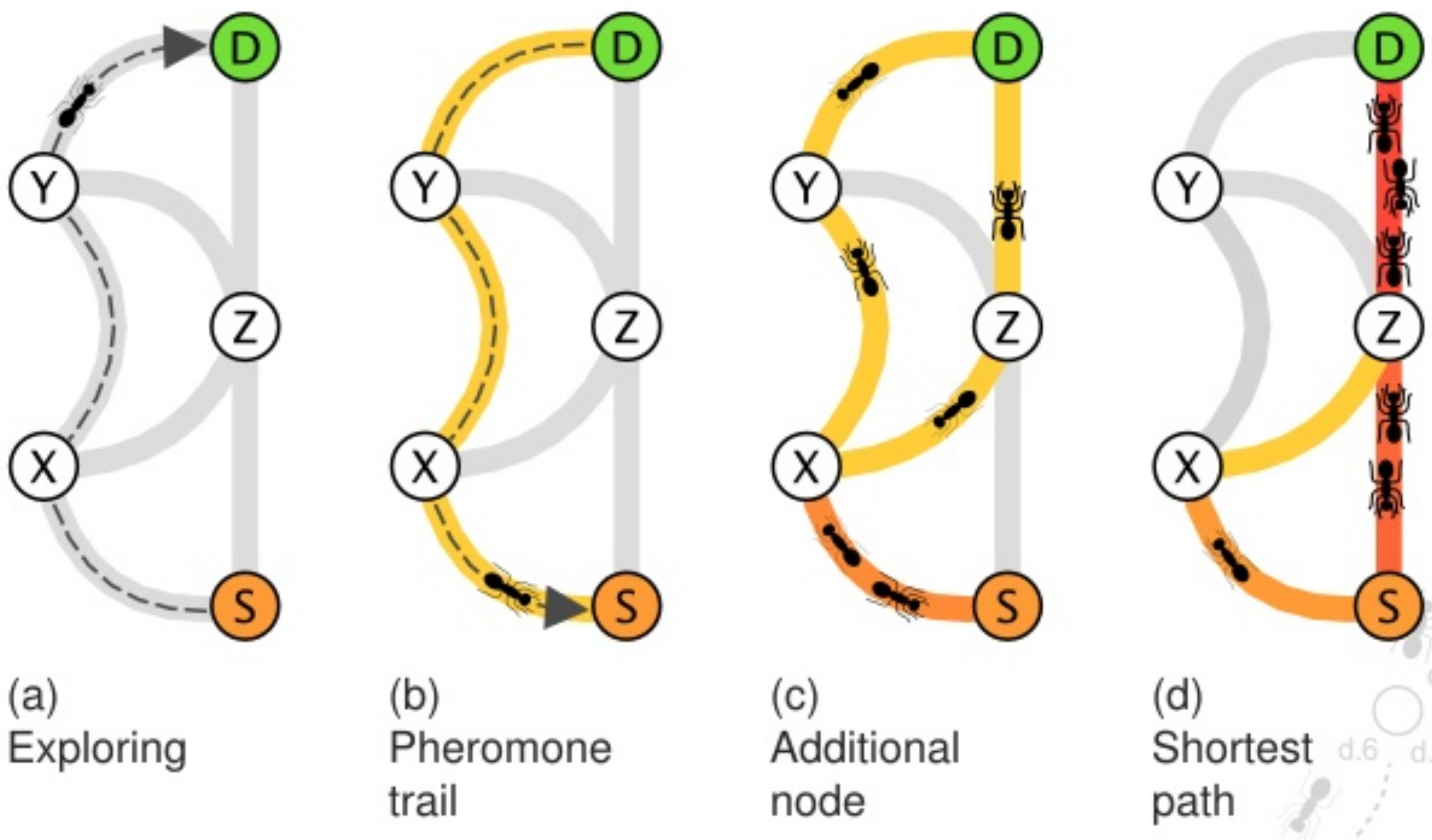
Evolutionary Computation Algorithm - SI

* **SI - Ant Colony Optimization (ACO)**
  * Inspired by the foraging behavior of ant colonies, that is, the **indirect communication between ants by means of chemical pheromone trails**, which enables them to find short paths between their nest and food
  * Can used to solve, for example, discrete optimization problems.



Evolutionary Computation Algorithm - SI

* **SI - Ant Colony Optimization (ACO)**



(a)
Exploring

(b)
Pheromone
trail

(c)
Additional
node

(d)
Shortest
path

Evolutionary Computation Algorithm - SI

* **SI - Particle Swarm Optimization (PSO)**
  * Solves a problem by having **a population of candidate solutions, called particles**, around in the search-space according to simple mathematical formulation over the particle's position and velocity.
  * Each particle's movement is influenced by their own best- known position in the search-space as well as the entire swarm's best-known position, that is, **each particle combines both self and swarm's experiences**.

Evolutionary Computation Algorithm - SI

* **SI - Particle Swarm Optimization (PSO)**
    * **Inspired by the movement of organisms in a bird flock or fish school**.
    * For example, a flock of birds is finding food over an area.
        * The one who is closest to the food via smelling chirps the loudest, then other birds will move to that bird's direction.
        * If another bird is closer to the food than the previous one, it will chirp even louder to attract others move toward it.
        * Thus, this process is expected to move the swarm toward the best solutions (the food).