Heuristic analysis report on planning search

By Asarudheen

For

Udacity Artificial Intelligence nano degree

Introduction

In this project, we have used planning graph and automatic domain-independent heuristics with A* search to solve deterministic logistics planning problems for an Air Cargo transport system. In this analysis, we compare our results with the uninformed heuristics search methods.

Action schema for Air Cargo problem:

```
Action(Load(c, p, a),
        PRECOND: At(c, a) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
        EFFECT: ¬ At(c, a) ∧ In(c, p))
Action(Unload(c, p, a),
        PRECOND: In(c, p) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
        EFFECT: At(c, a) ∧ ¬ In(c, p))
Action(Fly(p, from, to),
        PRECOND: At(p, from) ∧ Plane(p) ∧ Airport(from) ∧ Airport(to)
        EFFECT: ¬ At(p, from) ∧ At(p, to))
```

Problem 1 initial state and goal:

```
Init(At(C1, SFO) ∧ At(C2, JFK)
        ∧ At(P1, SFO) ∧ At(P2, JFK)
        ∧ Cargo(C1) ∧ Cargo(C2)
        ∧ Plane(P1) ∧ Plane(P2)
        ∧ Airport(JFK) ∧ Airport(SFO))
Goal(At(C1, JFK) ∧ At(C2, SFO))
```

Problem 2 initial state and goal:

```
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL)
        ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ At(P3, ATL)
        ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3)
        ∧ Plane(P1) ∧ Plane(P2) ∧ Plane(P3)
        ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL))
Goal(At(C1, JFK) ∧ At(C2, SFO) ∧ At(C3, SFO))
```

Problem 3 initial state and goal:

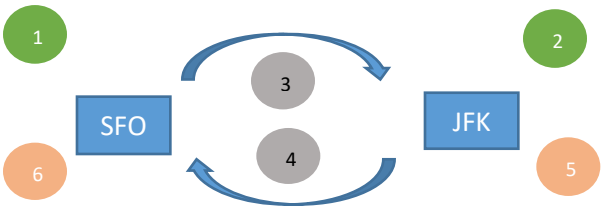
```
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(C4, ORD)
        ∧ At(P1, SFO) ∧ At(P2, JFK)
        ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Cargo(C4)
        ∧ Plane(P1) ∧ Plane(P2)
        ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL) ∧ Airport(ORD))
Goal(At(C1, JFK) ∧ At(C3, JFK) ∧ At(C2, SFO) ∧ At(C4, SFO))
```

OPTIMAL SOLUTION FOR PROBLEM 1, PROBLEM 2 AND PROBLEM 3

Problem 1 Solution :

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

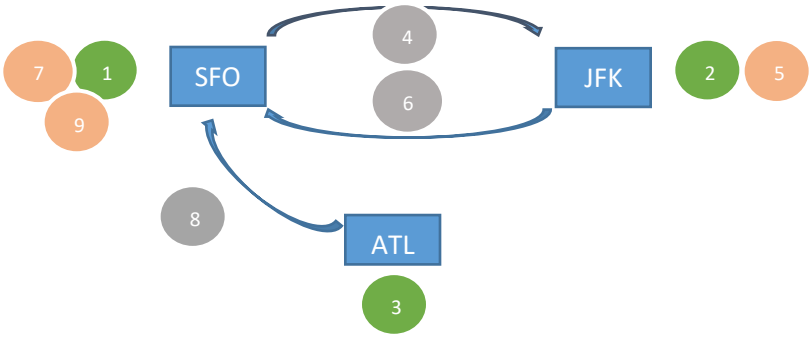
Optimal plan length: 6



Problem 2 Solution :

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

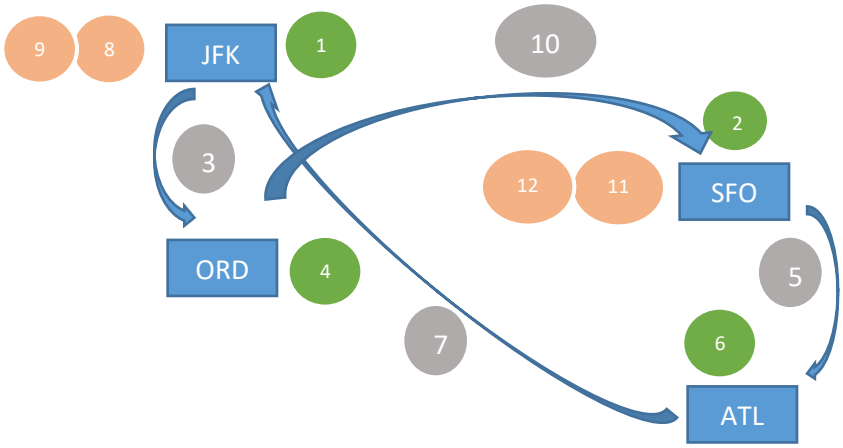
Optimal plan length: 9



Problem 3 Solution:

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

Optimal plan length: 12



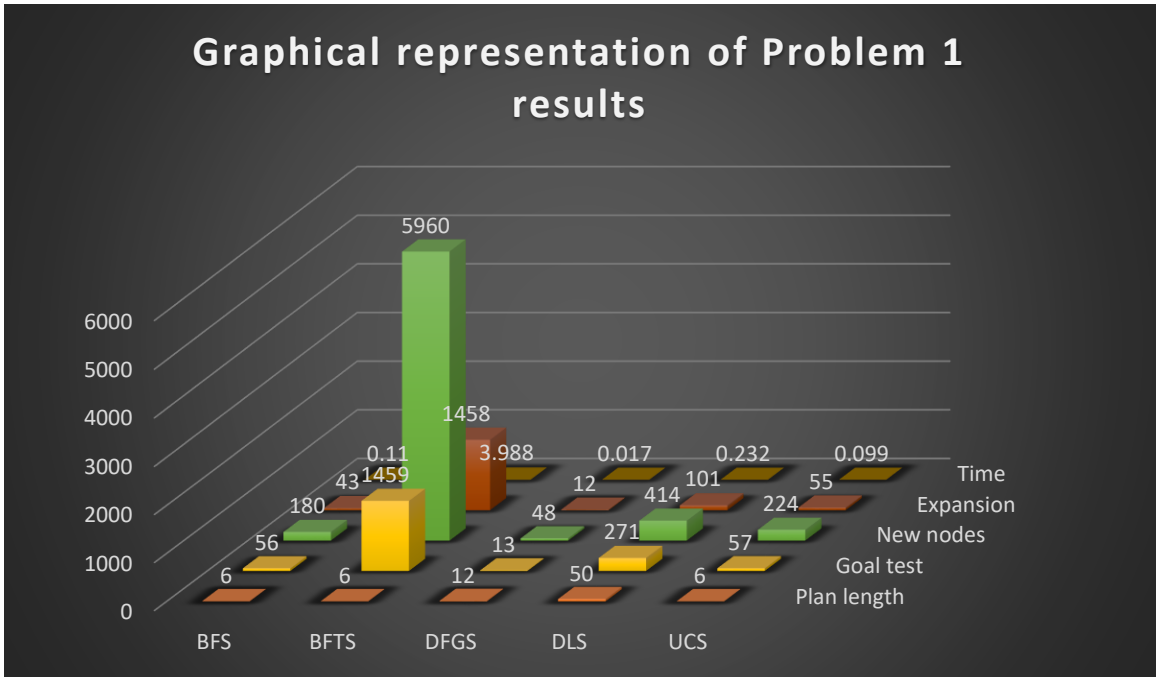
****

Compare and contrast non-heuristic search result metrics

Problem 1:

Results obtained for problem one are below

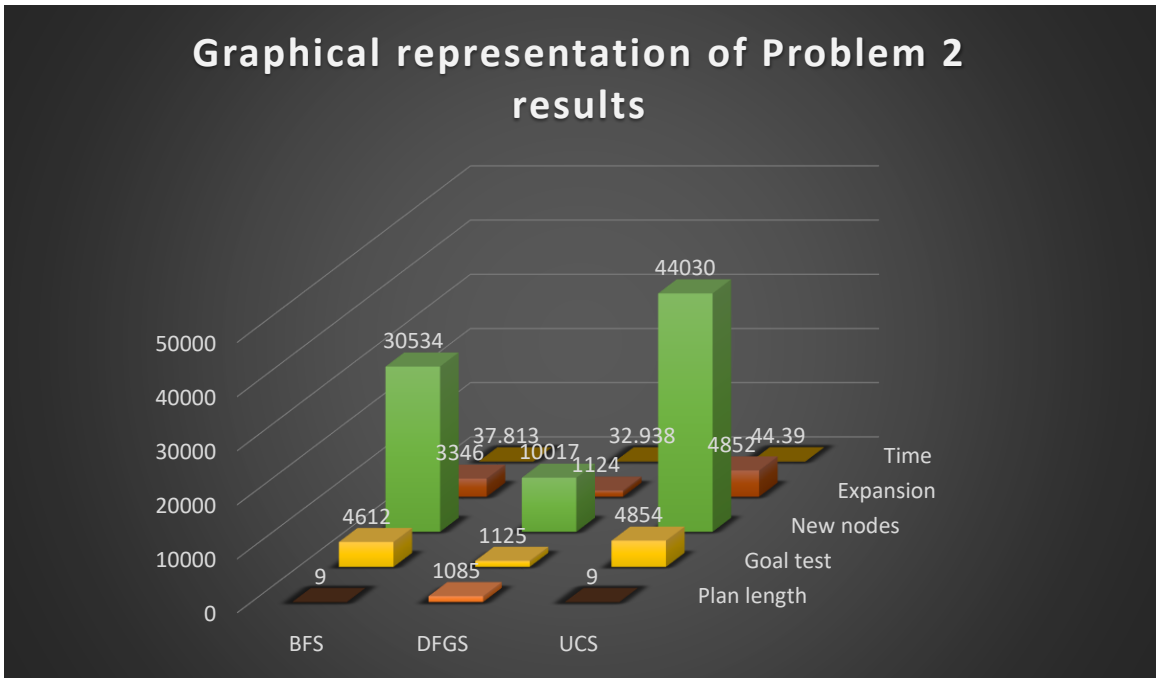
ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
BFS	**6**	**43**	**56**	**180**	**0.11**
BFTS	**6**	**1458**	**1459**	**5960**	**3.988**
DFGS	12	12	13	48	0.017
DLS	50	101	271	414	0.232
UCS	**6**	**55**	**57**	**224**	**0.099**



Optional plan length for problem 1 is 6. BFS, BFTS, and UCS obtained the optimal plan length for this problem.

Problem 2:

ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
BFS	**9**	**3346**	**4612**	**30534**	**37.8143**
BFTS	-	-	-	-	-
DFGS	1085	1124	1125	10017	32.938
DLS	-	-	-	-	-
UCS	**9**	**4852**	**4854**	**44030**	**44.39**

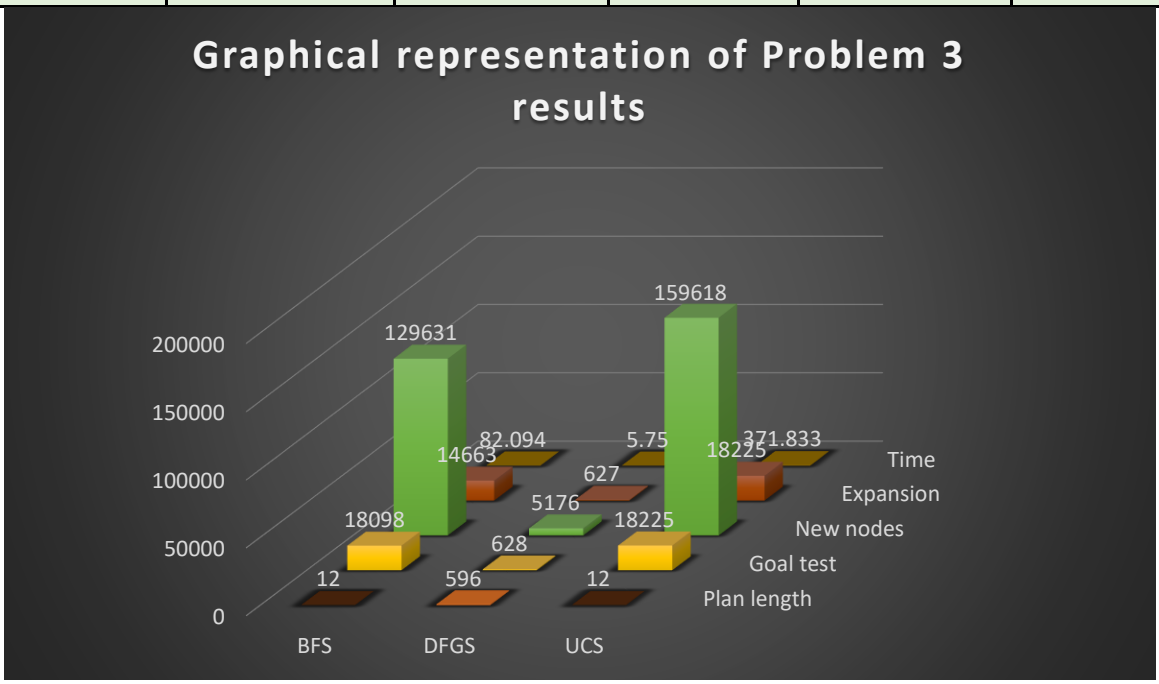


Note: BFTS and DLS took more than 10 minutes to calculate the solution.

Optional plan length for problem 2 is 9. BFS and UCS obtained the optimal plan length for this problem.

Problem 3:

ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
BFS	**12**	**14663**	**18098**	**129631**	**82.094**
BFTS	-	-	-	-	-
DFGS	596	627	628	5176	5.75
DLS	-	-	-	-	-
UCS	**12**	**18225**	**18225**	**159618**	**371.833**



Note: BFTS and DLS took more than 10 minutes to calculate the solution.

Optional plan length for problem 3 is 12. BFS and UCS obtained the optimal plan length for this problem.

Analysis on results obtained by BFS, DFS and UCS for P1, P2 and P3:

Observation on plan length: In the above 3 problems only BFS and UCS are the only two uninformed search which found the optimal plan.

Observation on plan time elapsed: In DFGS algorithm we found the solution path is less than for all the three problems but the plan length is not optimal.

Observation on number of node expansion: In DFGS algorithm expansion is less when compared to other algorithms. For UCS and BFS expansion is high but optimal solution is obtained.

Conclusion: Now the competition is between UCS and BFS as they both found the optimal solution. When taking memory and time into consideration BFS is optimal.

Justification of Observed results:

From Reference 2 it's clearly mentioned that BFS is complete and optimal. Its only drawback is memory usage, if the branching factor is high for the given problem.

From Reference 1 its clear now why DFS is not an optimal solution for this kind of problem which is in sync with the results obtained for above problems.

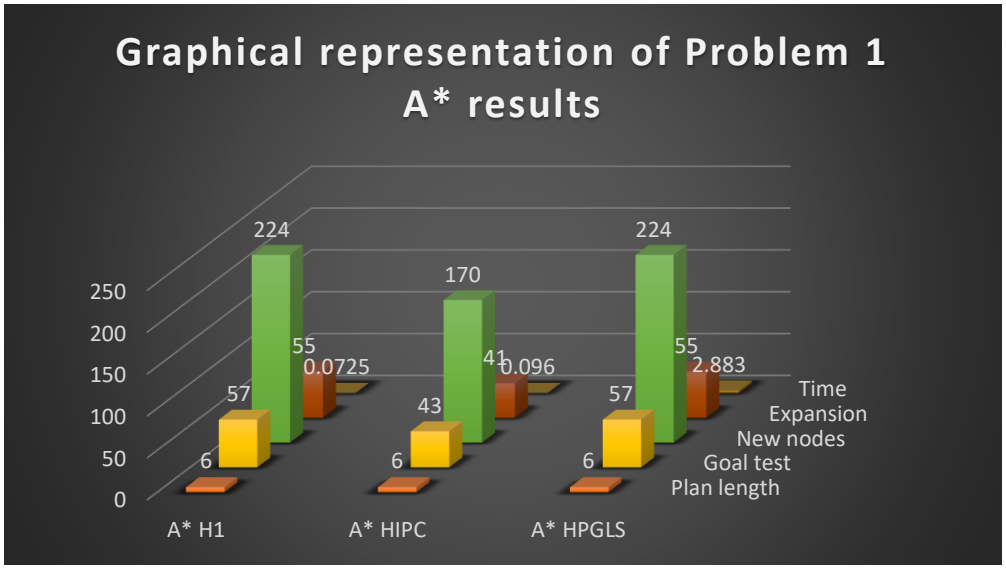
****

Compare and contrast heuristic search result metrics

Problem 1:

Results obtained for problem one are below

ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
A* H1	6	55	57	224	0.0725
A* HIPC	6	41	43	170	0.096
A* HPGLS	6	55	57	224	2.883

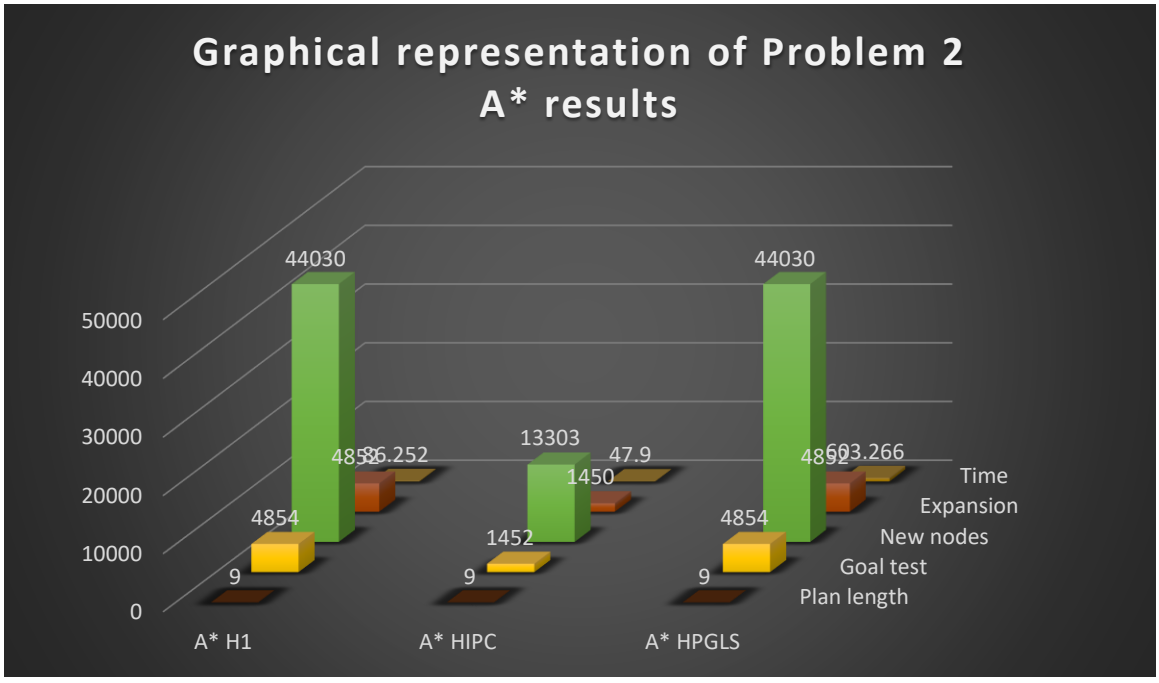


Optional plan length for problem 1 is 6. A*H1, A* HIPC, and A*HPGLS obtained the optimal plan length for this problem.

Problem 2:

Results obtained for problem one are below

ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
A* H1	9	4852	4854	44030	86.252
A* HIPC	9	1450	1452	13303	47.9
A* HPGLS	9	4852	4854	44030	603.266

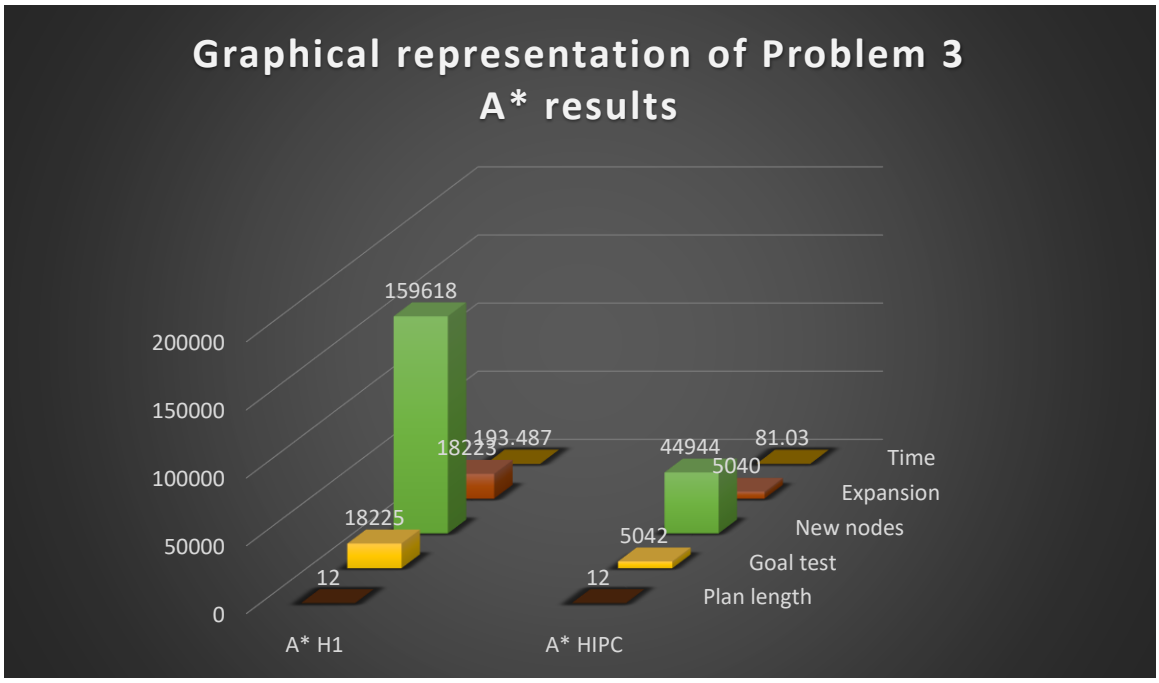


Optional plan length for problem 2 is 9. A*H1, A* HIPC, and A*HPGLS obtained the optimal plan length for this problem.

Problem 3:

Results obtained for problem one are below

ALGORITHM	PLAN LENGHT	EXPANSIONS	GOAL TEST	NEW NODES	TIME IN SEC
A* H1	**12**	**18223**	**18225**	**159618**	**193.487**
A* HIPC	**12**	**5040**	**5042**	**44944**	**81.03**
A* HPGLS	-	-	-	-	-



Optional plan length for problem 3 is 12. A*H1 and A* HIPC obtained the optimal plan length for this problem.

Note: A* HPGLS took more than 10 minutes to calculate the solution.

****

Analysis of results obtained by A* HIPC and A*HPGLS for P1, P2 and P3:

Observation on plan length: In the above 3 problems A* HIPC and A*HPGLS (not for problem 3) search which found the optimal plan.

Observation on plan time elapsed: In A* HIPC we found the solution path is less time than other searches in all the three problems and the plan length is also optimal.

Observation on number of node expansion: In A* HIPC we found the solution path by having less node expansion when compared to others searches and plan length obtained also optimal for all the problems.

Conclusion: Now the competition is between A*HPGLS and A* HIPC as they both found the optimal solution for problem 1 and problem 2. A*HPGLS failed to find the optimal path in 10 minutes but A*HIPC did. When taking memory and time into consideration A* HIPC has less time and memory. So in heuristic search, A* HIPC is optimal.

Justification of Observed results:

All the heuristics optioned the optimal result but A* HIPC outperformed. The reason A*HPGLS taking a long time for problem 3 is explained in Reference 3. It explains how the sum of all actions count which results in bad heuristic and leads to poor performance.

BEST HEURISTIC:

From the above results, it's clearly proven that A*HIPC uses less memory and faster when compared with other heuristics and obtained optimal results. Hence A*HIPC would be the best choice for this Air Cargo problem.

****

Is A*HIPC better than non-heuristic search planning methods?

A*HIPC vs BFS:

BFS and A*HIPC found the optimal path but A*HIPC found the optimal solution in less time and A*HIPC require less expansion when compared to BFS.

A*HIPC vs UCS:

UCS and A*HIPC found the optimal path but A*HIPC found the optimal solution in 50% less time and A*HIPC require less expansion when compared to UCS.

From the above comparison is clearly proven that A*HIPC is better than non-heuristic searches.

****

Abbreviations

BFS – Breadth-first search

BTFS– Breadth tree first search

DFGS – Depth first graph search

DLS– Depth Limited search

UCS – Uniformed cost search

A*HIPC – A*Search heuristic ignoring precondition

A*HPGLS – A* search Heuristic planning graph level sum

****

References:

Reference 1: AIND VIDEO LESSON 10 (SEARCH COMPARISON)

Reference 2: AIMA EDITION 3 (3.4.7 Comparing uninformed strategies)

Reference 3: AIMA EDITION 3 (3.6.1 The effect of heuristic accuracy on performance)

****

Thankyou ☺