Disaster Management

A multiagent system based approach

Advisor: Prof. Dr. Jaime Simão Sichman
Co-advisor: Prof. Dr. Luis Gustavo Nardin

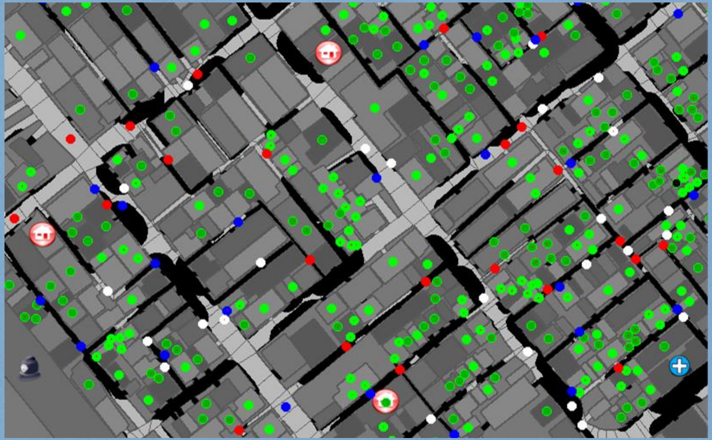
Bruno Freitas Rêgo
Gabriel Rocha Amorim
Guilherme Yuiti Miazaqui

Img. 1: Earthquake in Italy in 2016.
Source: BBC

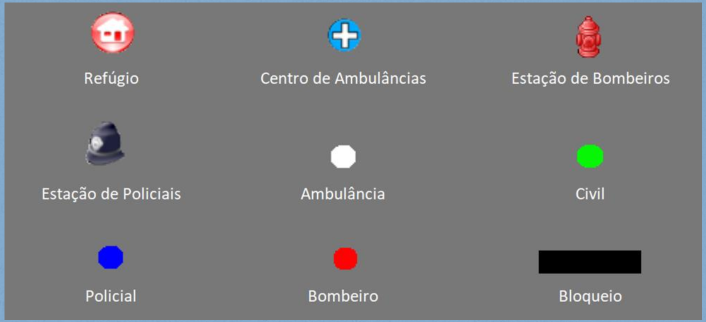
Img.2: Great Hanshin-Awaji Earthquake in Japan in 1995.
Source: iStock

Introduction and Purpose

- Natural disasters are extreme phenomena that are beyond human control
- There are Social, economic and environmental impacts
- There is partial and uncertain knowledge of the current state of the environment
- Effective rescue enable for a greater number of survivors and less economic impact
- Necessity: Better environmental exploration strategies



Img. 3: Kobe Map of RoboCup Rescue Simulation



Img. 4: Simulation entities

Introduction and Purpose

- Robocup Rescue Simulator: practical application of Multi-Agent Systems to study rescue strategies in natural disasters
- Evaluation of strategies involving Ambulances, Police and Firefighters in the rescue of civilians after a disaster
- Lack of a complete knowledge of the scenario is a characteristic of Multi-Agent Systems
- Joint efforts of entities are optimal in the mission
- Creation algorithms that implement rescue strategies

Introduction and Purpose

Focus on the ambulances

Simulator complexity requires focus of efforts

Exploring the ambulance algorithm in a context of competitions that ended up placing more value on the performance of firefighters

Disabled fire for more precise measurement

Baseline

Autonomous tasks

Sample team provided by the development framework

Team 1

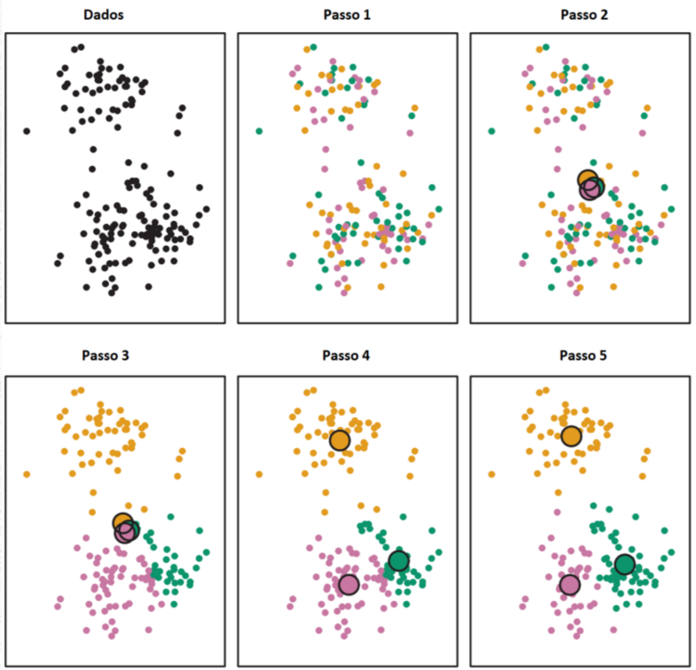
Clustering, centralized task allocation and autonomous tasks

Team 2

Team 1 + restriction of tasks exclusively to the assigned cluster

Concepts

Classification Algorithm: K-Means



Img. 5: Steps of the execution of the K-Means clustering algorithm for K = 3

Unsupervised learning

Classification of data based on similarity patterns, without the presence of a supervisor

K-Means

Partition of K clusters where each point is assigned to the cluster with the nearest means

Concepts

Wilcoxon Rank Sum Test

Nonparametric Hypothesis Test

Comparison of two unpaired groups that don't assume a normal distribution

$$z = \frac{W - \mu_W}{\sigma_W}$$

$$\mu_W = \frac{n_A\left(n_A + n_B + 1\right)}{2}$$

$$\sigma_W = \sqrt{\frac{n_A n_B\left(n_A + n_B + 1\right)}{12}}$$

Metrics

Total Score and rescue metrics

Score

Based on two main metrics: average civilian health and level of physical damage in buildings (normal simulation score)

$$Score = HPscore * buildingScore$$

$$HPscore = agentsAlive + (hpLeft/hpMax)$$

$$buildingScore = \sqrt{areaLeft/areaMax}$$

Based on two key metrics: average civilian health and level of physical damage to buildings

Rescue metrics:

- Civilians identified
- Civilians allocated
- Rescues initiated
- Civilians transported
- Civilian deaths
- Civilian Identification Step
- Task Allocation Step

Our Contributions

Centralized clustering allocation

K-Means used as an algorithm in defining clusters

Greater number of ambulances allocated to regions with many buildings

Centralized partitioning

Target

Target allocation only within the same cluster as the ambulance

Target's cluster identification

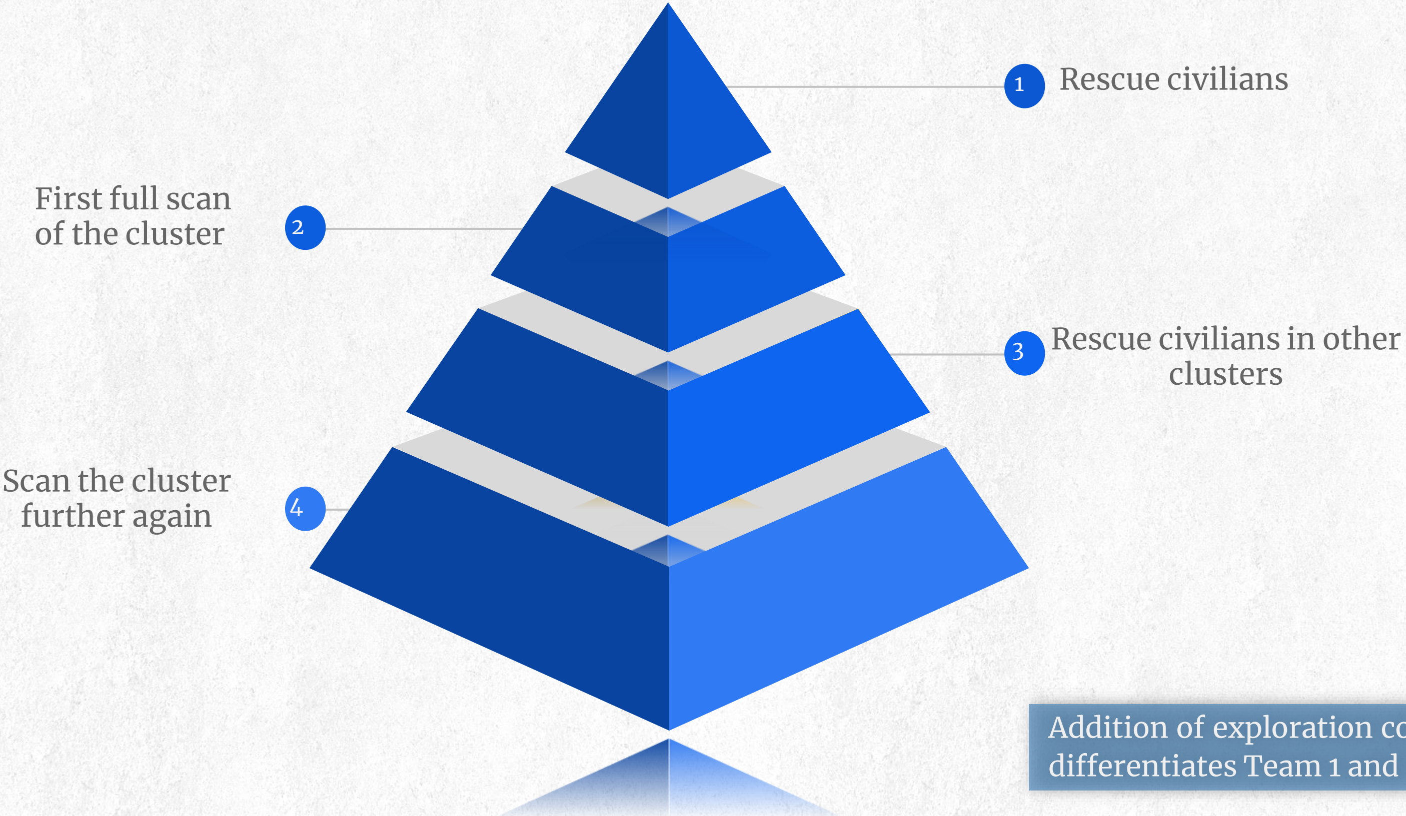
Autonomous behavior in the lack of assigned tasks

Different priority lists

Our Contributions

Controlled Exploration

Order of Priorities in the Cluster



Addition of exploration commands is what differentiates Team 1 and Team 2

Our Contributions
Debug Viewer

Graphical display of relevant information

- Cluster buildings
- Cluster buildings to be explored next (Team 2)
- Civilians identified
- Ambulances transporting civilians

Img. 6: Debug Viewer used on the Berlin2 map

Our Contributions
Bash Script

- Automation of simulation log collection
- Each simulation can generate up to 8 GB of data
- Up to 45 minutes per simulation
- 3,5 TB of total data

```bash
for K in ${CLUSTERS[*]};
do
  # Change the K value within the code of the team
  # Clean project and recompile
  for MAP in ${MAPS[*]};
  do
    for SEED in ${SEEDS[*]};
    do
      # Change the simulation seed in the server
      # Create folder in the format team-name/map/k/seed
      # If it exists, skip simulation
      # Start server
      # If the debug flag is active, start Visual Debugger
      # Start team
      # Wait for the end of simulation
      # Close agent
      # Copy simulation output to log folder passed as a parameter
      # If open, close Visual Debugger if open
      # Extract score lines from kernel log
    done ;
  done ;
done ;
```

Img. 7: Bash Script to automate log collection

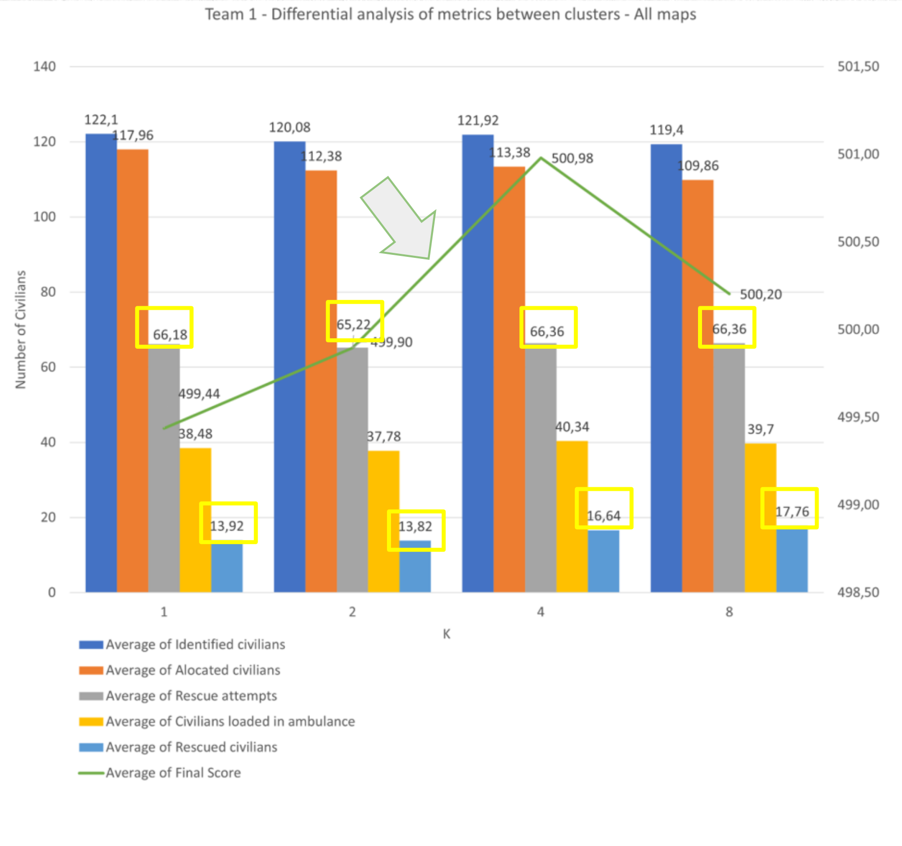
Experiments

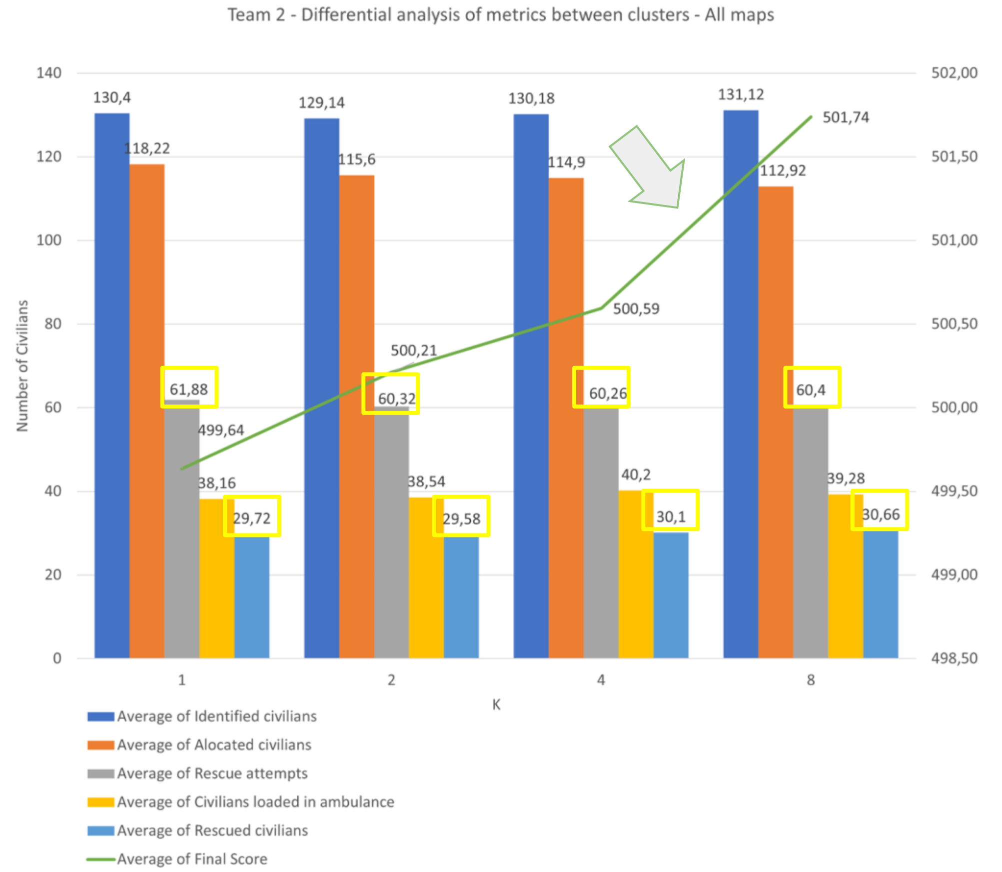
- Team 1, Team 2 and Sample (Baseline)
- 4 values of K: 1, 2, 4 e 8
- 5 maps: Berlin, Joao, Kobe, Paris and Berlin2
- 10 different seeds
- 450 simulations
- Comparison with Sample Team

	Time 1		Time 2		Sample		Total	
Mapa	Média de Score	StdDev	Média de Score	StdDev	Média de Score	StdDev	Média de Score	StdDev
Berlin	**698,31**	**43,56**	**698,12**	**43,91**	**699,24**	**40,99**	**698,55**	**42,84**
K = 1	699,00	42,23	698,04	44,22	699,24	40,99	698,76	42,51
K = 2	697,85	44,41	698,05	44,26	699,24	40,99	698,38	43,25
K = 4	698,22	43,87	697,85	44,43	699,24	40,99	698,44	43,13
K = 8	698,15	43,67	698,53	42,72	699,24	40,99	698,64	42,48
Joao	**153,28**	**8,00**	**153,03**	**8,29**	**153,85**	**9,07**	**153,39**	**8,47**
K = 1	153,05	8,50	153,05	8,29	153,85	9,07	153,32	8,64
K = 2	153,15	7,51	152,95	7,94	153,85	9,07	153,32	8,21
K = 4	153,35	8,12	153,25	8,38	153,85	9,07	153,49	8,54
K = 8	153,55	7,84	152,85	8,51	153,85	9,07	153,42	8,50
Kobe	**448,05**	**5,58**	**448,77**	**5,80**	**436,27**	**6,32**	**444,36**	**8,23**
K = 1	445,89	5,61	448,10	4,86	436,27	6,32	443,42	7,62
K = 2	447,69	4,39	447,49	7,04	436,27	6,32	443,82	8,05
K = 4	449,30	5,97	449,40	5,21	436,27	6,32	444,99	8,50
K = 8	449,30	5,48	450,10	5,46	436,27	6,32	445,22	8,57
Paris	**439,20**	**14,64**	**441,63**	**14,27**	**431,25**	**14,82**	**437,36**	**15,24**
K = 1	438,15	13,97	442,15	13,87	431,25	14,82	437,19	14,92
K = 2	440,45	16,01	441,35	13,84	431,25	14,82	437,69	15,60
K = 4	439,85	13,31	440,95	14,15	431,25	14,82	437,35	14,76
K = 8	438,35	15,01	442,05	15,14	431,25	14,82	437,22	15,65
Berlin2	**761,82**	**30,39**	**761,18**	**30,30**	**755,25**	**28,25**	**759,41**	**29,81**
K = 1	761,09	31,19	756,83	32,09	755,25	28,25	757,72	30,66
K = 2	760,33	30,17	761,21	28,36	755,25	28,25	758,93	29,06
K = 4	764,18	29,18	761,52	30,26	755,25	28,25	760,32	29,48
K = 8	761,67	30,83	765,16	29,79	755,25	28,25	760,69	29,92
Total	**500,13**	**218,00**	**500,55**	**217,73**	**495,17**	**217,33**	**498,62**	**217,70**

Table 1: Scores obtained on maps using different K values and testing different teams

Experiments





Images 8 and 9
To the left: Evaluating Rescue Metrics for Team 1
Above: Evaluating Rescue Metrics for Team 2

Wilcoxon Rank Sum Test

Parameter	Map	Team a	Average	Team b	Average	statistic	p-value
Dead civilians	Kobe	Sample	64,40	Team 1	51,92	4,075	0,000046
Dead civilians	Kobe	Sample	64,40	Team 2	52,27	3,881	0,000104
Step of civilian death	Kobe	Sample	137,76	Team 1	127,01	2,692	0,007099
Step of civilian death	Kobe	Sample	137,76	Team 2	127,13	2,765	0,005694
Identified civilians	Berlin2	Team 1	220,95	Team 2	235,38	-1,968	0,049091
Identified civilians	Kobe	Team 1	191,38	Team 2	202,97	-3,026	0,002476
Identified civilians	Paris	Team 1	146,22	Team 2	163,88	-3,594	0,000326
Rescued civilians	Berlin2	Team 1	38,23	Team 2	60,05	-5,398	0,000000
Rescued civilians	Kobe	Team 1	33,10	Team 2	73,95	-7,332	0,000000
Rescued civilians	Paris	Team 1	5,10	Team 2	15,38	-7,289	0,000000
Step of civilian rescue	Berlin2	Team 1	132,43	Team 2	138,24	-3,315	0,000917
Step of civilian rescue	Paris	Team 1	252,02	Team 2	203,62	6,795	0,000000

Tabela 4: Wilcoxon tests for different teams in the same map.

Relevant metrics

- Dead civilians, Identified civilians, Rescued civilians, Civilians being rescued, Score
- Kobe, Berlin2, Paris
- Differences between Team 1, 2 and sample

Parameter	Team 1	Team 2	statistic	p-value
Step of allocation	108,44	116,07	-3,278	0,001045
Step of allocation	95,9	100	-2,156	0,031060
Rescued civilians	15,54	30,02	-2,769	0,005628
Civilians being rescued	66,03	60,72	2,841	0,004498

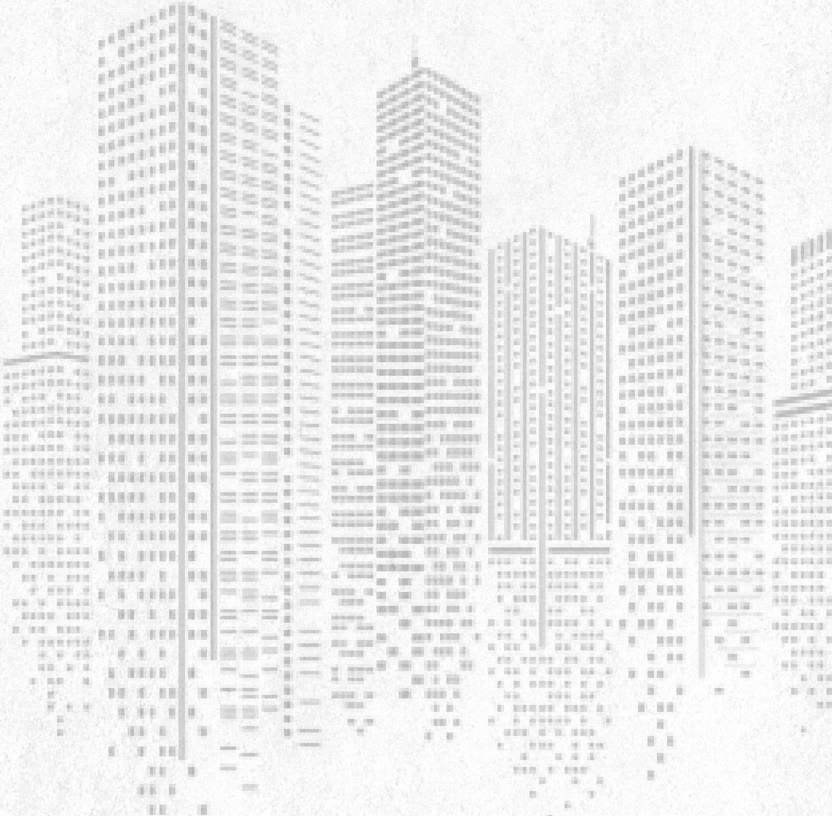
Tabela 5: Wilcoxon tests for Team 1 e Team 2.

Parameter	Map	K	Average	K	Média	statistic	p-value
Dead civilians	Kobe	1	57,00	4	51,05	2,683	0,007290
Dead civilians	Kobe	1	57,00	8	50,65	2,733	0,006280
Final score	Kobe	1	443,42	4	449,35	-2,693	0,007077
Final score	Kobe	1	443,42	8	449,70	-2,654	0,007964
Rescued civilians	Berlin2	1	44,50	8	55,30	-2,380	0,017293

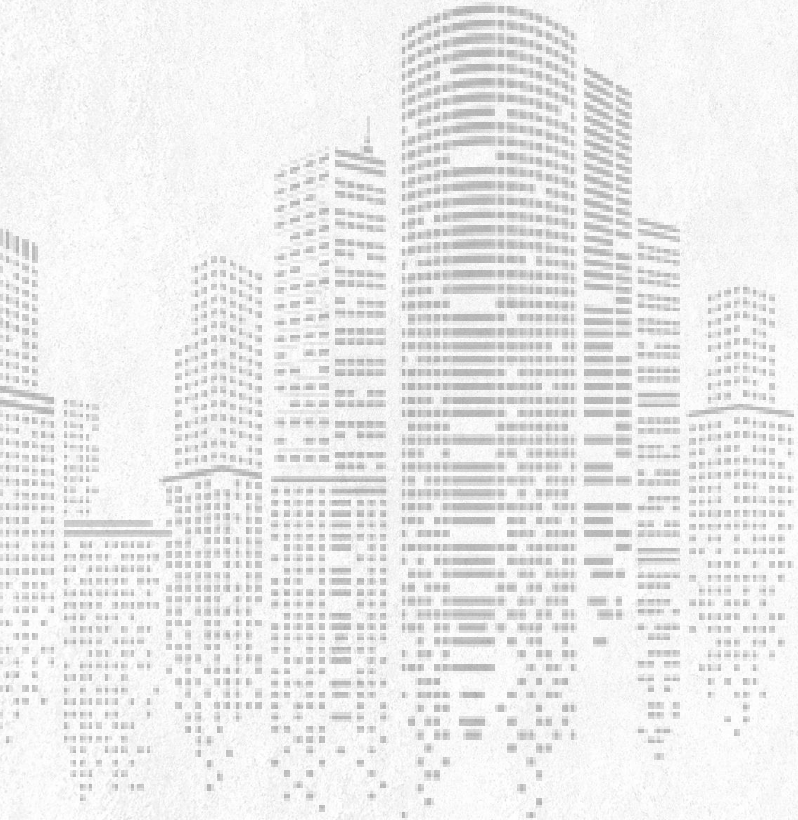
Tabela 6: Wilcoxon tests for different K's.

Conclusion

- Centralized K-Means clustering algorithm showed positive impact mainly on maps with fewer blockades, like Kobe and Paris
- The statistical relevance of centralizing clustering was more relevant throughout the parameters than changing the value of K
- The impact of restricting ambulances to their own cluster (Team 2) was more relevant than the impact of changing the value of K



Future Works



Implementation of other algorithms such as DBSCAN

- Advantage: it is not necessary to inform K

Evaluate results in different maps

- Specific aspects such as spatial distribution of buildings may influence

Compare results with non-centralized algorithms

Use K-means in a supervised manner

- Such as using refuges as cluster centers