1 Search

Total cost = Time to find a solution (off-line cost) + Cost of the solution path (on-line cost).
Strategy evaluation criteria:

- **Completeness**: does the strategy guarantees to find a solution if one exists?
- **Time complexity**: how long does it take to find a solution?
- **Space complexity**: how much memory is needed to carry out the search?
- **Optimality**: does the strategy find the best solution when there are more solutions?

2 Games

TODO

3 Constraint Programming

TODO

3.1 Propagation algorithms

- **Standard Backtracking**: Assign without checking future constraints, then check validity
- **Forward Checking**: After each assignment propagate the constraints from the assigned value to the free values
- **Partial Look Ahead**: After Forward Checking check constraints between free values in one direction
- **Full Look Ahead**: After Forward Checking check constraints between free values in both directions

4 Algorithms

b branching factor
d solution depth
m maximum depth of the search tree
l depth limit

Name	Complete	Optimal	Time	Space	Notes
Non-informed search strategies					
Breadth-First	Yes	Yes	b^d	b^d	
Uniform-Cost	Yes	Yes	b^d	b^d	
Depth-First	No	No	b^m	bm	
Depth-First, limited depth	If $l \geq d$	No	b^l	bl	
Iterative Deepening	Yes	Yes	b^d	bd	
Informed search strategies					
Best-First	No	No	b^d	b^d	Breadth-first with cost of passed steps
A*	Yes	Optimistic heuristic			Breadth-first with cost of passed and future steps
Local search					
TODO	TODO	TODO	TODO	TODO	
Swarm intelligence					
Ant Colony Optimization					Based on ants' behaviour, positive feedback based on pheromone trails
Artificial Bee Colony					Individuals with different functions
Particle Swarm Optimization					Based on the observation of bird flocks or fish shouls. Stigmergy is used as communication.
Games					
Min-Max	Yes	Yes	b^m	bm	
Min-Max, α/β pruning	Yes	Yes	$\leq b^m$, $\geq b^{\frac{m}{2}}$		
Constraint programming					
TODO	TODO	TODO	TODO	TODO	