

Project 3: Informed and Local Search Presentation

LINFO1361 – Intelligence Artificielle

Amaury Fierens

Summary

- Informed Search: N-Queens
- Local Search: Sudoku Solver
- Grading details

Summary

- **Informed Search: N- Amazons**
- Local Search: Sudoku Solver
- Grading details

Informed Search: A*

What is A*?

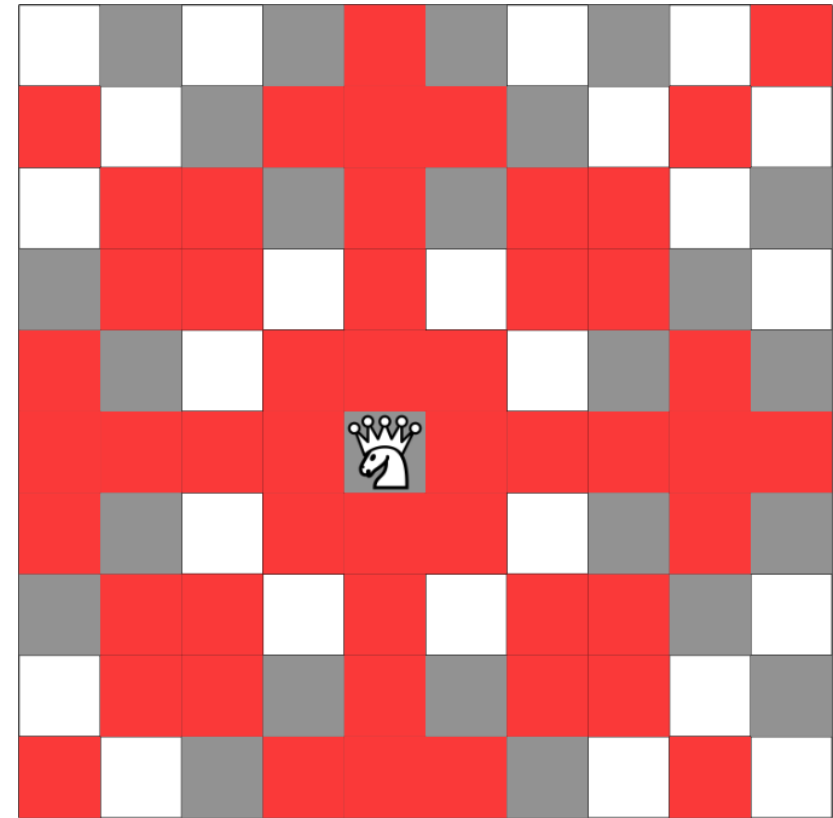
A **best first search** where the idea is to use an **optimistic heuristic** to guide the search (guarantees the convergence). **Optimistic** means the heuristic **underestimates** the path that remains to be done.

See S3 slides for examples.

N- Amazons: Description

What is an Amazon?

Custom piece that possess the move of a **Queen** and of an **extended Knight** (4 move in one direction then 1 in another or 3 in one direction then 2 in another).



Example of the tiles that an Amazon threatens on a 10x10 board

N- Amazons: Rules

Goal of the problem:

Places **N amazon pieces** on the **NxN board**, avoiding that any of the amazon attacks another one. The goal is reached when the **rightmost column** (or **downmost row** depending of the orientation of your board) of the board is filled with an amazon.

State of the problem:

A **state** of the problem is represented by the board with **one more** amazon piece than the previous state.

N- Amazons: Implementation

Input format:

An **integer** is given as input, which is the **N** of N-Amazons problem. It means that if the input is 10, your solver must place 10 amazons in a 10x10 chess board without any of these attacking each other.

[illegible]

Summary

- Informed Search: N- Amazons
- **Local Search: Sudoku Solver**
- Grading details

Sudoku: Description

What is Sudoku?

Sudoku is a game where a 9x9 grid with 9 3x3 sub-grids is partially filled with digits from 1 to 9.

The **goal** of Sudoku is to **fill the empty cells** with digits from 1 to 9 by following **three rules**.

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

Sudoku grid example

Sudoku: Rules

The four rules of Sudoku:

1. Each digit in the initial grid cannot be changed in any way
2. Each column must contain all digits from 1 to 9 exactly once
3. Each row must contain all digits from 1 to 9 exactly once
4. Each sub-grid must contain all digits from 1 to 9 exactly once

Sudoku: Implementation

Objective score:

Find the **best fitting objective score** (an example is number of conflicts with the rules) to determine if the given grid of sudoku is **close to resolution**.

Possible actions:

Find all the actions that could be explored for next step of the Local Search /
Generate the next action to be performed.

Simulated annealing:

Find the **best cooling rate** to **achieve a solution** for all the given instances while **spending less time** than the maximum limit.

Sudoku: Implementation

Input format:

An **input instance** is a **9x9 grid of integers in a .txt file**. Each tile containing digits from 1 to 9 is meant to stay as it is. Each tile containing 0 is meant to be changed by your algorithm.

```
1 745090000
2 032150046
3 000280503
4 200000060
5 980600351
6 000540207
7 308000002
8 020760010
9 060908034
```

Instance file example

Sudoku: Implementation

Output format:

The **output** of your program should be the **9x9 grid of your best solution**, followed by a **blank line** and then a **text line**:
« Value(C): ... » with ... being the value of your objective score.

```
745396128
832157946
196284573
257831469
984672351
613549287
378415692
429763815
561928734

Value(C): 0
```

Output example

Summary

- Informed Search: N-Queens
- Local Search: Sudoku Solver
- **Grading details**

Grading details

Report: 12/20

- Search Algorithms in general: 3/20
- N-Amazons problem: 6/20
- Sudoku problem: 2/20

Inginious: 8/20

- N-Amazons problem: 2/20
- Sudoku problem: 6/20