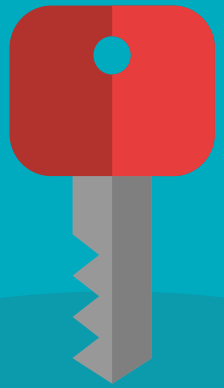
Rush Hour

By Ebner, Lamplmair, Mayer

01

A* Algorithm



A* implementation Rush Hour

- Finding the best possible solution to reach the goal
- Minimize f(n) = g(n) + h(n)
- Use Open and Closed List
- Open List: Java Priority Queue
- Closed List: Set
- Costs for each move is 1 (=depth)

A* implementation Rush Hour

- Checking if current state = Goal
  - Yes => End
- Else add Node to closed list / remove from open list
- Expand node
  - Get the cost of the expanded nodes
    - Node on close list -> ignore Node
    - Node on open list
      - Check if costs are better → add better node on open list

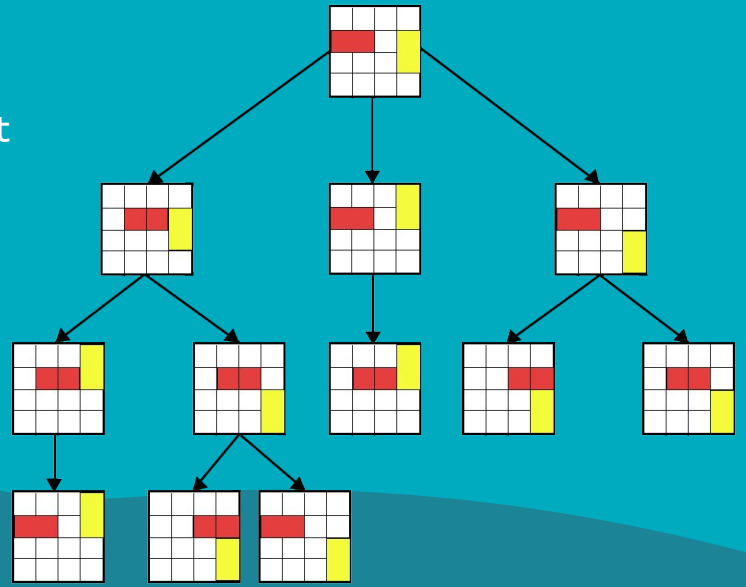
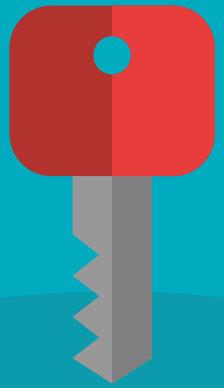


Image source: Jelle_van_Dijk_BSc_Thesis_08_06_2018

02

Heuristics



Our Heuristics

Zero

Returns 0 for each node. Equals Breadth-First Search

Blocking

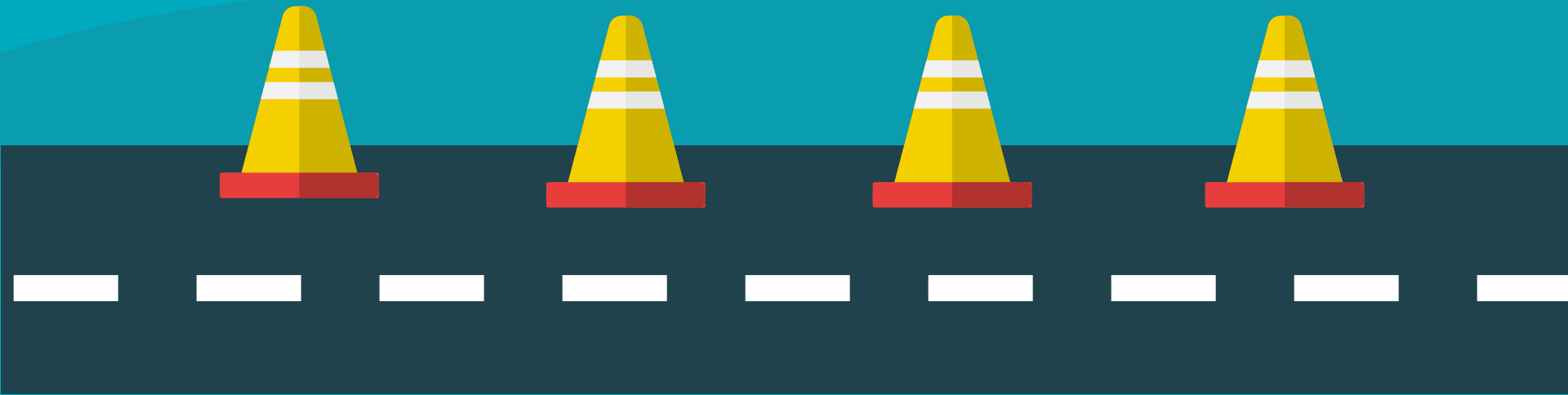
Returns number of cars blocking the goal car

Distance (!)

Returns distance of goal car to exit

Advanced

Returns number of all cars blocking the blocking cars



Admissible and Consistent

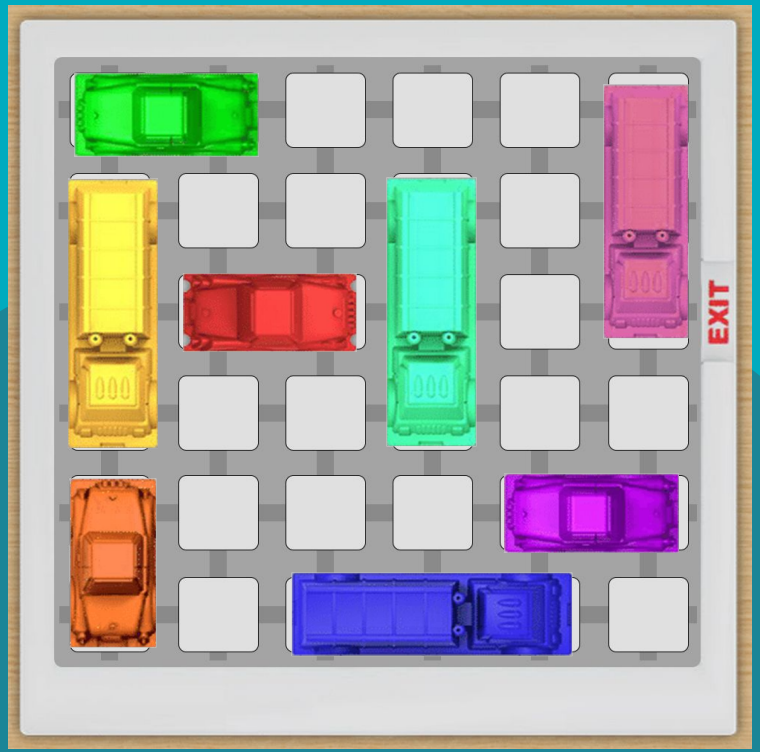
"An admissible heuristic never overestimates the cost to reach the goal."

ADMISSIBLE

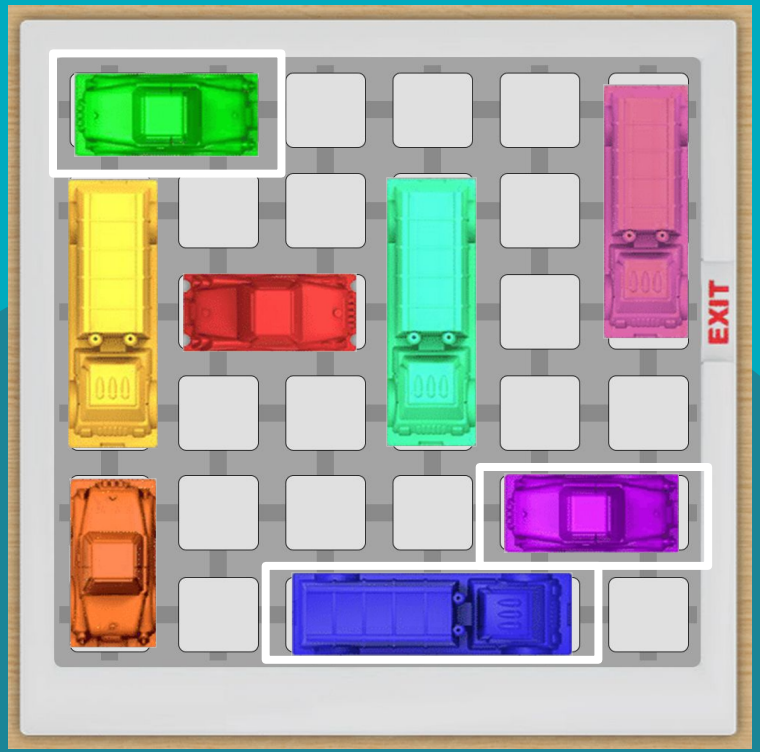
"A consistent heuristic never overestimates the distance from any neighboring node to the goal, plus the cost of reaching that neighbor."

CONSISTENT

Blocking Heuristic



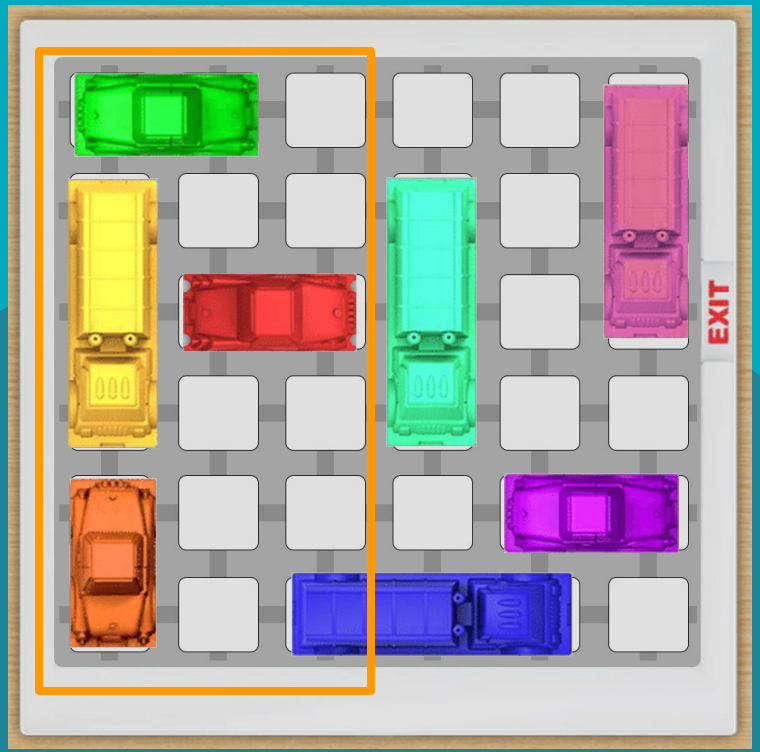
Blocking Heuristic



Cars don’t block if..

- they have same orientation

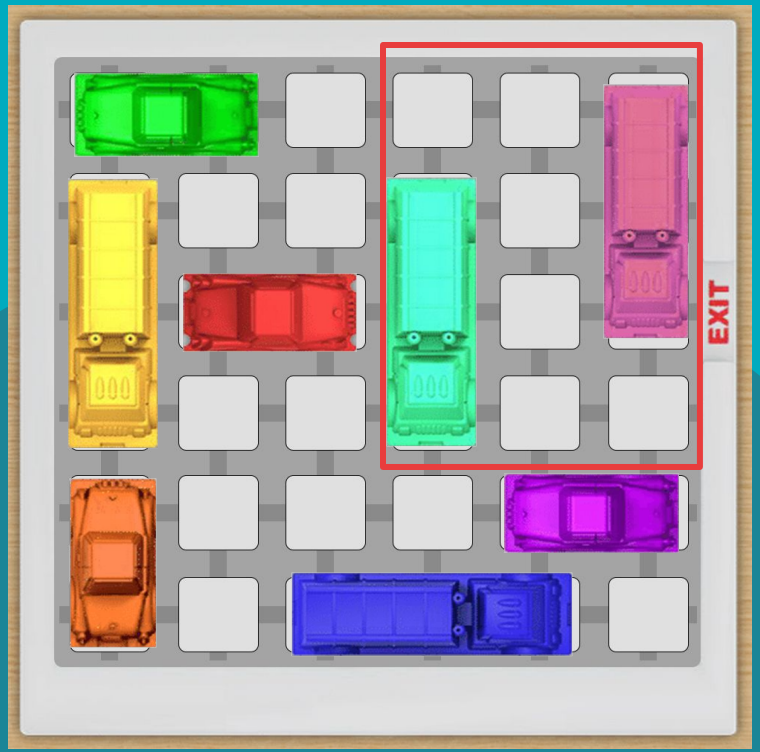
Blocking Heuristic



Cars don't block if..

- they have same orientation
- **Their X-position** <= end of goal car

Blocking Heuristic



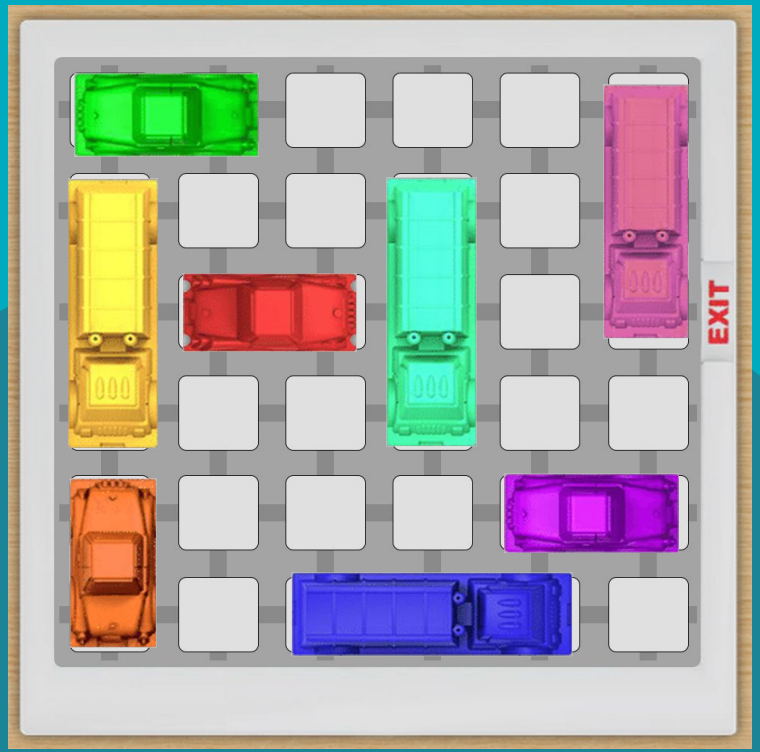
Cars are **blocking** if..

- Their X-position > x of goal car
- Their Y-position + size > y of goal car

AND

Their Y-position <= y of goal car

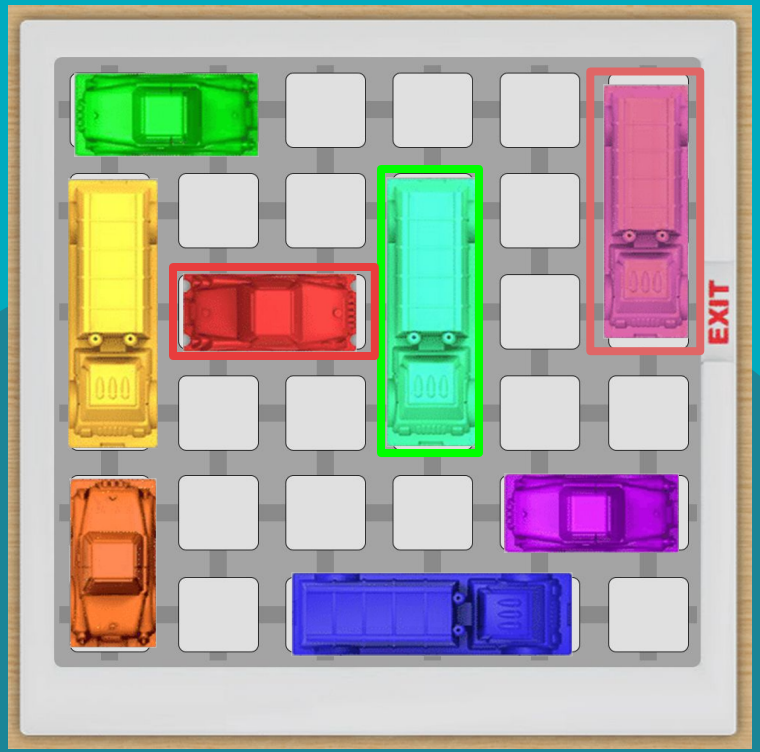
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

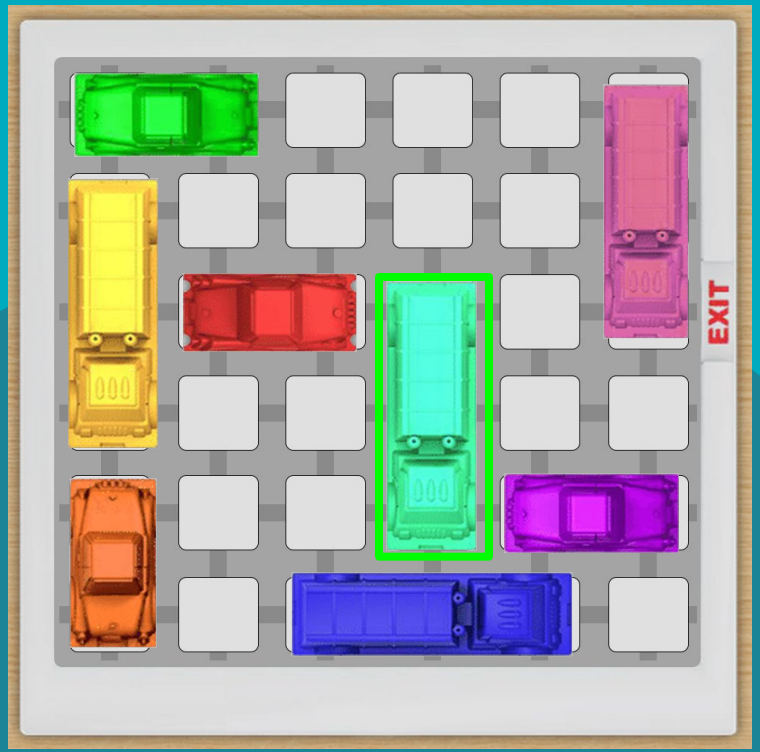
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if nearest car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

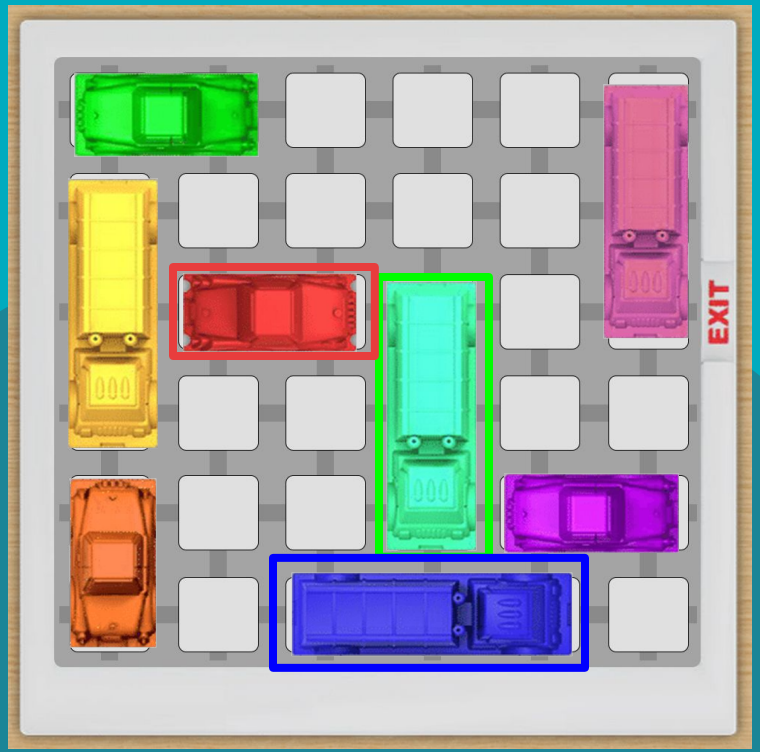
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

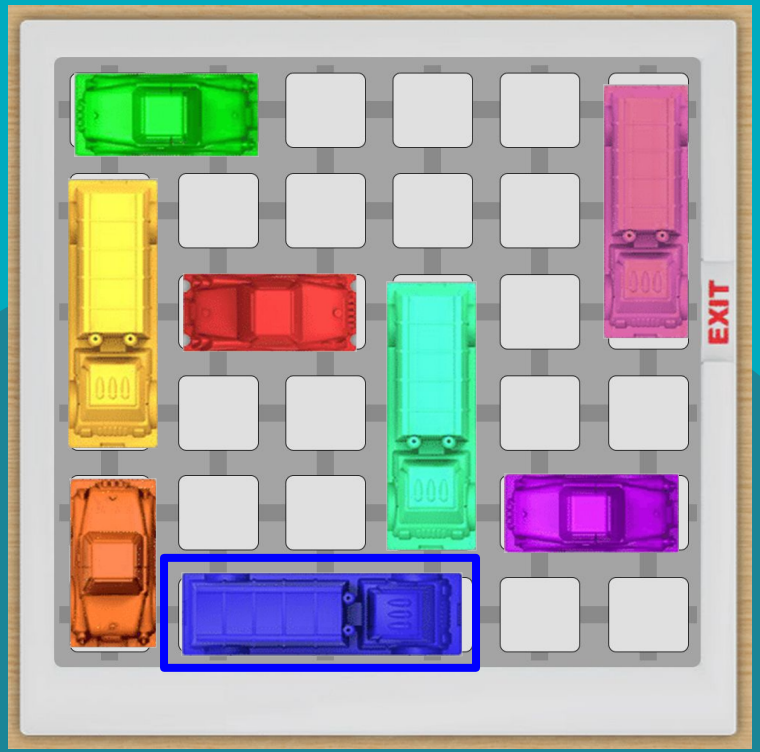
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

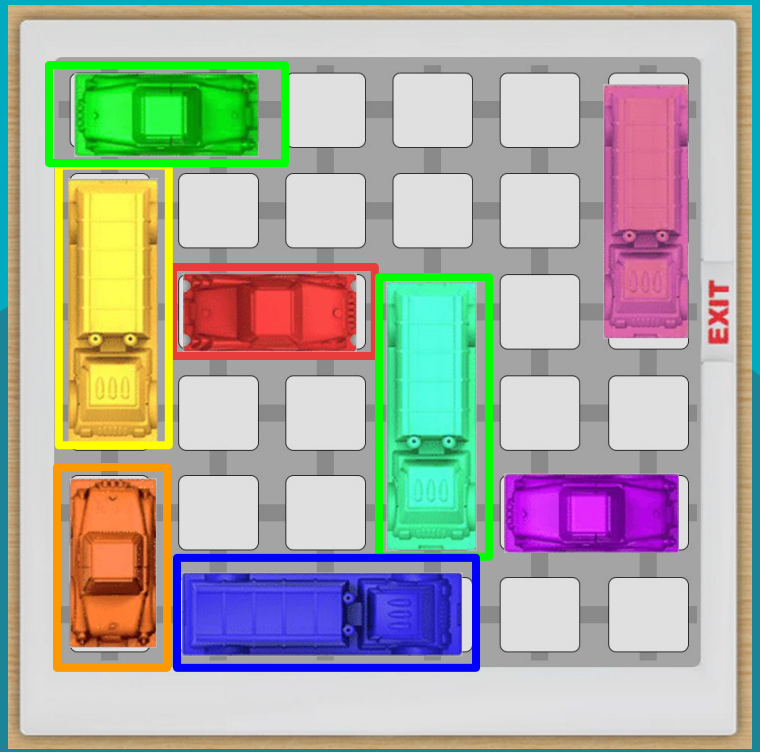
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

Advanced Heuristic

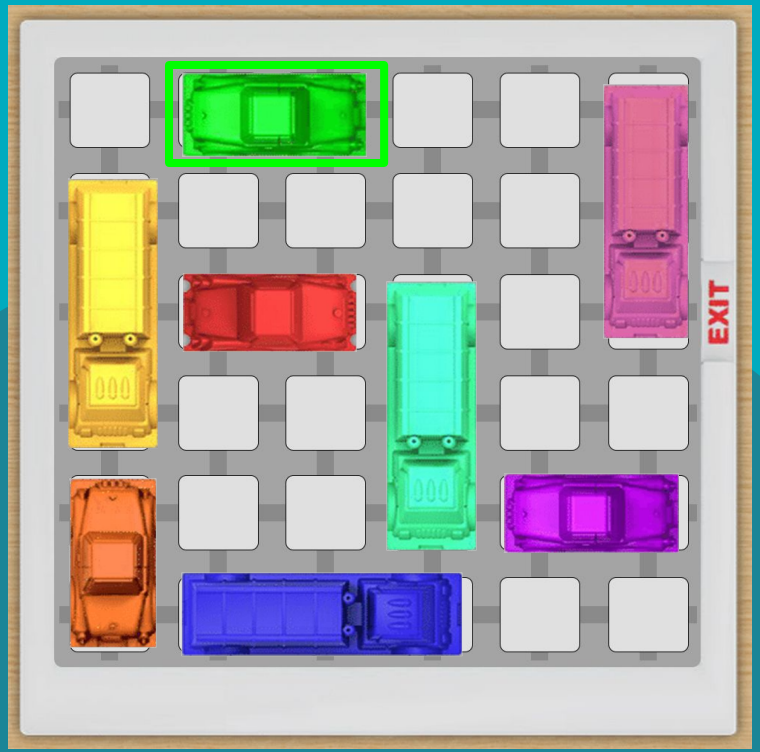


Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

Advanced Heuristic

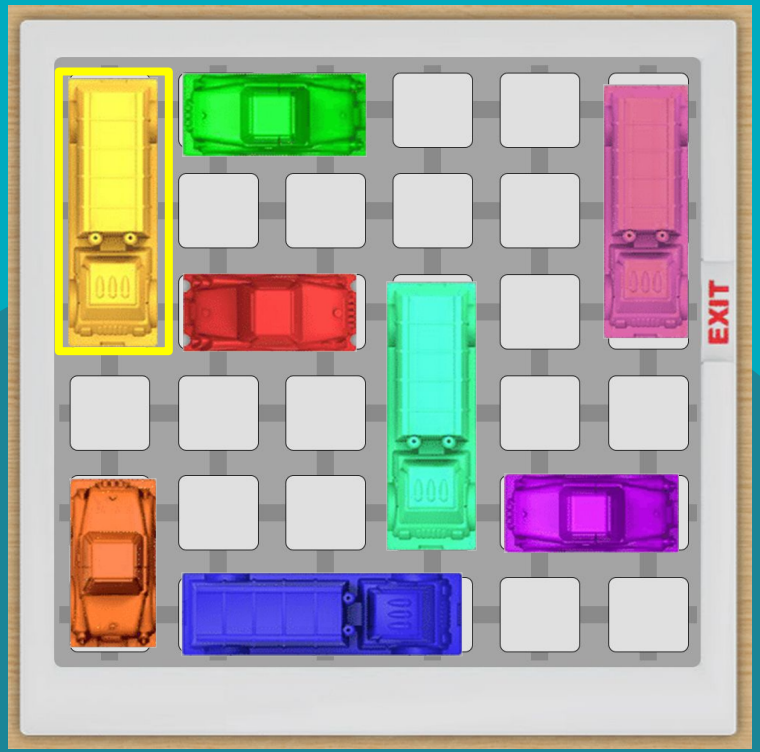
Blocking Blocking Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

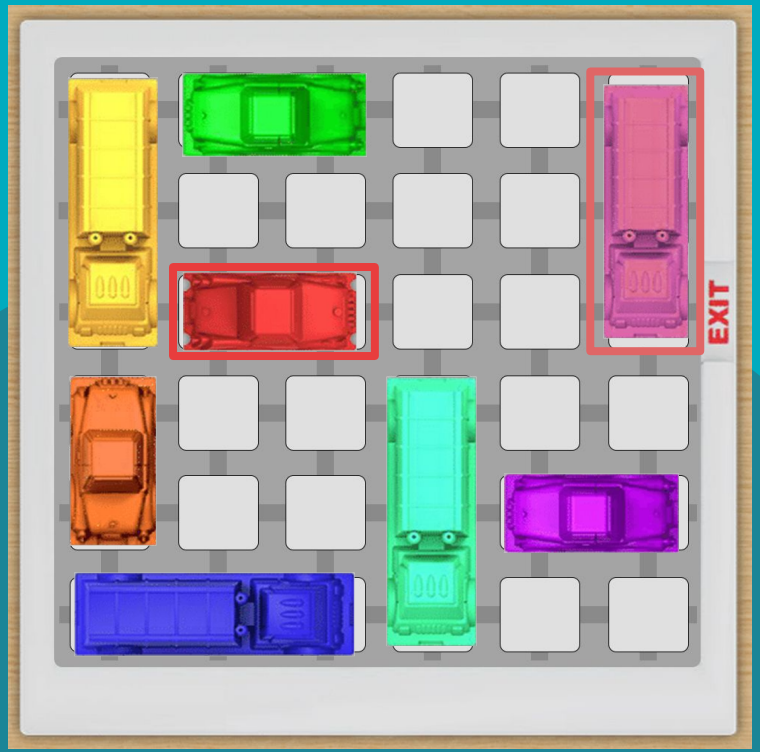
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

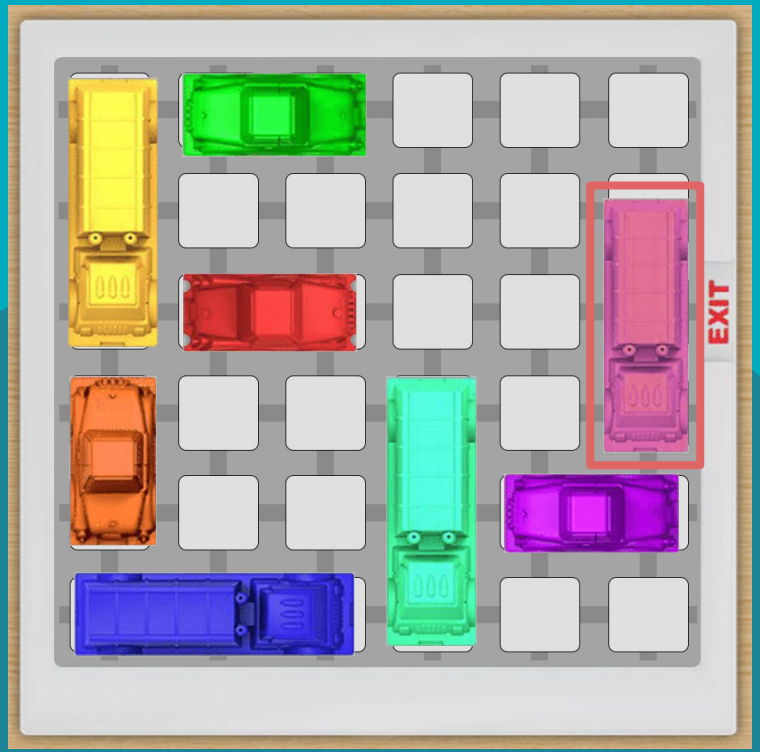
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

Advanced Heuristic

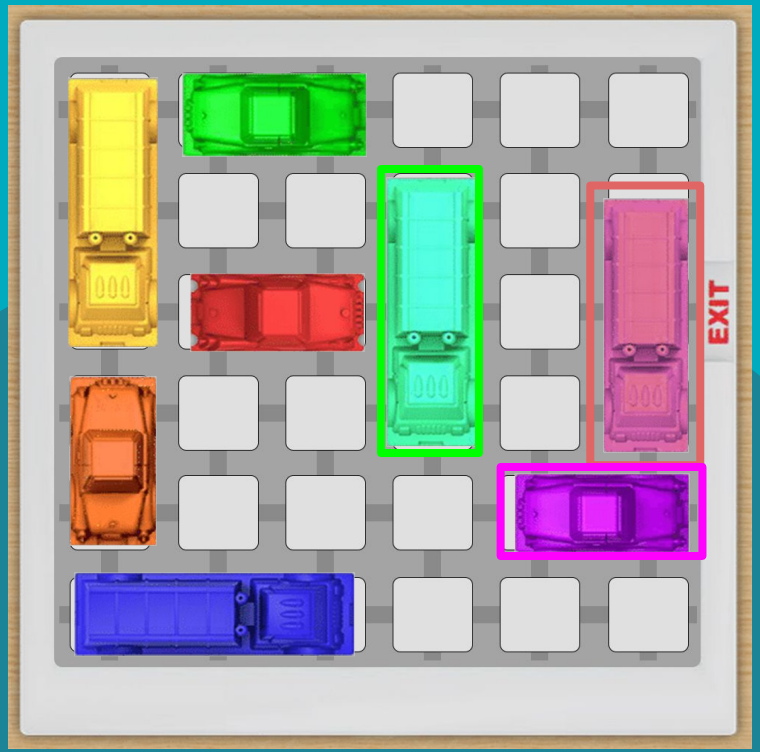


Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

Advanced Heuristic

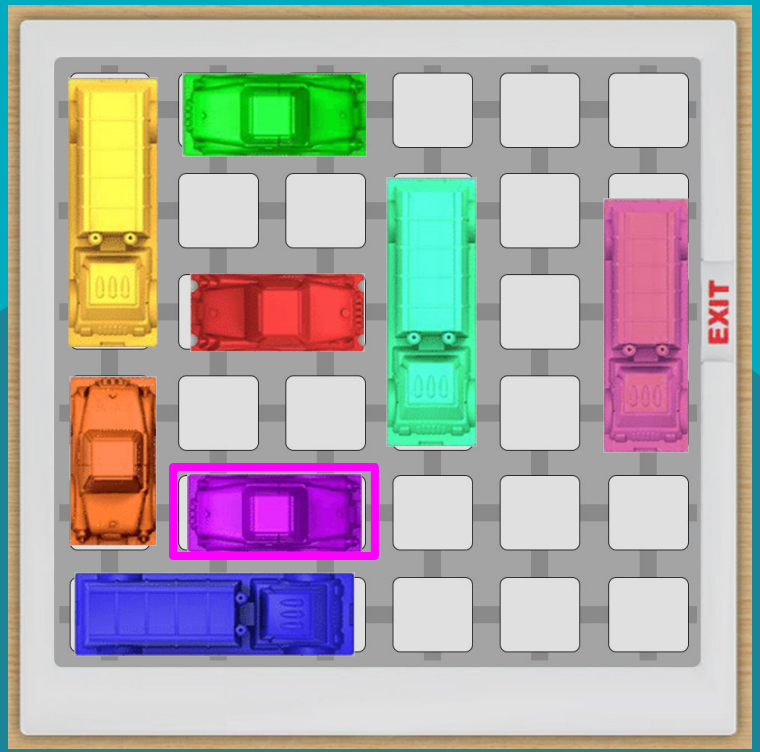
Blocking Blocking Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

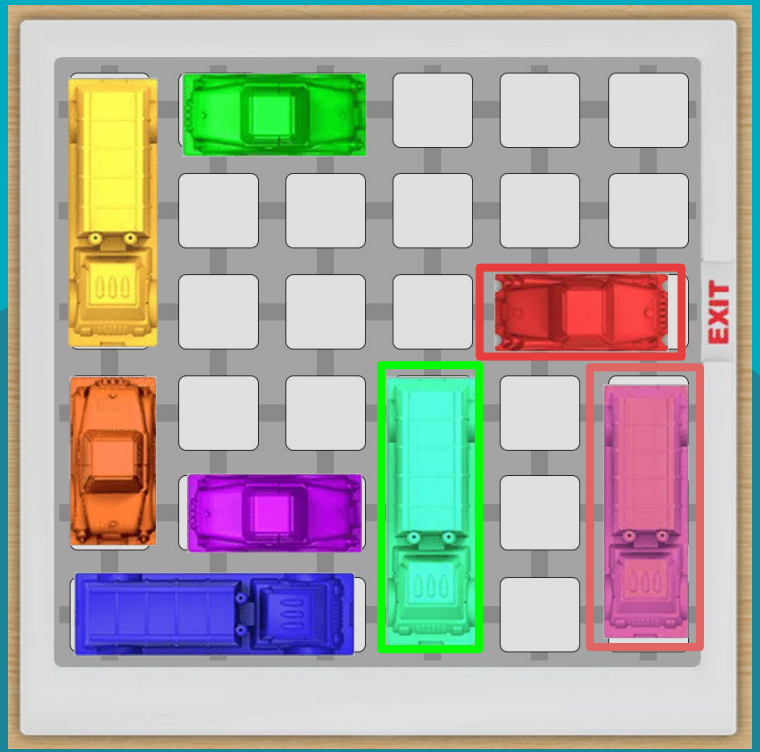
Advanced Heuristic



Blocking Blocking Heuristic

1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

Advanced Heuristic

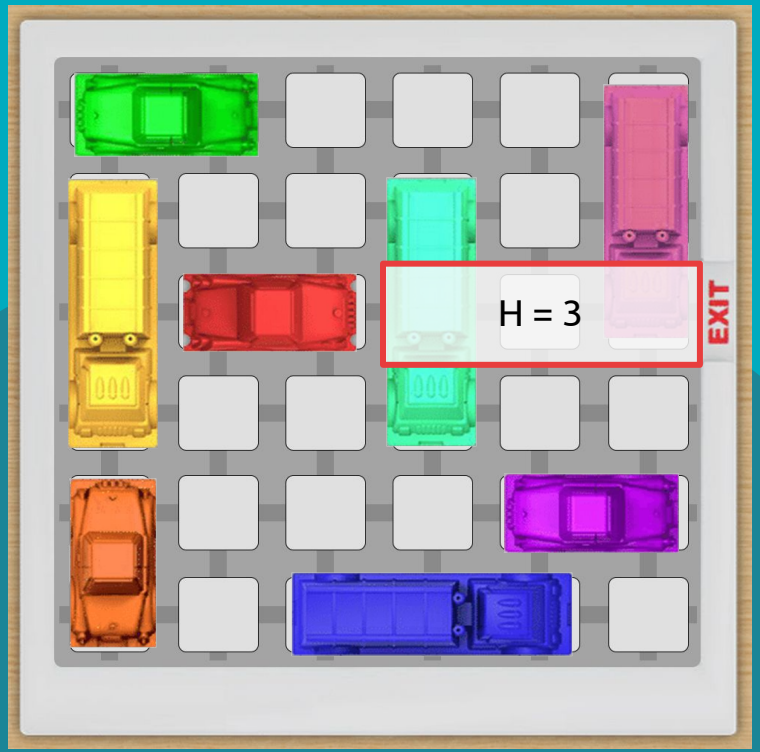


Blocking Blocking Heuristic

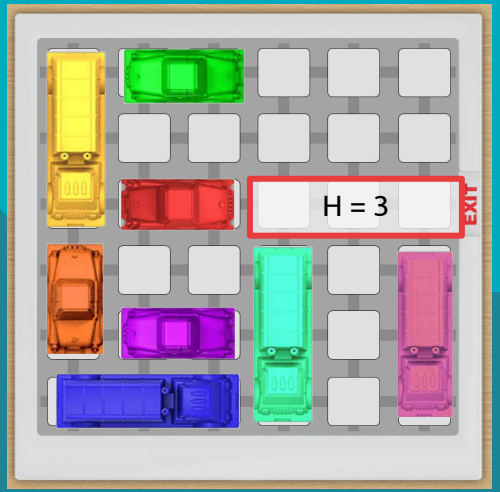
1. Checking for blocking cars
2. Checking if this car get block too
    a. Repeat until no blocking
3. Move car
    a. Repeat until the main car can drive
4. Check again for blocking cars
    a. Do Step 2

- Both the blocking and the advanced heuristic are consistent
- All blocking cars need at least one move to get out of the way
- Never overestimates the real costs

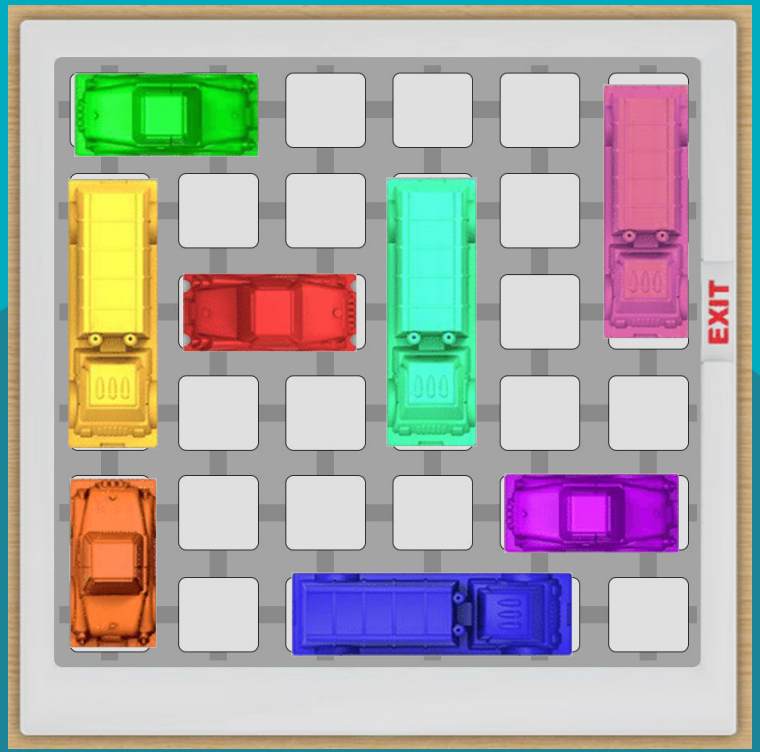
Distance to Goal



- Checking the distance to the goal
- Moves goal car and check if the distance = 0
- Not admissible because the distance could be more than steps needed



2. Advanced Heuristic

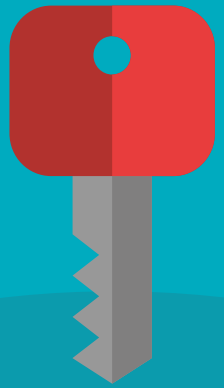


Blocking Blocking Distance

- Combination of Distance & Blocking Blocking
- Not admissible

03

Analysis



Comparison

	Zero			Blocking			Distance			Advanced			AdvancedDistance		
name	nodes	dpth	br.fac	nodes	dpth	br.fac	nodes	dpth	br.fac	nodes	dpth	br.fac	nodes	dpth	br.fac
Jam-1	11589	8	3,066	6829	8	2,857	9561	9	2,625	991	8	2,196	959	8	2,186
Jam-2	24081	8	3,378	4044	8	2,663	11407	9	2,681	2643	9	2,248	1570	8	2,340
Jam-3	7731	14	1,788	4059	14	1,699	4249	15	1,639	3688	14	1,686	3410	15	1,613
Jam-4	3203	9	2,301	1281	9	2,057	2707	9	2,255	409	9	1,781	206	9	1,627
Jam-5	21390	9	2,888	5075	9	2,433	15724	11	2,284	668	9	1,896	398	9	1,774
Jam-6	15992	9	2,791	6567	9	2,510	12786	11	2,239	2287	10	2,025	2465	11	1,901
Jam-7	52493	13	2,202	20143	13	2,035	24401	14	1,955	19262	13	2,027	18272	13	2,018

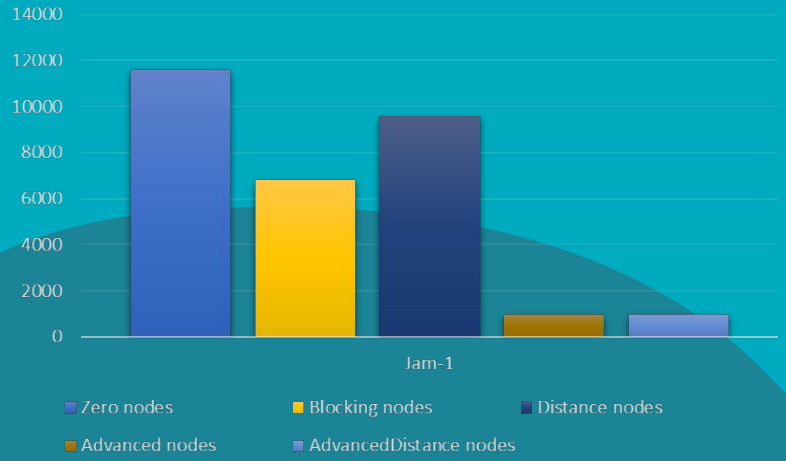
- Heuristics using the Distance are performing good but are not admissible.
- Our Advanced Heuristic performed way better than the Blocking (and Zero)

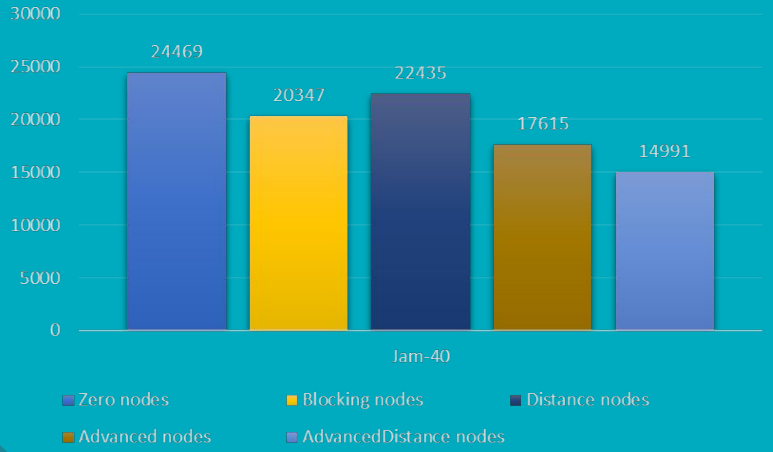
Analysis

- The more cars clustered in one spot, the harder it was for the algorithm to get to a solution
- Clusters were detected and resolved with more ease by humans
    - E.g. 7, 14, 24

- The more intricate blocking dependencies were, the harder it was for the human eye and mind to have an overview over the situation as only a few moves could be thought through before moving

Analysis





- The influence of the heuristics on the expanded nodes decreases with increasing puzzle difficulty.