Run Iterations and Results:

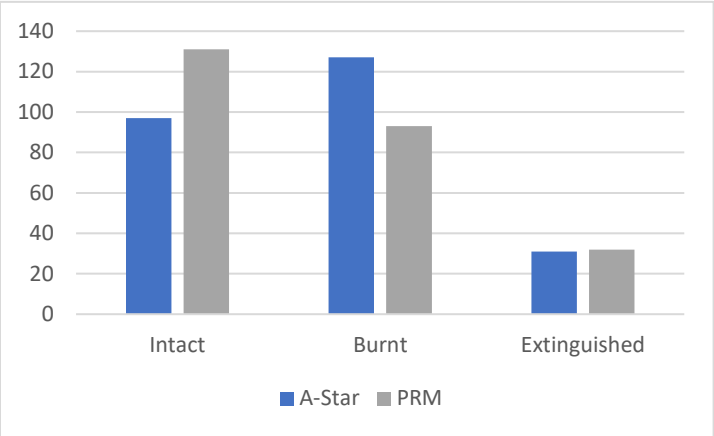
- 1ST Run

```
not working
Done
Intact 97
Burned 127
Extinguished 31
Total CPU time 0.34891843795776367
```

A-STAR

```
Done
Intact 131
Burned 93
Extinguished 32
Total CPU time 8.67925238609314
```

PRM



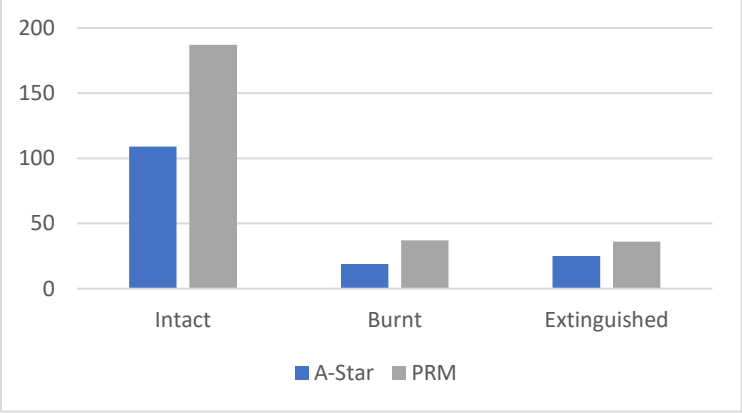
- 2nd Run

```
Intact 109
Burned 19
Extinguished 25
Total CPU time 0.20943498611450195
```

A-STAR

```
Done
Intact 187
Burned 37
Extinguished 36
Total CPU time 38.9136757850647
```

PRM

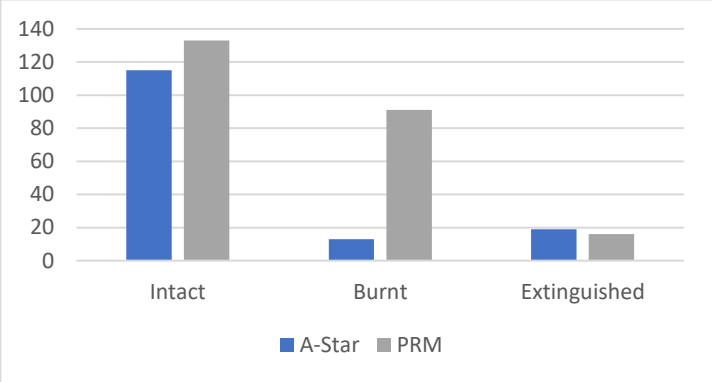


- 3rd Run

```
Intact 115
Burned 13
Extinguished 19
Total CPU time 0.31314873695373535
```

```
Intact 133
Burned 91
Extinguished 16
Total CPU time 7.280277729034424
```

A-STAR PRM

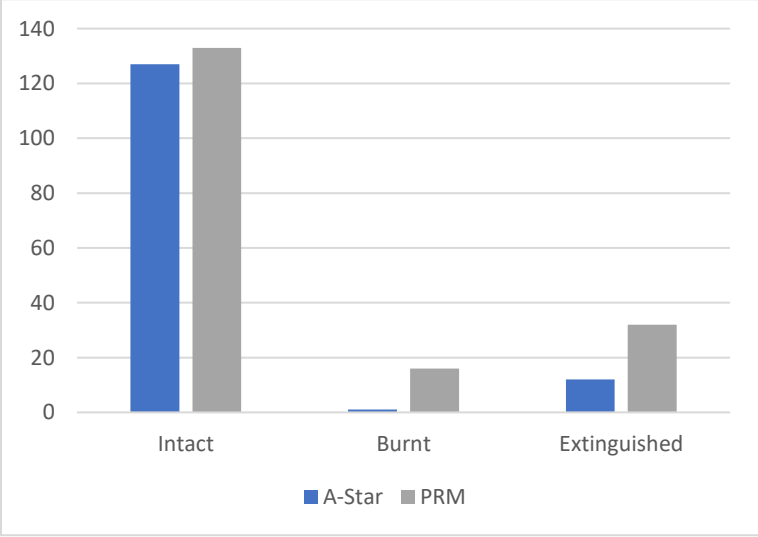


- 4th Run

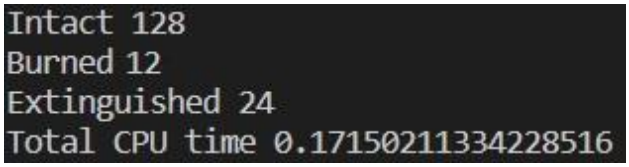
```
Intact 127
Burned 1
Extinguished 12
Total CPU time 0.13362455368041992
```

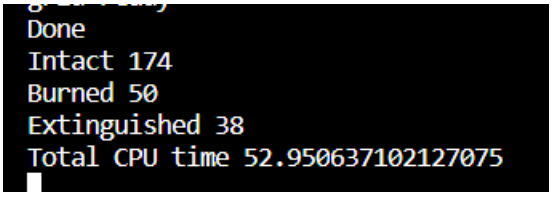
```
Intact 133
Burned 91
Extinguished 16
Total CPU time 7.280277729034424
```

A-STAR PRM

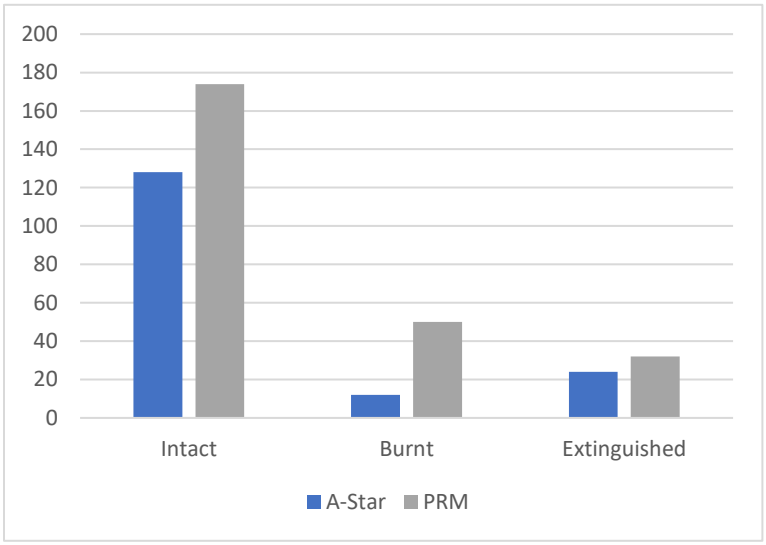


- 5th Run





A-STAR PRM



Discussion:

The experimentation yielded the following results:

- Hybrid A* was able to find optimal parts, but at the cost of exploring more space than PRM.
- Pre-computing roadmaps can help the firetruck reach its destination faster, allowing it to cover more ground before the fire spreads, thus reducing its spread.
- The graph-based planner was observed to have a higher intact ratio than the sampling-based algorithm. This is because the graph-based planner selects an optimal path for the firetruck to follow.

Reference:

[1]GitHub - AtsushiSakai/PythonRobotics: Python sample codes for robotics algorithms.

[2] Planning Algorithms / Motion Planning (lavalle.pl)