Heuristic Analysis

In this document I am going to analyze three different planning problems and how each search strategy deal with it in terms of performance and optimal result. The search strategies are divided in two categories, Uninformed and Heuristic based.

Uninformed Search: also known as "blind search", are those methods where the strategy has no additional information about the states beyond what is provided in problem description itself. In its basic form there will be non-goal and goal states. These search methods will try to reach a goal state transversing the branches by using different node expansion techniques.

Heuristic Search: the main difference is that a heuristics are used, adding problem specific knowledge, that will help to find solutions more efficiently. Nodes provide more information to the search algorithm that will help to reach the goal. States are not just empty data nodes anymore.

Problem 1

Description of the problem:

- **Planes**, P1 and P2.
- **Cargos**, C1 and C2.
- **Airports**, JFK and SFO.
- **P1 and C1** are at **SFO**.
- **P2 and C2** are at **JFK**.
- GOAL: **C1 at JFK** and **C2 at SFO**.

Optimal Plan:

1. Load(C2, P2, JFK)
2. Load(C1, P1, SFO)
3. Fly(P2, JFK, SFO)
4. Fly(P1, SFO, JFK)
5. Unload(C2, P2, SFO)
6. Unload(C1, P1, JFK)

Uninformed Search Strategies

Search result data:

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Plan
Breadth First Search	**43**	**56**	**0.04**	**YES**
Breadth First Tree Search	1458	1459	1.29	**YES**
Depth First Graph Search	12	13	0.008	NO
Depth Limited Search	101	271	0.10	NO
Uniform Cost Search	55	57	**0.04**	**YES**
Recursive Best First Search	4229	4230	3.25	**YES**

From the three search strategies that found the optimal plan, **Breadth First Search** is the one that performs better. It has to do less expansions, goal tests and with lower time needed to find it.

Heuristic Search Strategies

Search result data:

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Plan
Greedy Best First Graph Search	**7**	**9**	**0.008**	**YES**
A* Search	55	57	0.04	**YES**
A* with h_ignore_preconditions	41	43	0.04	**YES**
A* with h_pg_levelsum	11	13	0.73	**YES**

Greedy Best First Graph Search is the clear winner, both in efficiency and optimal plan. The difference with the other 3 search methods is noticeable.

Comparing Search Strategies

If we compare Heuristics with the Uninformed Search Strategies, **Greedy Best First Graph Search** performs better and faster. For small sized problems, it is the fastest and more efficient solution.

Problem 2

Description of the problem:

- **Planes**, P1, P2 and P3.

- **Cargos**, C1, C2 and C3.
- **Airports**, JFK, SFO and ATL.
- **P1 and C1** are at **SFO**.
- **P2 and C2** are at **JFK**.
- **P3 and C3** are at **ATL**.
- GOAL: **C1 at JFK**, **C2 at SFO** and **C3 at SFO**.

Optimal Plan:

1. Load(C1, P1, SFO)
2. Load(C2, P2, JFK)
3. Load(C3, P3, ATL)
4. Fly(P1, SFO, JFK)
5. Fly(P2, JFK, SFO)
6. Fly(P3, ATL, SFO)
7. Unload(C1, P1, JFK)
8. Unload(C2, P2, SFO)
9. Unload(C3, P3, SFO)

Uninformed Search Strategies

Search result data:

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Solution
Breadth First Search	**3315**	**4572**	15.47	**YES**
Breadth First Tree Search	--	--	--	--
Depth First Graph Search	1609	1610	7.13	NO
Depth Limited Search	--	--	--	--
Uniform Cost Search	4811	4813	**14.62**	**YES**
Recursive Best First Search	--	--	--	--

Those entries without data needed more than 10 minutes to perform the search so I canceled the process without waiting for them to finish. From the ones that could finish in less than 10 minutes, two provided the optimal solution. **Breadth First Search** is again a clear winner, with less expansion and less goal tests. That means less memory and CPU process needed while being only 0,85 seconds slower than Uniform Cost Search.

Heuristic Search Strategies

Search result data:

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Plan
Greedy Best First Graph Search	998	1000	**2.95**	NO
A* Search	4811	4813	14.54	**YES**
A* with h_ignore_preconditions	1449	1451	5.36	**YES**
A* with h_pg_levelsum	**86**	**88**	128.69	**YES**

In this case **A* with h_pg_levelsum** needs more time than the others but it is more efficient, since it finds an optimal results with less expansions and goal tests. So, the heuristic makes the search more efficient but the overall time needed is greater. Greedy Best First Graph Search is still the fastest, but this time the resulting plan is not optimal.

Comparing Search Strategies

Heuristics are faster and more efficient, with the exception of **A* with h_pg_levelsum** that it is actually slower (but still more efficient).

Problem 3

Description of the problem:

- **Planes**, P1 and P2.
- **Cargos**, C1, C2, C3 and C4.
- **Airports**, JFK, SFO, ATL and ORD.
- **P1 and C1** are at **SFO**.
- **P2 and C2** are at **JFK**.
- **C3** is at **ATL**.
- **C4** is at **ORD**.
- GOAL: **C1 at JFK**, **C2 at SFO**, **C3 at JFK** and **C4 at SFO**.

Optimal Plan:

1. Load(C2, P2, JFK)
2. Load(C1, P1, SFO)
3. Fly(P2, JFK, ORD)
4. Fly(P1, SFO, ATL)
5. Load(C4, P2, ORD)
6. Load(C3, P1, ATL)
7. Fly(P1, ATL, JFK)

8. Unload(C1, P1, JFK)
9. Unload(C3, P1, JFK)
10. Fly(P2, ORD, SFO)
11. Unload(C2, P2, SFO)
12. Unload(C4, P2, SFO)

Uninformed Search Strategies

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Solution
Breadth First Search	**14663**	18098	143.13	**YES**
Breadth First Tree Search	--	--	--	--
Depth First Graph Search	592	593	3.65	NO
Depth Limited Search	--	--	--	--
Uniform Cost Search	17797	**17799**	**66.66**	**YES**
Recursive Best First Search	--	--	--	--

The same three search strategies that in previous problem couldn't find a solution in less than 10 minutes, failed here again. From the two that created an optimal solution, **Uniform Cost Search** performs faster and with less goal tests. Still the number of expansions is greater, so the memory needed to solve the problem is larger than in Breadth First Search. Since this problem is a bit more complex that the previous one, the difference on "Time Elapsed" increased significantly, making clear that for more complex problems Uniform Cost Search can provide results faster. The difference increased to 114%, while the expansion only increased by 21%.

Heuristic Search Strategies

Search result data:

Search Performed	Expansions	Goal Tests	Time Elapsed	Optimal Plan
Greedy Best First Graph Search	**4273**	**4275**	**15.65**	NO
A* Search	17797	17799	65.37	**YES**
A* with h_ignore_preconditions	5034	5036	21.28	**YES**
A* with h_pg_levelsum	--	--	--	--

A* with h_pg_levelsum needed more than 10 minutes, so the data is not included in this table. Greedy Best First Graph Search again showed a non optimal plan, so **A* with h_ignore_preconditions** is the winner this time, since it is faster and more efficient than the regular A*.

Comparing Search Strategies

Again, Heuristics are more efficient and produce a plan faster than uninformed strategies.

Conclusion

In all the problems, Heuristic Search Strategies are faster and more efficient than the Uninformed ones.