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 Swarm intelligence

TODO

3 Games

TODO

4 Constraint Programming

TODO

4.1 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

5 Algorithms

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

Name	Complete?	Optimal?	Time complexity	Space complexity	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	
A*	Yes	If heuristic is optimistic			
Local search					
TODO	TODO	TODO	TODO	TODO	
Swarm intelligence					
TODO	TODO	TODO	TODO	TODO	
Games					
Min-Max	Yes	Yes	b^m	bm	
Min-Max, α/β pruning	Yes	Yes	Depends, $\sim b^{\frac{3}{4}d}$	TODO	
Constraint programming					
TODO	TODO	TODO	TODO	TODO	