COOPERATIVE AND ADAPTIVE ALGORITHMS

Spring 2020
Otman Basir

WEEK 1

COURSE INTRODUCTION

- Motivation
- Course Outline
- Course Objectives
- Course topics
- Introduction
    1. AI- Definition and Categorization
    2. Goals
    3. Rationality
    4. Intelligent Agents/Algorithms
    5. Applications

WHY BOTHER TAKING THIS COURSE?

I have done many programming courses, and

I know how to program in more than one programming language, and

I know algorithms and data structures, and

I have written many programs!

There are problems and there are problems!

There are problems that are Computable

There are many computational problems in our real life that are computable within a reasonable amount of time. For this type of problems we seek algorithms that can deterministically search for optimal solutions in reasonable time.

Computability is the ability to solve a **problem** in an effective manner.

We tend to speak of these types of problems as P or Polynomial problems.

There are problems and there are problems!

There are problems that are well-defined:

In a **well-defined problem,** the given state of the **problem,** the goal state of the **problem,** and the allowable operators (or moves) are each **clearly** specified

Traditional algorithms and programming techniques work well for solving computable well-defined problems.

Most likely, the types of problems you have tackled in the past.

There are problems and there are problems!

There are problems that are hard to compute.

We refer to this type of problems as NP problems, which stands for nondeterministic polynomial time. Which is the set of problems whose solutions can be verified in polynomial time.

Vast majority of NP problems whose solutions seem to require exponential time are what's called NP-complete, meaning that a polynomial-time solution to one can be adapted to solve all the others.

NP-complete problems are fairly common in real life, especially in large scheduling tasks. The most famous NP-complete problem, for instance, is the so-called traveling-salesman problem

HARD TO COMPUTE PROBLEMS

Other examples

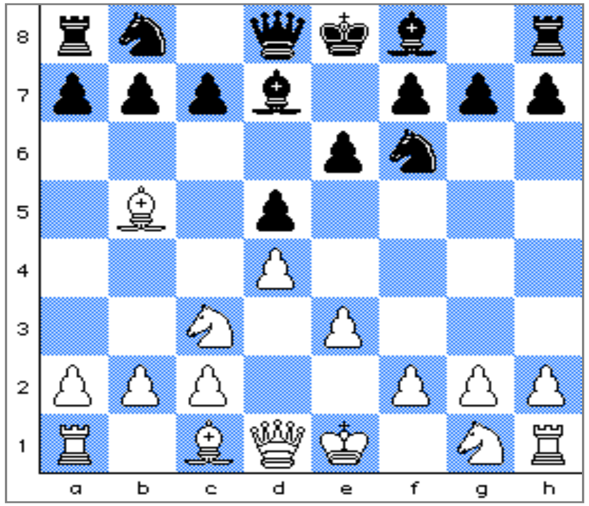
- communication
- transportation
- industrial production
- administration, writing, and bookkeeping
- technological advances / science
- entertainment

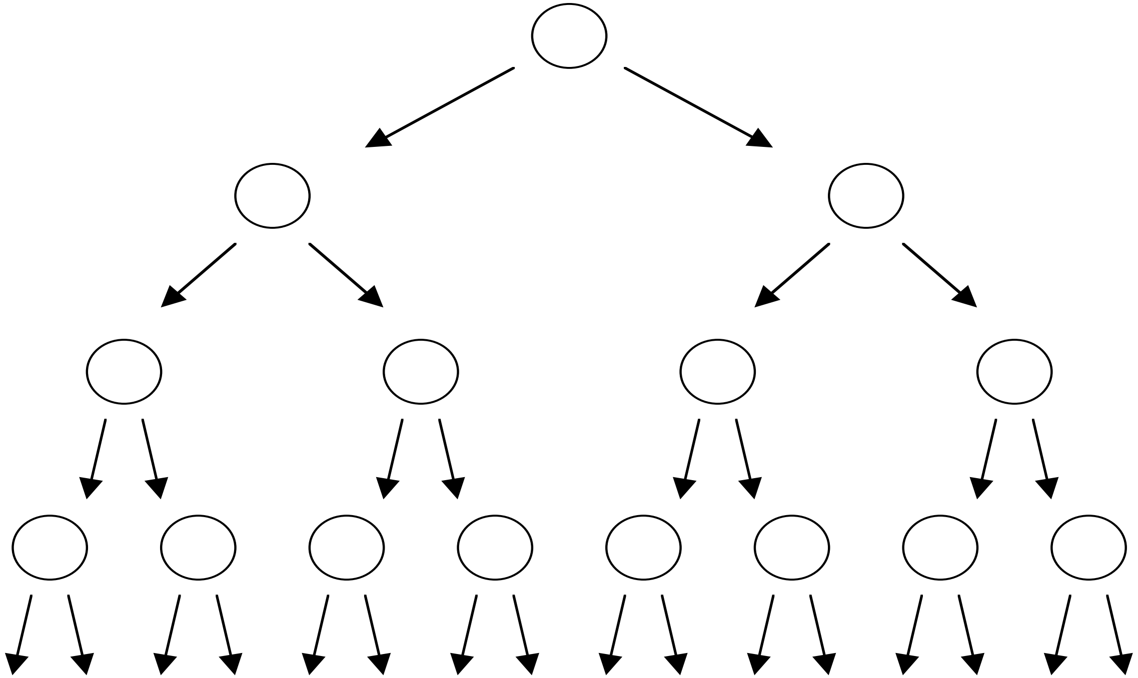
But many problems in these applications are complex enough

Hard problems

Well-defined, but computational hard problems

- NP hard problems (Travelling Salesman Problem)
- Action-response planning (Chess playing)





IT MAY SOUND AS A SIMPLE PROBLEM

the **four color theorem**, or the **four color map theorem**, states that, given any separation of a plane into contiguous regions, producing a figure called a *map*, no more than four colors are required to color the regions of the map so that no two adjacent regions have the same color. *Adjacent* means that two regions share a common boundary curve segment, not merely a corner where three or more regions meet

If you were to program a computer agent to 4-color a map for you, what general steps would you take?

Well defined, but hard to solve, NP-Complete.

THERE ARE PROBLEMS AND THERE ARE PROBLEMS!

There are problems that are ill-defined

those that do not have clear goals, solution paths, or expected solution.

An ill-defined problem is one that deals with complex issues and thus cannot easily be described in a concise, complete manner. Furthermore, competing factors may suggest several approaches to the problem, requiring careful analysis to determine the best approach.

Drawback of traditional techniques

Computing tasks have to be

- well-defined
- fairly predictable
- computable in reasonable time with serial computers

HENCE COMPUTATIONAL INTELLIGENCE

complex and thus not effective in real life applications. Thus the broad definition is given by: computational intelligence is a branch of computer science studying problems for which there are no effective computational algorithms. Biological organisms solve such problems every day: extracting meaning from perception, understanding language, solving ill-defined computational vision problems thanks to evolutionary adaptation of the brain to the environment, surviving in a hostile environment. However, such problems may be solved in different ways. Defining computational intelligence by the problems that the field studies there is no need to

Computational Intelligence Paradigms: Theory & Applications using MATLAB
By S. Sumathi, Surekha Paneerselvam

PROBLEM SOLVING: AI→ ADAPTATION AND COOPERATION

The ability of machines to mimic human behavior and abilities is evolving and demonstrating success in various applications.

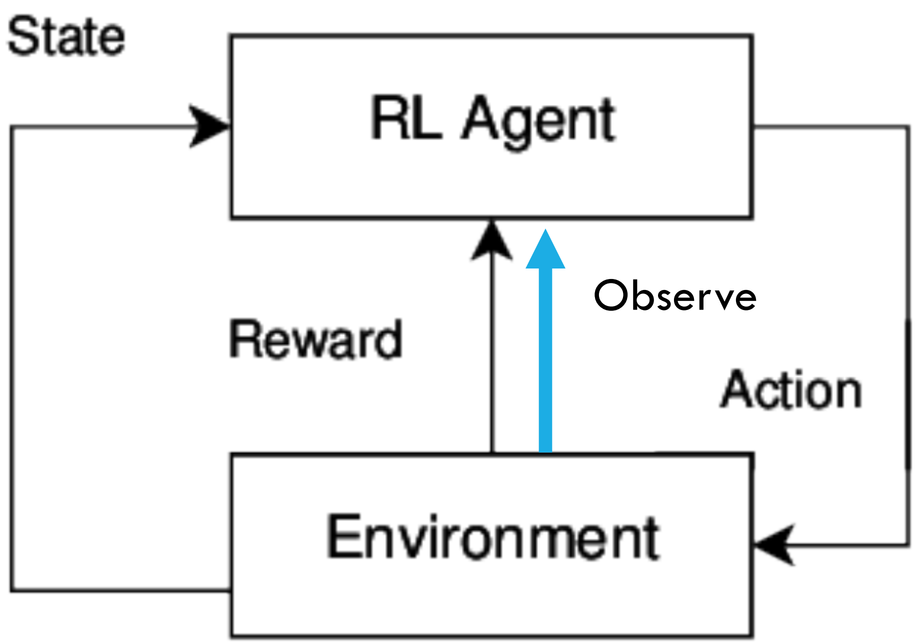
→ hence the emergence of computing domains such as AI, Machine learning, deep learning, and biologically inspired computing.

An important aspect of the human problem solving is adaptation to changes and new learning, and the ability to cooperate with others to tackle complex problems.

SOFTWARE AS AN AGENT

What elements are involved?

- The problem
- The Agent(s)
- The Environment



REAL LIFE HAS IT ALL

- Computable and well-defined
- Well-defined but hard
- ill-defined, and
- ill-defined and hard

Hence this course

LEARNING OBJECTIVES

- Motivate the need for algorithms that exhibit a degree of intelligence: logical, computational, and biologically inspired.
- Introduce the concepts of cooperation and adaptations and how they are influencing new methods for solving complex problems.
- Study meta-heuristics, evolutionary computing methods, swarm intelligence, ant-colony algorithms, particle swarm methods. reinforcement learning,
- Illustrate the use of these algorithms in solving continuous and discrete, and machine learning problems that arise in engineering applications.

COURSE DESCRIPTION

- The course addresses ill-structured problems and the need for computational intelligence methods.
- It introduces the concepts of heuristics and their use in conjunction with search methods, solving problems using heuristics and metaheuristics, constraints satisfaction.
- The course also introduces the concepts of cooperation and adaptations and how they are influencing new methods for solving complex problems.
- The course discusses how the concepts of cooperation and adaptation are manifested in nature and how such models are inspiring new types of solution methods.

COURSE TOPICS

Engineering problems are Optimization Problems: Engineering by definition is search tak where the goal is to find a solution to an optimization probelm [that is a given quantity is optimized subject to a set of constraints].

Optimization Algorithms

Exact Algorithms

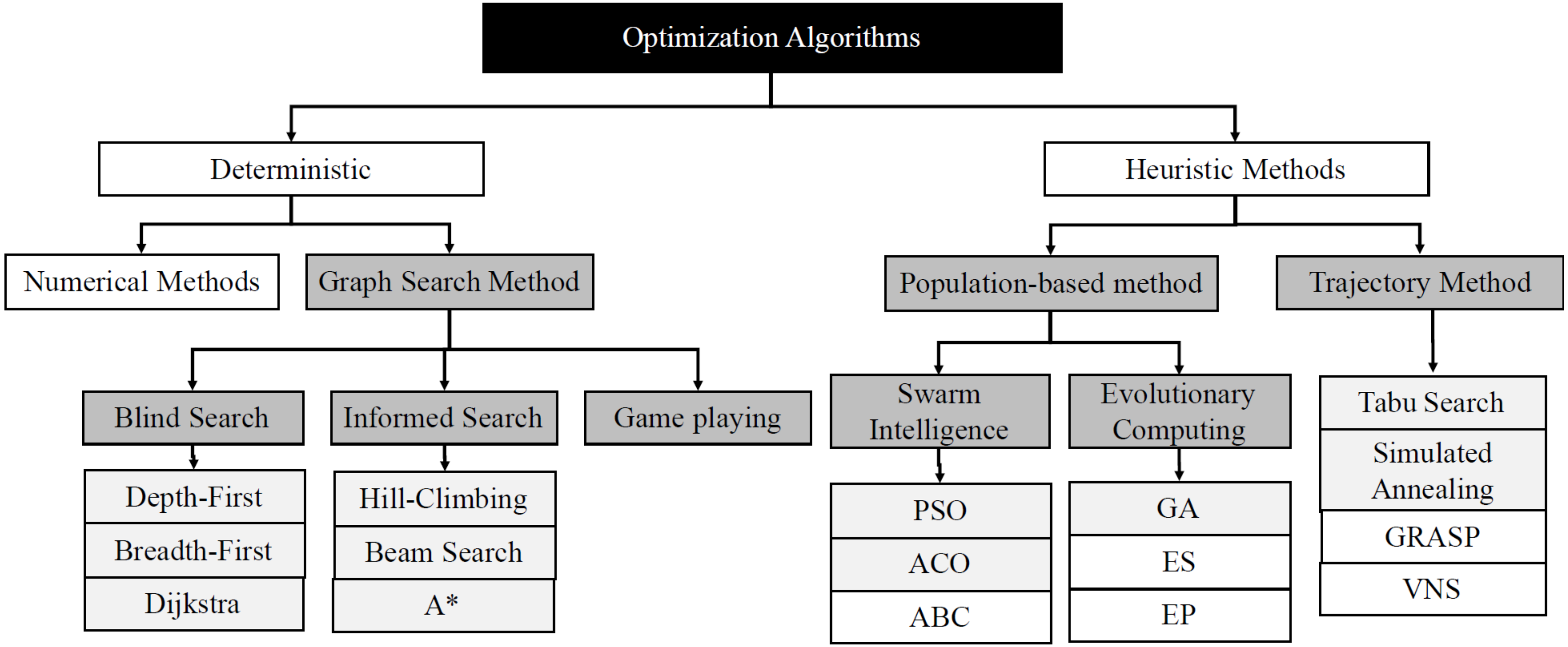
- Find the optimal solution
- High computational cost

For complex problems

Approximate Algorithms

- Find near-optimal solution
- Low computational cost

COURSE TOPICS



COURSE TOPICS

Topics to be covered include:

- *search algorithms,*
- *game playing,*
- *constraints satisfaction,*
- *meta-heuristics,*
- *evolutionary computing methods,*
- *swarm intelligence,*
- *ant-colony algorithms,*
- *particle swarm methods,*
- *Adaptive and learning algorithms*

Prereq: Level at least 4A Computer Engineering or Electrical Engineering or Software Engineering.

Antireq: CS 486, SYDE 422/522

COURSE OUTLINE

The following list is an approximate guide for the material to be covered in the course.

- Introduction: goals and definitions of artificial intelligence; intelligent agents and their environment.
- ill-structured problems and need for approximate algorithms, cooperation and adaptation in nature.
- Search: state space problem formulation and representation; uninformed search; heuristic search; iterative improvement; constraint satisfaction; game-playing.
- Review of blind search, use of heuristic search methods and meta-heuristic algorithms
- Trajectory Methods: Tabu search and Simulated Annealing.

COURSE OUTLINE- CONT.

- Genetic algorithms, cooperation in GA.
- Swarm Intelligence: cooperation and adaptation methods inspired by nature.
- Ant Colony algorithms: ACO- cooperative and multi-ant-colonies.
- Particle swarm algorithms: particle swarm optimization, cooperation within the swarms, cooperation among swarms, swarm ensembles.
- Evolutionary programming and Evolutionary strategies
- Reinforcement Learning
- Machine Learning and computational intelligence
- Engineering Applications: optimization, routing, text clustering.

REFERENCES

1. Fundamentals of Computational Swarm Intelligence, Andries P. Engelbrecht, Wiely and Sons, 2006.
2. Introduction to evolutionary computing, by Eiben and A, Smith, J.E, Springer, Berlin, 2003.
3. Search and Optimization by Metaheuristics Techniques and Algorithms Inspired by Nature

   Ke-Lin Du, M. N. S. Swam

4. Search Algorithms Types: Breadth and Depth First Search Algorithm

Merium Hazem Anwar Labib Bishara, Merihan Hazem Anwar Labib Bishara

OTHER REFERENCES IN THE LIBRARY

- Swarm Intelligence By James F. Kennedy, Russell C. Eberhart, Yuhui. Shi Published 2001 by Morgan Kaufmann
- Modern heuristic techniques for combinatorial problems, by Revees, C., Halsted Press, New York, 1993.

Course Performance Evaluation

•Assignments: There will be 4 assignment that will be used to reinforce course material. They will carry a total weight of 25% of the course's 100% final grade.
•Midterm: 25% of the course's 100% final grade.
•A final exam will be administered to test the student 's comprehensive understanding of the course material. This part will carry 50% of the course's 100% final grade.
The final course grade will be calculated using the following categories:

Assessment	Percentage of Final Grade
Assignments (4)	25%
Midterm	25%
Final Examination	50%

WE ARE HERE TO ASSIST

- Weekly tutorials will be set up.
- The tutorials will be used to reinforce course material and to help you with the assignments.
- Once you attempt the assignment and determine that you need help with it, please send your questions to the teaching assistants before the tutorial so they can cover your enquiry as a topic
- Alternatively, you can send your questions re the assignment, or any course topic, in general, to the teaching assistants
- We will set up a discussion group on LERAN to facilitate interaction among you, the teaching assistants, and I.