Distributed Localization in Partially GPS Denied Environments

Nishant Shah
Department of Robotics Engineering
Worcester Polytechnic Institute
Worcester, Massachusetts 01609
Email: nshah3@wpi.edu

Tesia Shizume
Department of Computer Science
Worcester Polytechnic Institute
Worcester, Massachusetts 01609
Email: ttshiz@wpi.edu

Amey Sutavani
Department of Robotics Engineering
Worcester Polytechnic Institute
Worcester, Massachusetts 01609
Email: aasutavani@wpi.edu

***Abstract*—The goal of this project was to develop a technique for distributed absolute localization of robot swarms that interoperate between GPS accessible and GPS denied environments. Such scenarios typically arise in search and rescue operations for disaster management tasks. Especially, in calamities such as earthquakes, where it is impossible for humans to reach victims trapped under collapsed debris and structures, an army of small robots that can crawl through tight spaces between the rubble to locate victims can be a life-saving tool. However, such robots would need to locate themselves reliably in an hostile GPS denied environment (such as a collapsed building). This problem can be approached by combining previous work on Simultaneous Localization And Mapping (SLAM) [3] and Distributed Relative Localization of Swarms [4] with Wireless Sensor Networks.**

I. Introduction and Background

The information available to robotic swarms is not always consistent across all the robots. This can happen through either environmental factors or failure of robotic components. Environmental factors that can contribute to these inconsistencies include constriction points in the terrain, areas with overhangs like caves or buildings, or areas with partial Faraday cage effects due to high concentration of ferrous materials in the walls or other structures in the environment. One such case was Operation Surya led by Indian Army during floods in 2013 at Kedarnath which was caused by monsoon rains, flash floods, and landslides. During this calamity, as many as 207 mobile towers were knocked down by the fury of the floods and approximately 10,000 troops were deployed to rescue and help the needy people. Today, we can use a number of robots and map and scan the area affected and deploy the rescue measures as per requirement. However, such systems heavily depend on GPS based localization for navigation, which limits their usage in situations like going in an underground cave or inside a metallic bunker. Failure of some sensors can cause an error in navigation and render the robot inoperable. Furthermore, when this inconsistency in information is on localization information this can pose challenges to determining the most appropriate behavior for the swarm and especially individual members of the swarm. In the case of damaged but still otherwise operable robots, the ability of the swarm to provide localization information may improve the chances of recovering said robots. In the case of disparities in localization information, the ability of the swarm to provide absolute localization information may aid in many other tasks including: obstacle avoidance, exploration, or search and rescue. Furthermore, for mixed ability robot swarms reducing the number of robots that have GPS may offer significant cost savings or the ability to use other resources on limited systems. Thus, the precise and accurate absolute localization of robotic swarms is an important ability to develop.

A. Prior Work

The challenge of localization has long been approached through the use of various techniques. In [8], problems about the localization of two robots without any prior information of each others location has been discussed. In this paper, two robots are initialized each unaware of others location. As they navigate, the robots wirelessly share and match laser-scans attempting to solve for the others pose in the local frame. After observing a common area, the robots compute a transformation between their local coordinates frames. Thus a combined 3D map is initialized and the map and estimated transform are refined online based on new sensor measurements. The combined map contains sections independently explored by each robot. To accomplish this strategy, pose correspondents are built by matching sensor measurements shared by each robot. Robots can localize to one another online, even after being initialized in different buildings. Hence larger areas can be explored using multiple robots.

In [5], the overall goal is to perform a building-clearing mission, where a swarm of robots enter whose layout is unknown. The robots then disperse through the building and attempt to locate an object of interest. Once the object has been located, the swarm remains in the building to protect the item of interest until friendly forces arrive. For this, they developed distributed algorithms. Their solution consisted of collaborative localization algorithm, dynamic task allocation algorithm and collaborative mapping. The authors showed some promising results using both simulated and real robots.

In [3] , they use Triangulation and Probabilistic techniques. Triangulation technique use simple geometric properties along with probability to calculate the location of an object from the locations of other objects.

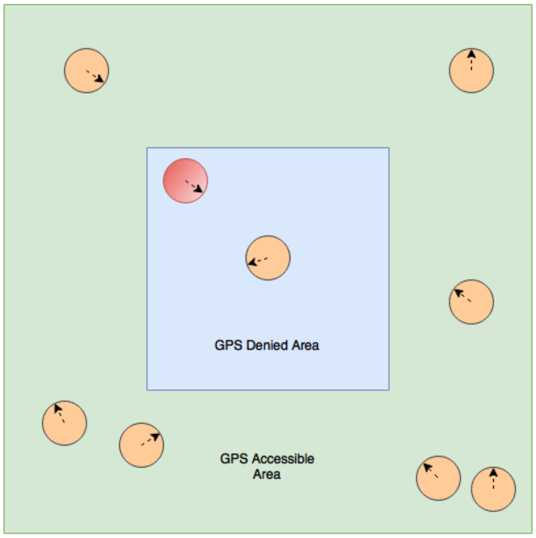


Fig. 1: Arena with GPS Denied and Accessible areas. The term GPS Denied is used to describe areas where the robots cannot obtain their own GPS location data. The term GPS Accessible is used to describe the areas where the robots can obtain their own GPS location data. The robot in red is the point of interest.

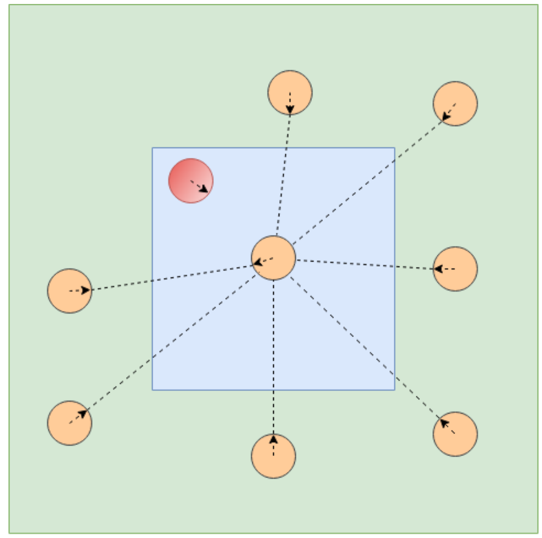


Fig. 2: Arena showing swarm providing localization information to a robot in GPS Denied area.

Other pertinent localization approaches include GPS [7] [6] and the approaches suggested by Blusu, Heidemann and Estrin. [1]. The basics of GPS involve the use of a constellation of satellites to enable the global use of a modified triangulation localization technique. By having more than three satellites visible to receivers across the globe the accuracy of the triangulation is improved. In practice, this is frequently supplemented with other localization data to further increase the accuracy of the localization. The approach suggested by Blusu, Heidemann and Estrin utilizes periodic short-range radio frequency beacons from a fixed number of reference points, in a similar system to GPS. However, they use an idealized radio model that assumes perfect spherical radio propagation and identical transmission range for their signal.

II. Experimental Setup

This project team proposes to simulate the effects of part of a swarm entering an absolute localization information sparse area, like a cave, and develop accurate and effective methods for the absolute localization of the swarm members in the information sparse area.

A. Representation of the Environment

The arenas were be divided into regions with different levels of absolute localization information available directly to the individual robots in the swarm, as shown in graphically in Figure 1 and in the ARGoS simulator in Figure 3.

B. Selection of Robots

For the purposes of this project the robots needed communication, localization, and locomotion abilities. The Khepera IV robot in as simulated in ARGoS provides all three of these properties. It has LED which can be used to convey information to other robots. It has a positioning sensor described as "a sort of GPS" which can give absolute localization information. It also has wheels and an actuator for the wheels which can be used for locomotion. [2] It also has a bottom mounted infrared sensor that can be used by the robot to detect the floor color. In our experimental setup we plan to differentiate the GPS accessible area from GPS denied area by giving different colors to the tiles of the arena. To simulate

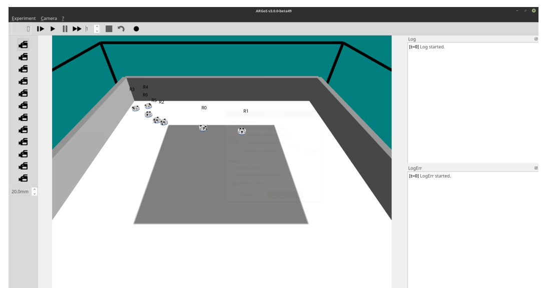


Fig. 3: Arena without Wall

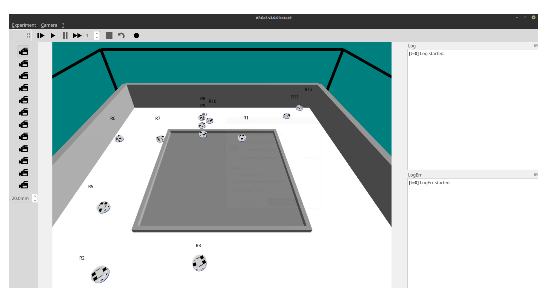


Fig. 4: Arena with Wall

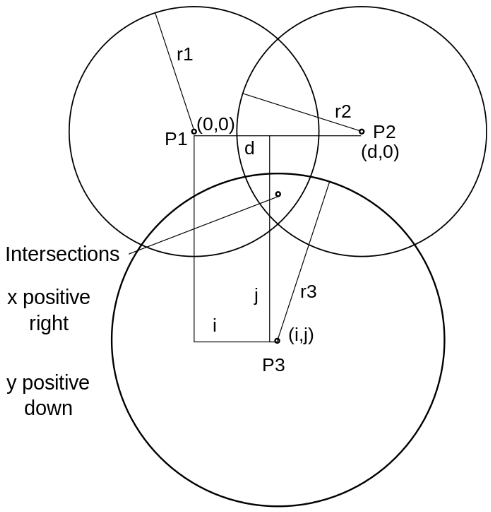


Fig. 5: Geometry of the trilateration algorithm. The GPS accessible robots are located at points P1, P2 and P3 with distances r1, r2 and r3 from the GPS denied robot, located at the intersection of the three circles.

the information provided by GPS in real Khepera IV robots, one of two techniques will be utilized. If the experiments are conducted in the lab the Vicon system will be used to provide the localization information. Otherwise or in addition markings in the arena and the downward facing sensor will be used. This project also used the Foot-bot simulated in ARGoS which has a blob camera which can be used for simple target location finding.

C. Description of Task

In the environments described in II-A, the swarm will be tasked with first enabling a rescue robot to localize itself absolutely in a GPS denied region and second locating and providing an absolute localization for a point of interest in the GPS denied region, as shown in Figures 2 and **??**.

III. Algorithm

A. Trilateration Algorithm

For localizing a robot in a GPS denied environment, we use the Trilateration algorithm. This algorithm essentially computes the intersection of three spheres. We start with a robot K in the GPS denied environment. This robot is surrounded by the neighbour robots who HAVE access to GPS information as shown in Figure 5. The neighbours broadcast their own location and distance to the robot K periodically. From all the available neighbours, select broadcasted information from 3 robots who themselves are located at points (P1, P2 and P3) with distance (r1, r2 and r3) respectively from robot K. Here, P1, P2 and P3 are 3x1 vectors and r1, r2 and r3 are scalars. With this information, the robot K can compute its own location by the algorithm shown below. This algorithm can then be repeated for all of the neighbour triplets of the robot K. The resulting location from all such triplets is then combined to get the absolute position estimate.

The following algorithm computes the intersection of three spheres. (Trilateration)

1. Given a robot K in GPS denied environment. This robot is surrounded by the neighbour robots who HAVE access to GPS information. The neighbours broadcast their own location and distance to the robot K periodically.
2. From all the neighbours, sample broadcasted information from 3 robots located at points (P1, P2, and P3) with distance (r1, r2, and r3) respectively from robot K. [P1, p2, and p3 are 3x1 vectors and r1, r2, and r3 are scalars]
3. Get the unit vector in direction from P1 to P2
   - $\widehat{e_x} = P2 - P1/\|P2 - P1\|$
4. Get the signed magnitude of the x component of vector P1 to P3
   - $i = \widehat{e_x} \cdot (P2 - P1)$
5. Similarly, get the unit vector in y direction
   - $\widehat{e_y} = P3 - P1 - i\widehat{e_x}/\|P3 - P1 - i\widehat{e_x}\|$
6. The third basis unit vector is given as:
   - $\widehat{e_z} = \widehat{e_x} \times \widehat{e_y}$
7. Distance between the points P1 and P2
   - $d = \|P2 - P1\|$
8. Get the signed magnitude of y component of vector P1 to P3
   - $j = \widehat{e_y} \cdot (P3 - P1)$
9. The location of the robot K can then be given as:
   - $P = P1 + x\widehat{e_x} + y\widehat{e_y} + y\widehat{e_y}$
   - $x = (r_1^2 - r_2^2 + d^2)/2d$
   - $y = (r_1^2 - r_3^2 + i^2 + j^2)/2j - (i/j)x$
   - $z = \sqrt{(r_1^2 - x^2 + y^2)}$

IV. Experiments

Initially, the test was carried out in a static environment, in which a total of 14 robots were placed in the environment, out of which 12 robots were placed in the GPS accessible area, 1 stranded robot and 1 rescuer robot was placed in the GPS denied environment. This was the case for every experiment that we carried out. In this configuration, we also tested the impact of noise on the performance of the algorithm, since its impact would be clear due to it being the only change in a static environment. The rest of the stages of our experiment were specified by how they varied the experimental parameters and in each stage we analyzed the effect of varying that parameter on the accuracy of the localization of the rescue robot.

A. Assumptions

The experiments were carried out in simulation with following assumptions:

1. No communication delay between robots when receiving range and bearing broadcast information.
2. Robots perform perfect motion (no wheel slip)

3. During early trials, there is no noise in the GPS information (absolute location) was be kept binary
   - Either robot knows its position through GPS or doesn't
   - Later, noise was added to the available GPS information and the localization accuracy was tested.

B. Parameters Explored

The later stages of our experiment, after testing the basic static case, were specified by how they varied the experimental parameters. The parameters explored were:

1. The effect of adding sensor noise.
2. The effect of varying the percent of GPS robots moving with static rescue and disabled robots.
3. The effect of moving the rescue robot with static GPS robots.
4. The effect of varying the percent of GPS robots moving with moving rescue robot and static disabled robot.
5. The effect of moving rescue robot purposefully with moving GPS robots and static disabled robot.

V. Results

For all the graphs shown below, the dashed line indicates the average (estimated) X and Y position of the rescuer robot to be located in the GPS denied environment. The shaded area in the graph shows the minimum and the maximum coordinate values estimated for the rescuer robot and the solid line shows the absolute(real) location of the rescuer robot.

A. Static Robots

For the first experiment, all the robots in the GPS accessible and GPS denied environment were kept static and the noise in the sensor was kept at zero. Multiple random starts were tested. The results for two of the seeds can be viewed in Figures 6 and 7.

To observe the effect of adding sensor noise, in the second experiment, all the robots in the GPS accessible and GPS denied environment were kept static and 0.01 noise was added to the sensor. Multiple random starts were tested. The results for two of the seeds can be viewed in Figures 6 and 7.

For the second experiment, all the robots in the GPS accessible and GPS denied environment were kept static and 0.01 noise was added to the sensor. Multiple random starts were tested. The results for two of the seeds can be viewed in Figures 10, 11, 12, 13, and 14.

B. Moving Robots

To observe the effect of varying the percent of GPS robots moving with static rescue and disabled robots. Trials were run with 10%, 30%, 50%, 70%, and 90% of the GPS robots moving. The results for these trials are shown in Figures 10 and 11.

For the fourth experiment, all the robots in the GPS accessible environment were kept static and the rescue robot was moving. The results of this experiment can be seen in Figure 15.

To observe the effect of varying the percent of GPS robots moving with moving rescue robot and static disabled robot, trials were run with 30% and 70% of the GPS robots moving. These combined with the fourth experiment (see figure **??** gave three data points that indicated performance at no, low, and high amounts of motion. The results for these trials are shown in Figures 16 and 17.

The last experiment was designed to observe the effect of moving rescue robot purposefully with moving GPS robots and static disabled robot. Foot-bots were used for this experiment, since they have a blob camera that could locate an LED beacon lit on the disabled robot. This is meant to simulate rescue of the disabled robot. In this experiment the rescue robot located the disabled robot by use of the LEDs and camera and traveled towards the disabled robot. The results of this experiment can be seen in Figure 18.

VI. Analysis

As expected, the results shown in Figures 6 and 7 indicate that the localization is highly accurate when all robots are stationary. The spread between the maximum and the minimum estimated positions for both X and Y and the error associated with those positions is very essentially zero. Similarly, the actual position lines follow the estimated position lines so precisely that they appear as one line in the visualization. Adding sensor noise, as shown in Figures 8 and 9 increased the spread of the estimate maximum and minimum possible values for X and Y but still kept the estimated position very close to the actual position. The increase in this error can be seen in the error plots where there is a wider spread. Even with the wider spread the error is still constrained to approximately plus or minus 4 centimeters. The effect of varying the percent of GPS robots moving with static rescue and disabled robots was minimal on the accuracy of the localization. As shown Figures 10, 11, 12, 13, and 14 The effect of moving the rescue robot with static GPS robots and a significant impact on the accuracy of the localization and also the calculated minimum and maximum values, as shown by the spread in Figure 10. Here it can also be observed that there is a lag in the estimated position in reacting to changes in the actual position. This may indicate that a different filter might provide better performance over the average filter used in this study. The effect of varying the percent of GPS robots moving with moving rescue robot and static disabled robot reflect the observations made in previous sections of the project as shown in Figures 15, 16, and 17 as the errors combine to produce a very wide spread in the estimated minimum and maximum positions. The effect of moving rescue robot purposefully with moving GPS robots and static disabled robot also reflected the previous results. At the start of the experiment the robot was stationary since it was still locating the disabled robot, this is reflected in Figure 18 where the predicted locations follow the actual locations tightly for the first portion of the position graphs. Similarly, the spread between the minimum calculated position and the maximum calculated position is very small for this portion. Then the predicted and actual position diverge more while the Foot-bot is moving towards the target. Similarly the minimum and maximum values show an increase in spread. As the robots velocity stabilizes the spread from minimum to maximum possible position reduces. Then, as expected, the accuracy significantly increases once the Foot-bot has reached the disabled robot and becomes stationary. While carrying out the experiments, the experimenters found that maintaining the consistency of the GPS beacon data, especially with the moving beacon was difficult. Its location and its distance from the rescue robot was continuously changing. The team figured out that this inconsistency was mostly due to the lag in transmitting the data from the GPS accessible robots to the GPS denied robot in the simulator. The team attempted to solve this problem by adding some bias to the values received by the GPS denied robot, by analyzing the transmitted data continuously and observing the error. This could also be solved by putting time-stamps on beacon messages like the real GPS transponder in future work. The team also noticed that the average filter is sensitive to the spikes in the estimates. A median filter or a low pass filter can solve this problem and were considered but these were ruled out due to their introduction of latency in the estimation. The team hypothesizes that ideally, a Kalman filter could be used to provide the needed fusion in the predicted position from trilateration and predicted position obtained by integration of velocity.

VII. Conclusion and Future Work

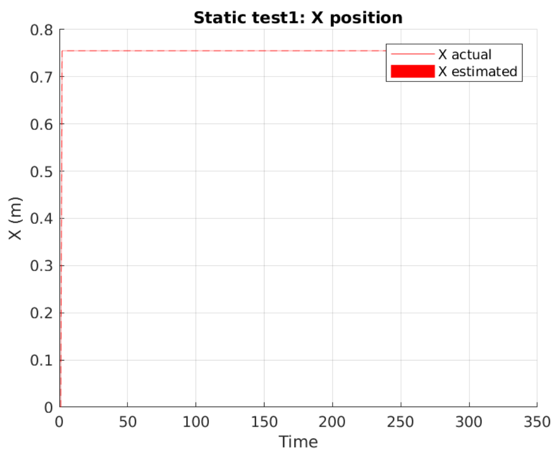
It can be concluded that the movement of the Rescue Bot itself introduced the most inaccuracy in localizing itself. Similarly, the presence of noise in the sensor data transmitted by the GPS accessible robots to the rescue robot in the GPS denied environment introduced the next most inaccuracy.

In these experiments the filter used was an average filter. Future work could include implementing a Kalman Filter on the sensor data and determining whether it provides better performance. Other interesting avenues to pursue in future work include:

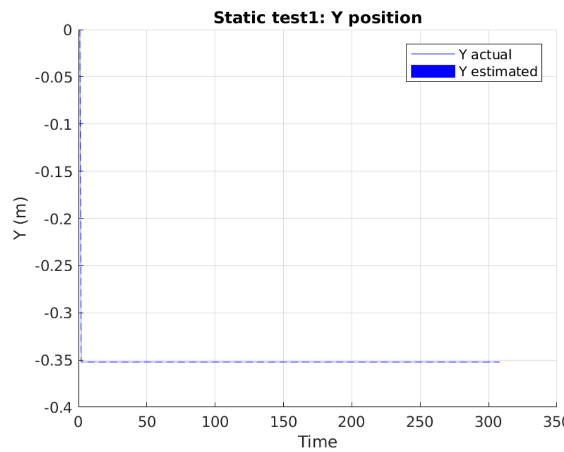
1) Minimization of the increase in localization inaccuracy associated with rapid location change.
2) Exploration performance characteristics when the swarm is associated with more complicated behaviors such as moving the GPS accessible robots and changing the behaviour of that robot to a rescuer robot depending on the closeness of that particular robot to the stranded robot.
3) Exploration of the effect of obstacles
4) Exploration of the effect more complicated arena geometry
   - ex: horseshoe shaped arenas, hourglass shaped arenas, mazes, etc
5) Exploration of the effect of how sharply the information availability drops off at the constriction is another area to investigate.
   - representation of decrease in GPS through simulating dropped packets or increased noise
6) Solving the simulator inconsistency problem and at the same time making the algorithm more directly applicable to real robots, through the introduction of the use of time-stamps, velocity calculations, and the use of more of the sensors.

Acknowledgment

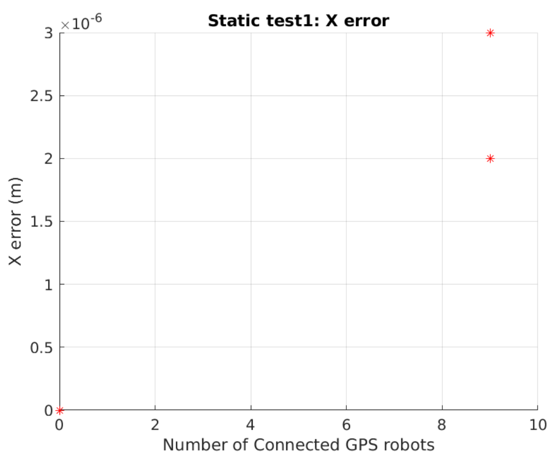
We would like to thank Professor Pinciroli and Jayam Patel for their support and help with both the ARGoS simulator and the Buzz programming language interface to ARGoS.



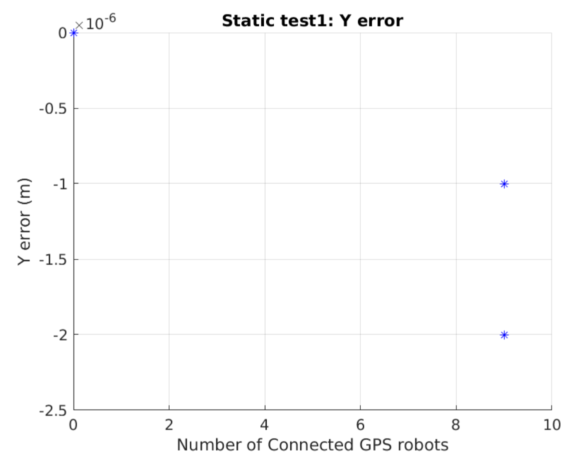
(a) X Position



(b) Y Position

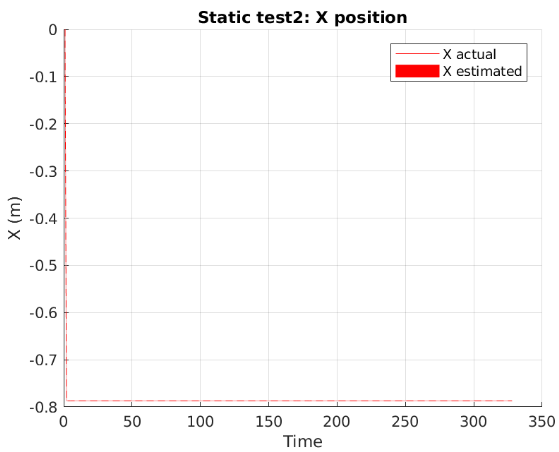


(c) X Position Error

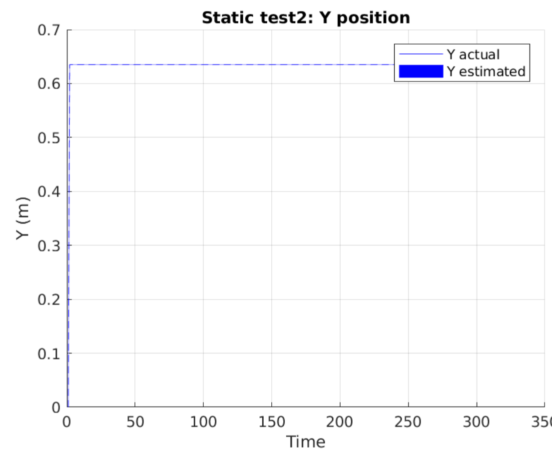


(d) Y Position Error

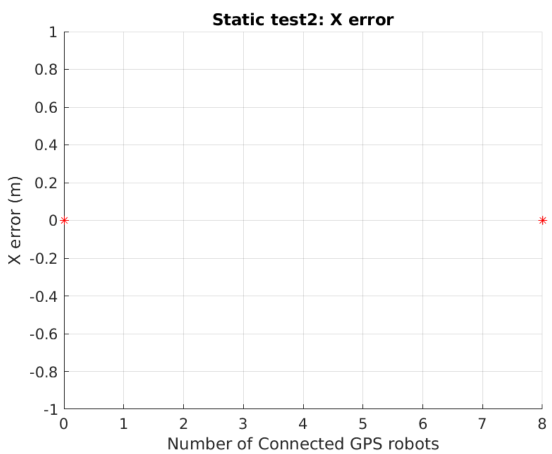
Fig. 6: Static Environment - Seed 1: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



(a) X Position



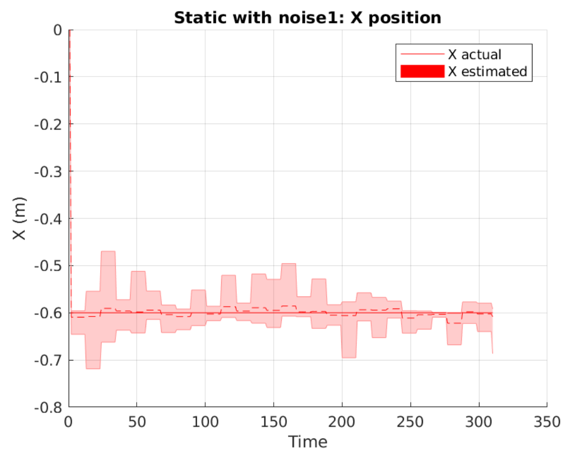
(b) Y Position



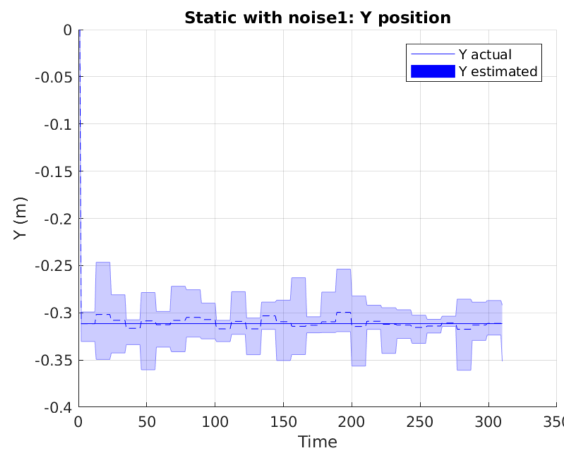
(c) X Position Error

(d) Y Position Error

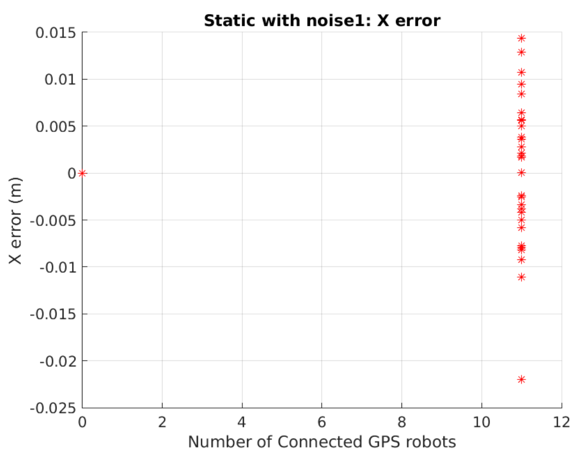
Fig. 7: Static Environment - Seed 2: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



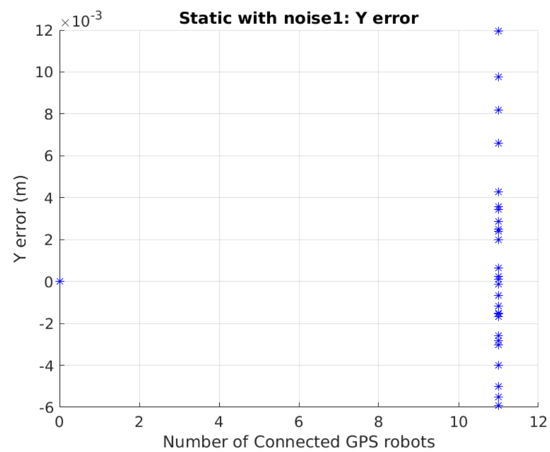
(a) X Position



(b) Y Position

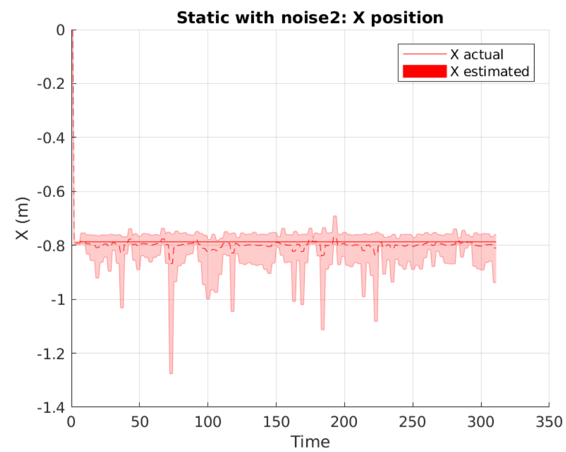


(c) X Position Error

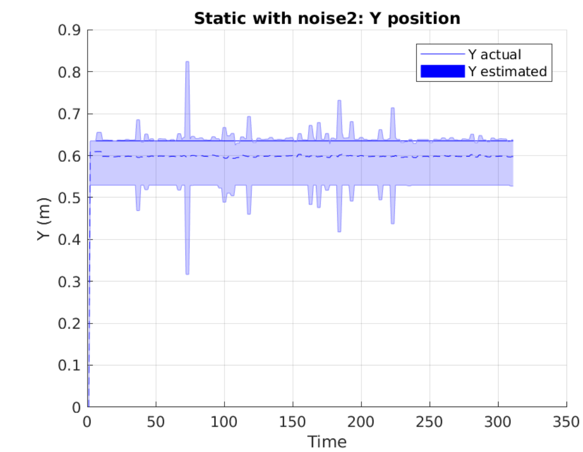


(d) Y Position Error

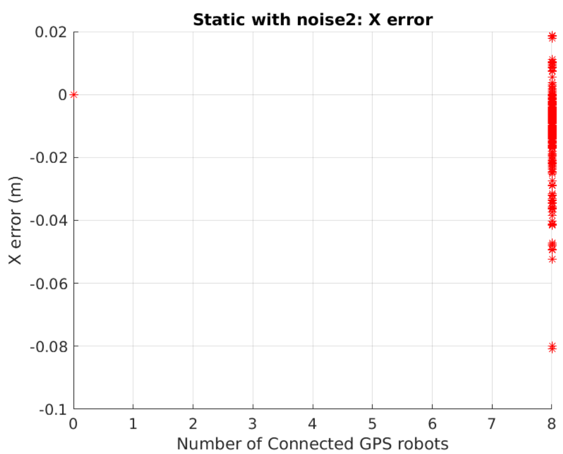
Fig. 8: Static Environment with noise - Seed 1: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



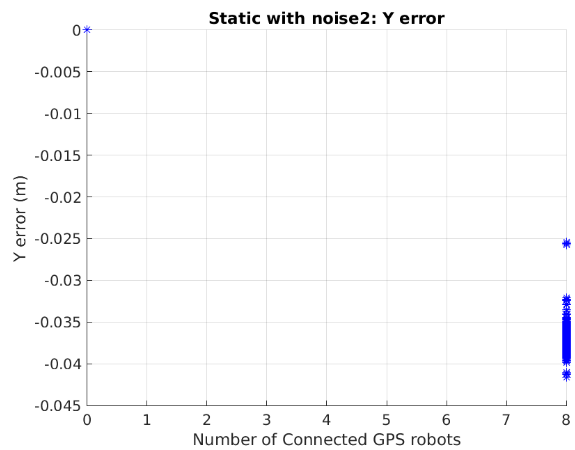
(a) X Position



(b) Y Position

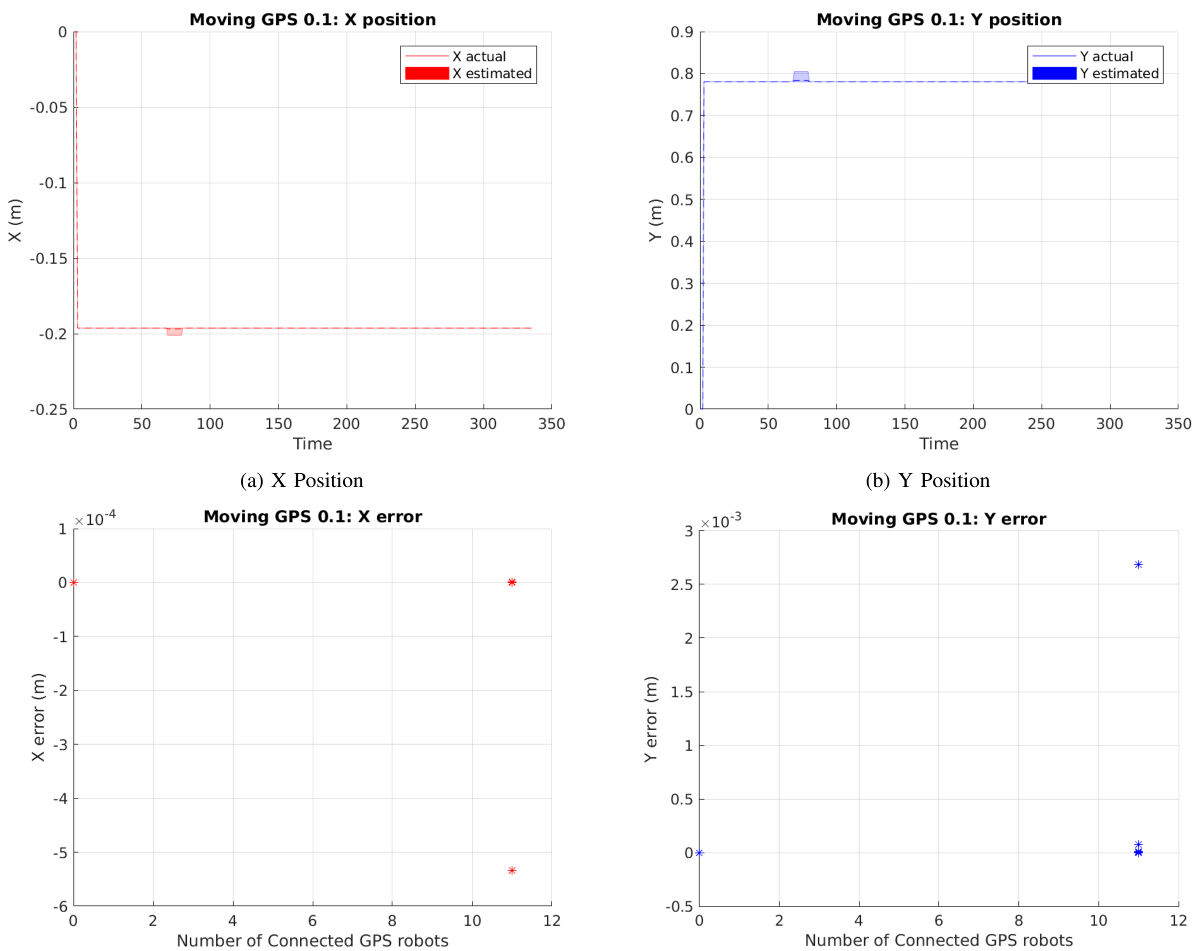


(c) X Position Error



(d) Y Position Error

Fig. 9: Static Environment with noise - Seed 2: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



(a) X Position

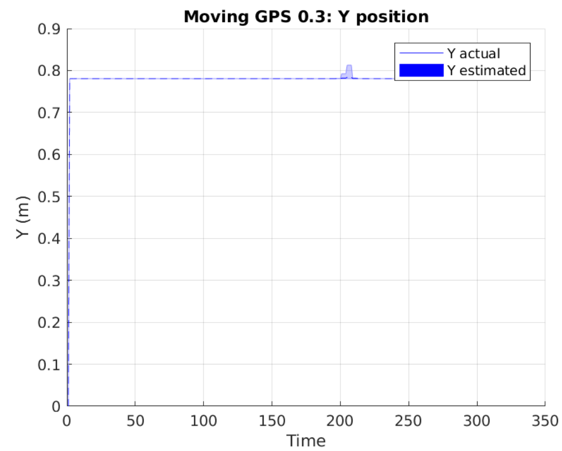
(b) Y Position

(c) X Position Error

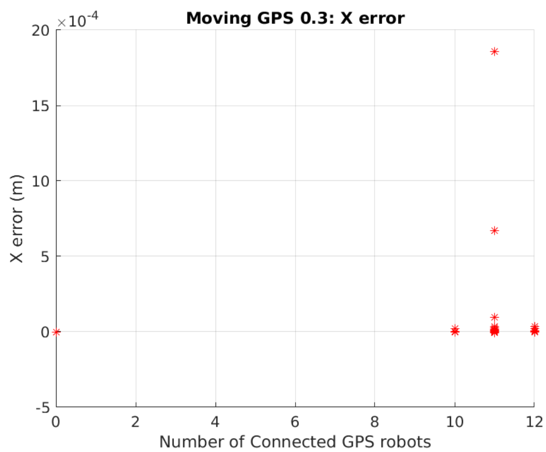
(d) Y Position Error

Fig. 10: 10% Moving GPS robots: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.

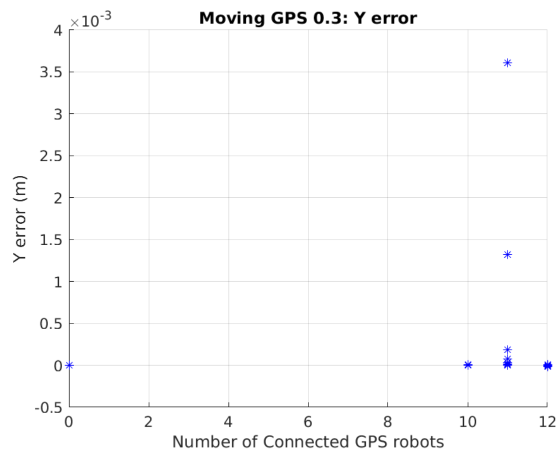
(a) X Position



(b) Y Position



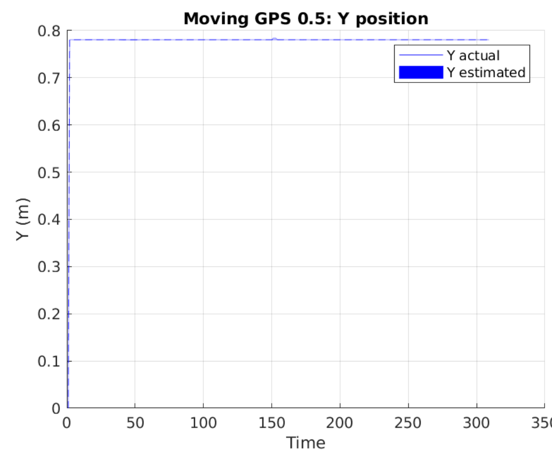
(c) X Position Error



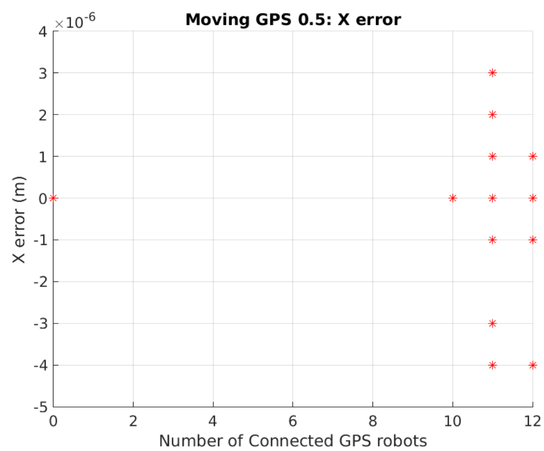
(d) Y Position Error

Fig. 11: 30% Moving GPS robots: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.

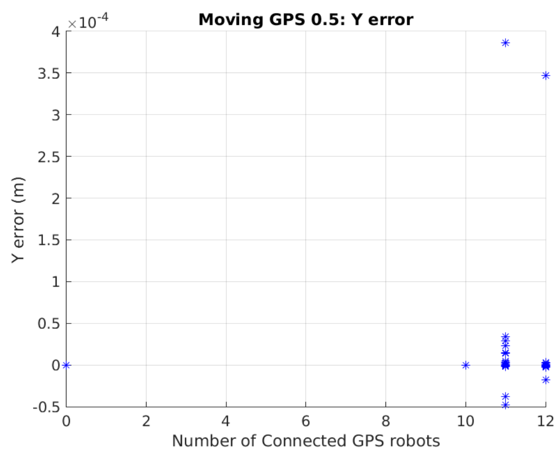
(a) X Position



(b) Y Position

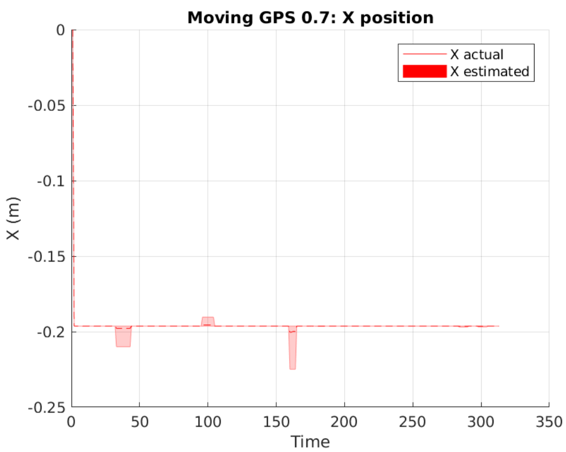


(c) X Position Error

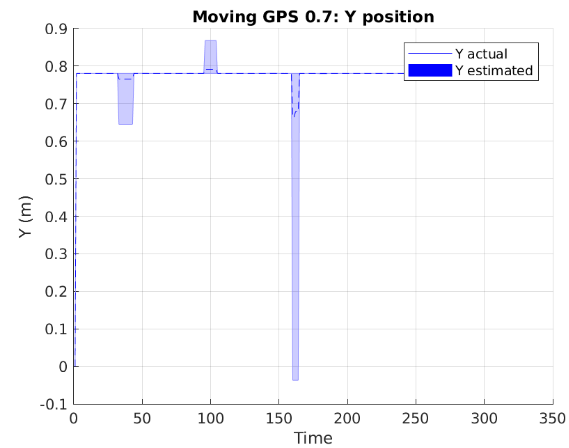


(d) Y Position Error

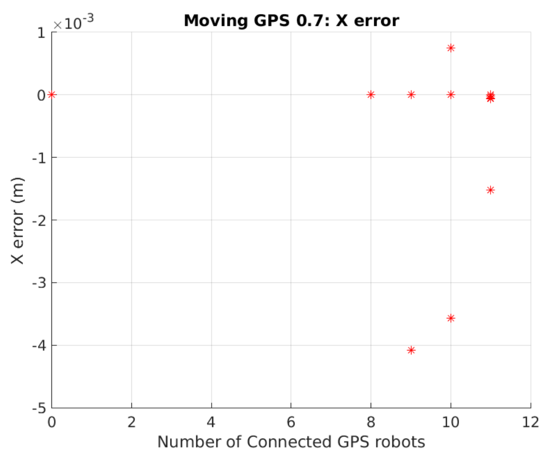
Fig. 12: 50% Moving GPS robots: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



(a) X Position



(b) Y Position

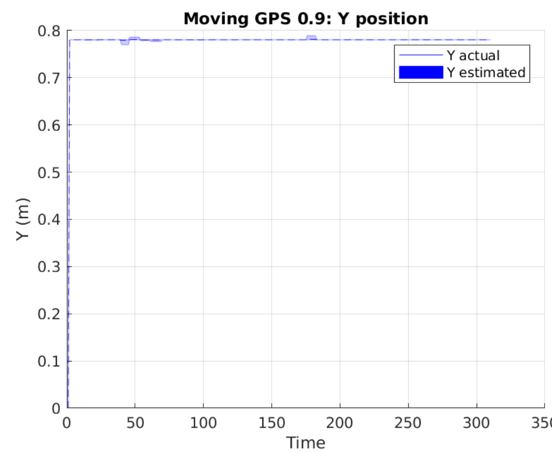


(c) X Position Error

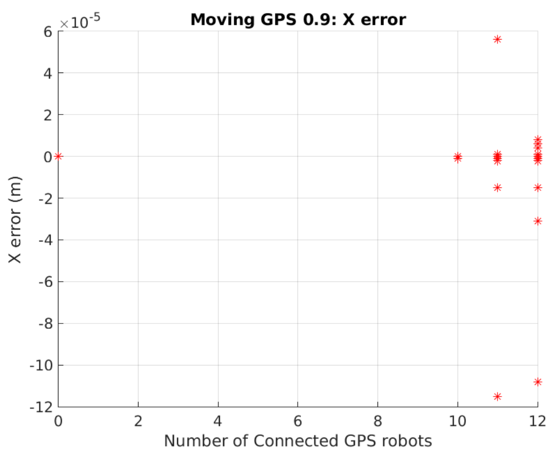
(d) Y Position Error

Fig. 13: 70% Moving GPS robots: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.

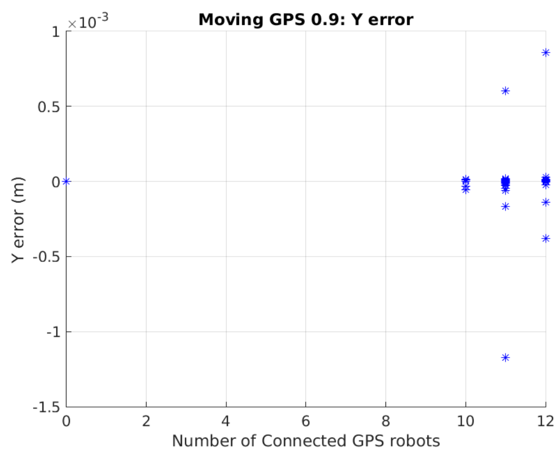
(a) X Position



(b) Y Position

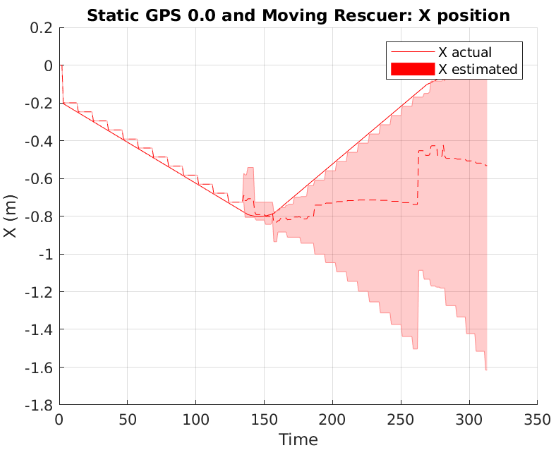


(c) X Position Error

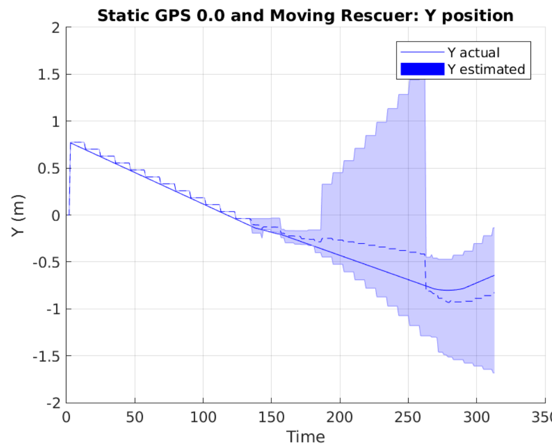


(d) Y Position Error

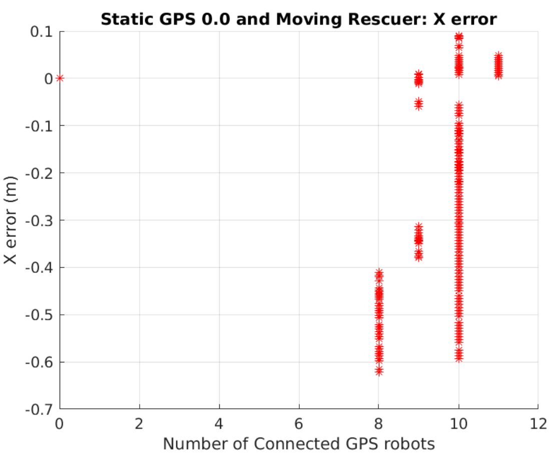
Fig. 14: 90% Moving GPS robots: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



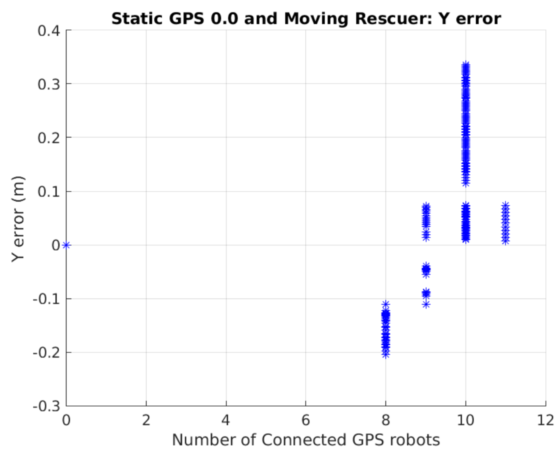
(a) X Position



(b) Y Position

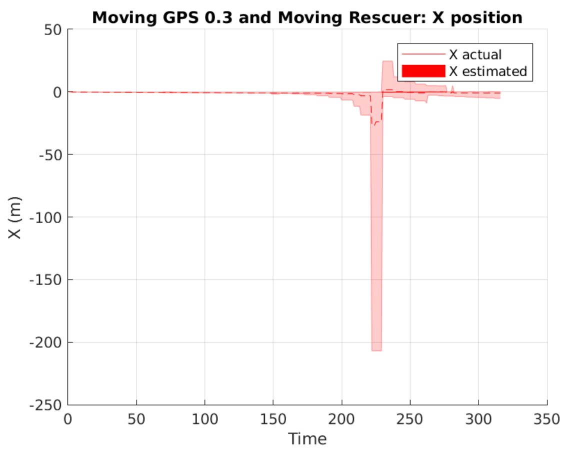


(c) X Position Error

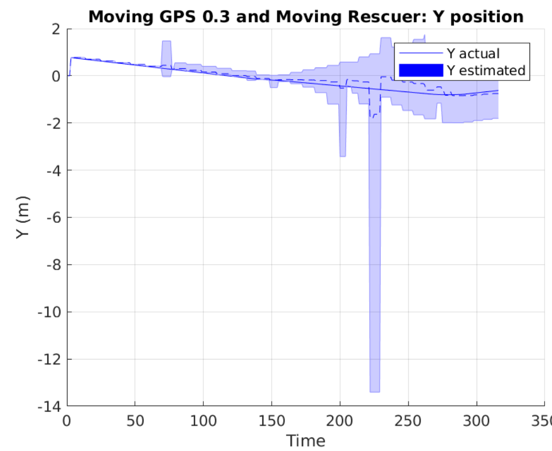


(d) Y Position Error

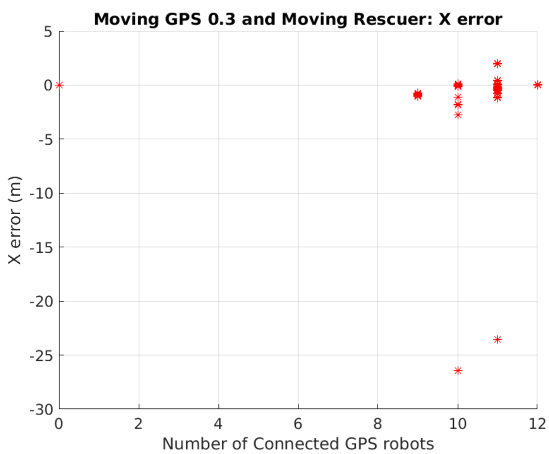
Fig. 15: Static GPS Robots and Moving Rescue Robot: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



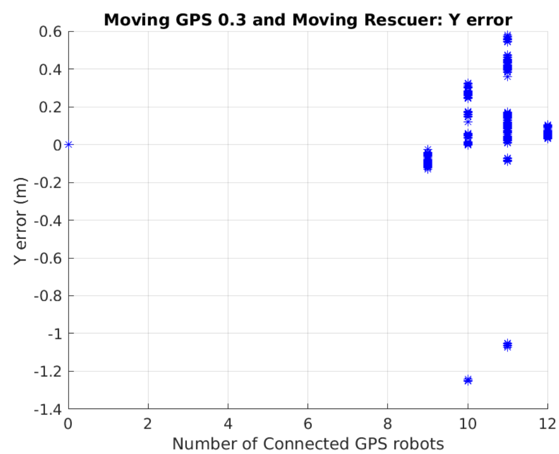
(a) X Position



(b) Y Position



(c) X Position Error

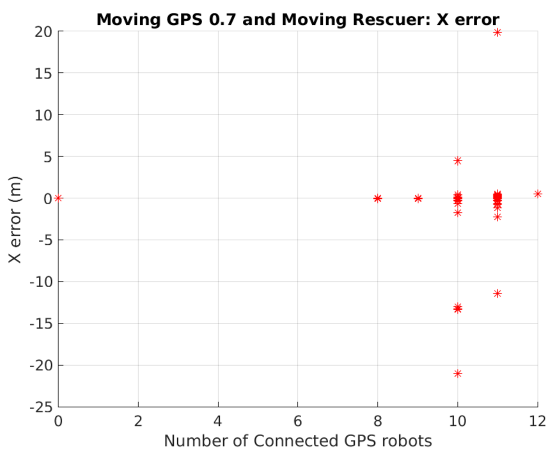


(d) Y Position Error

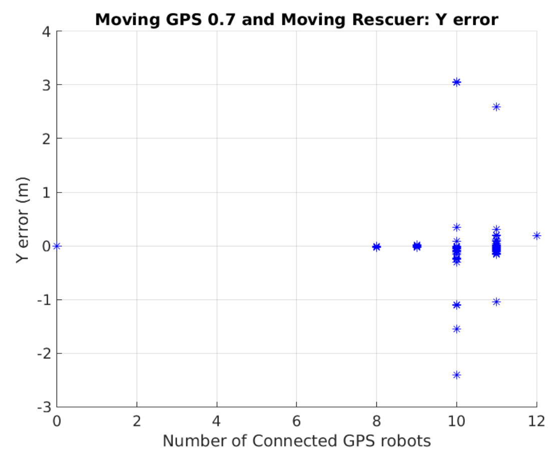
Fig. 16: 30% Moving GPS Robots and Moving Rescue Robot: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.

(a) X Position

(b) Y Position

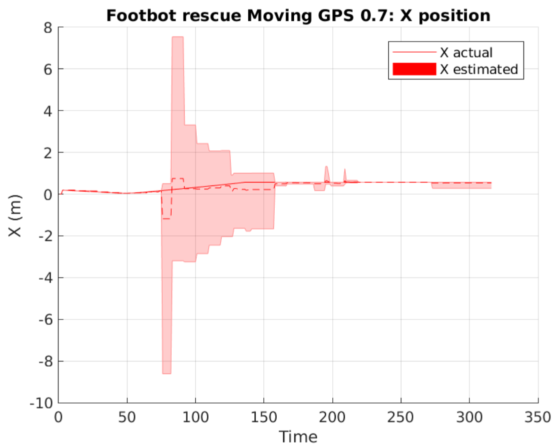


(c) X Position Error

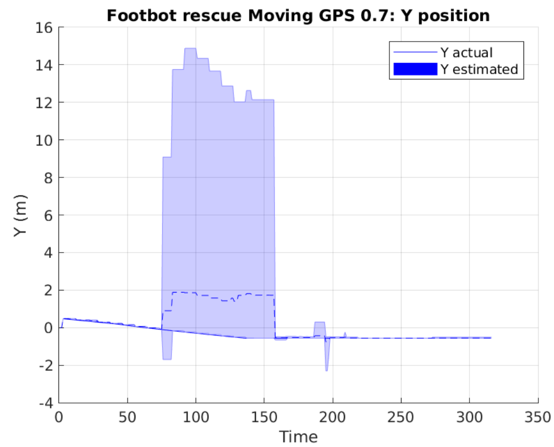


(d) Y Position Error

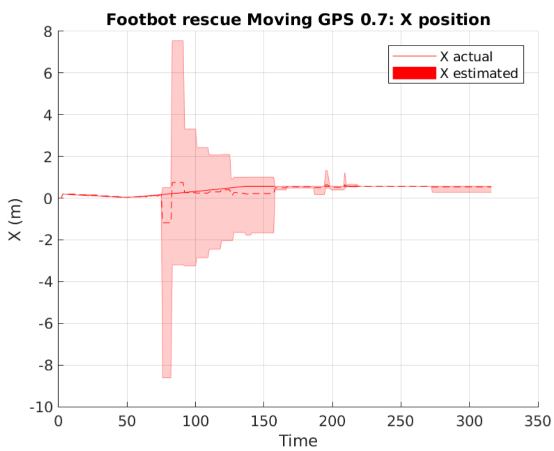
Fig. 17: 70% Moving GPS Robots and Moving Rescue Robot: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.



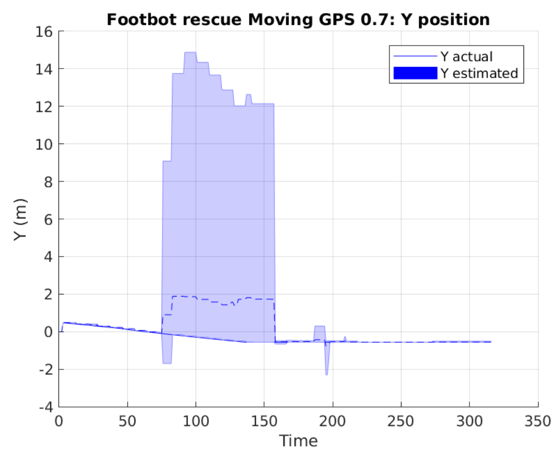
(a) X Position



(b) Y Position



(c) X Position Error



(d) Y Position Error

Fig. 18: 70% Moving GPS Foot-bots and Moving Rescue Robot: In a) and b), the bounds of the shaded regions are the calculated min and max possible location at that point in time. The dashed line is the estimated position and the solid line is the actual position.

References

[1] Nirupama Bulusu, John Heidemann, and Deborah Estrin. "GPS-less low-cost outdoor localization for very small devices". In: *IEEE personal communications* 7.5 (2000), pp. 28–34.

[2] *buzz kh4[The Buzz Language]: Buzz on the Khepera IV*. http://the.swarming.buzz/wiki/doku.php?id=buzz_kh4. Accessed: 2018-03-28.

[3] Titus Cieslewski, Siddharth Choudhary, and Davide Scaramuzza. "Data-Efficient Decentralized Visual SLAM". In: *arXiv preprint arXiv:1710.05772* (2017).

[4] Alejandro Cornejo and Radhika Nagpal. "Long-Lived Distributed Relative Localization of Robot Swarms". In: *arXiv preprint arXiv:1312.1915* (2013).

[5] Jing Dong et al. "Distributed real-time cooperative localization and mapping using an uncertainty-aware expectation maximization approach". In: *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE. 2015, pp. 5807–5814.

[6] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global positioning system: theory and practice*. Springer Science & Business Media, 2012.

[7] Tom Logsdon. *Understanding the Navstar: GPS, GIS, and IVHS*. Springer Science & Business Media, 2013.

[8] Joseph A. Rothermich, M. İhsan Ecemiş, and Paolo Gaudiano. "Distributed Localization and Mapping with a Robotic Swarm". In: *Proceedings of the 2004 International Conference on Swarm Robotics*. SAB'04. Santa Monica, CA: Springer-Verlag, 2005, pp. 58–69. ISBN: 3-540-24296-1, 978-3-540-24296-3. DOI: 10.1007/978-3-540-30552-1_6. URL: http://dx.doi.org/10.1007/978-3-540-30552-1_6.