Distributed Multi-Robot Multi-Target Tracking Using Heterogeneous Limited-Range Sensors

Jun Chen[1,*], Mohammed Abugurain[2], Philip Dames[3], Shinkyu Park[2], Fei Xie[1], Qi Mao[1]

[1]Nanjing Normal University, China
[1]Jiangsu Key Laboratory of 3D Printing Equipment and Manufacturing, China
[2]King Abdullah University of Science and Technology, Saudi Arabia
[3]Temple University, USA
*Corresponding Author Email: jun.chen@nnu.edu.cn

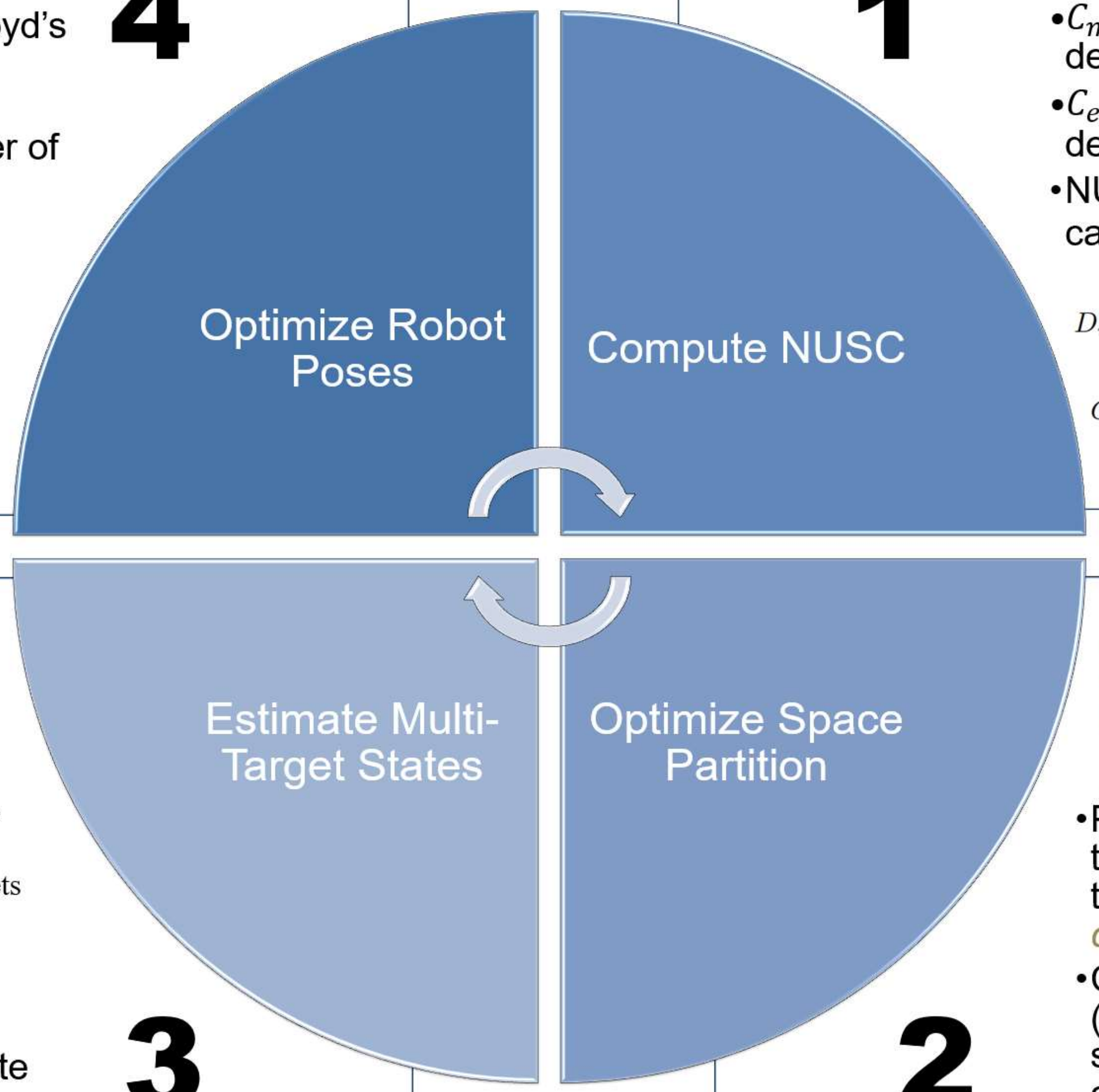
ICRA2024 YOKOHAMA | JAPAN

Abstract

This poster presents a cooperative multi-robot multi-target tracking framework aimed at enhancing the efficiency of the heterogeneous sensor network and, consequently, improving overall target tracking accuracy.

- The concept of normalized unused sensing capacity (NUSC) is introduced to quantify the information a sensor is currently gathering relative to its theoretical maximum. This measurement can be computed using entirely local information and is applicable to various sensor models, distinguishing it from previous literature on the subject.
- NUSC is then utilized to develop a distributed coverage control strategy for a heterogeneous sensor network, adaptively balancing the workload based on each sensor's current unused capacity.
- The algorithm is validated through a series of ROS and MATLAB simulations, demonstrating superior results compared to standard approaches that do not account for heterogeneity or current usage rates.

Methods



4

- Utilize Lloyd's algorithm to determine the control input u for each robot
- Set v as the weighting function in Lloyd's algorithm
- In this way, robots are driven to the weighted centroids of each cell, either of the power diagram or the CCVD

$$\mathcal{H}(Q,\mathcal{W}) = \sum_{i=1}^{n}\int_{\mathcal{W}_i} f(\|x-q_i\|)\phi(x)dx.$$

$$\frac{\partial \mathcal{H}_p(Q)}{\partial q_{\mathrm{cod},i}} = 2M_{\mathcal{W}_i}(q_{\mathrm{cod},i} - C_{\mathcal{W}_i})$$

$$M_{\mathcal{W}_i} = \int_{\mathcal{W}_i}\phi(x)\,dx. \quad C_{\mathcal{W}_i} = \frac{1}{M_{\mathcal{W}_i}}\int_{\mathcal{W}_i} x\phi(x)\,dx$$

$$u_i = \|C_{\mathcal{W}_i} - q_{\mathrm{cod},i}\|(dt)^{-1}$$

1

- D: The sensor's ability to detect information within its field of view
- C_{max}: Maximum possible number of target detections, depending on D
- C_{exp}: Expected number of target detections, depending on D
- NUSC $U = C_{max} - C_{exp}$: Quantifies current capacity for a sensor to track targets

$$D_i = \int_{F_i} p_d(x|q_i,\theta_i)\,dx \quad C_{\max,i} = \mu \max \int_{F_i} p_d(x|q_i,\theta_i)v(x)\,dx$$

$$C_{\exp,i} = \int_{F_i} p_d(x|q_i,\theta_i)p_t(x|F_i)\,dx \quad U_i = C_{\max,i} - C_{\exp,i}$$

3

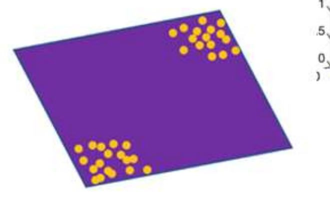


Fig. 2. True Targets

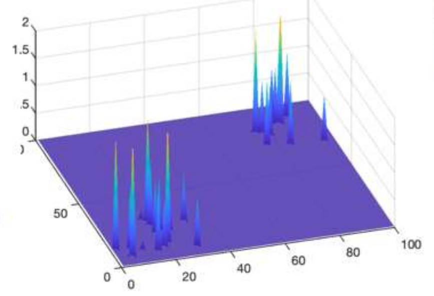


Fig. 3. Estimated Targets

- Utilize the distributed probability hypothesis density filter[1] or other distributed multi-robot multi-target Bayesian state estimators to estimate the multi-target state v

2

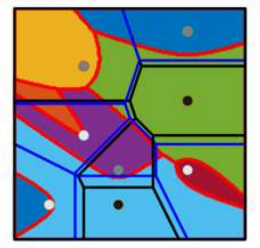


Fig. 1. Voronoi Diagram (Blank) vs Power Diagram (Blue) vs CCVD (Red). Darker dots weight more

- Power Diagram Implementation: Partition the space into power cells, set NUSCs as the power radius of each cell – *Soft constrain, but fast*
- Capacity constrained Voronoi Diagram (CCVD) Implementation: Partition the space using CCVD by our proposed distributed algorithm, set NUSCs as the weight of each cell – *Hard constrain, more computationally expensive*

Results

We conduct simulations using both ROS and MATLAB. In ROS simulations five TurtleBot 3 Burger robots with different sensors track 30 moving targets. In MATLAB simulations, we show the results of 60 robots track 30-50 moving targets. More results can be found in our preprint[2].

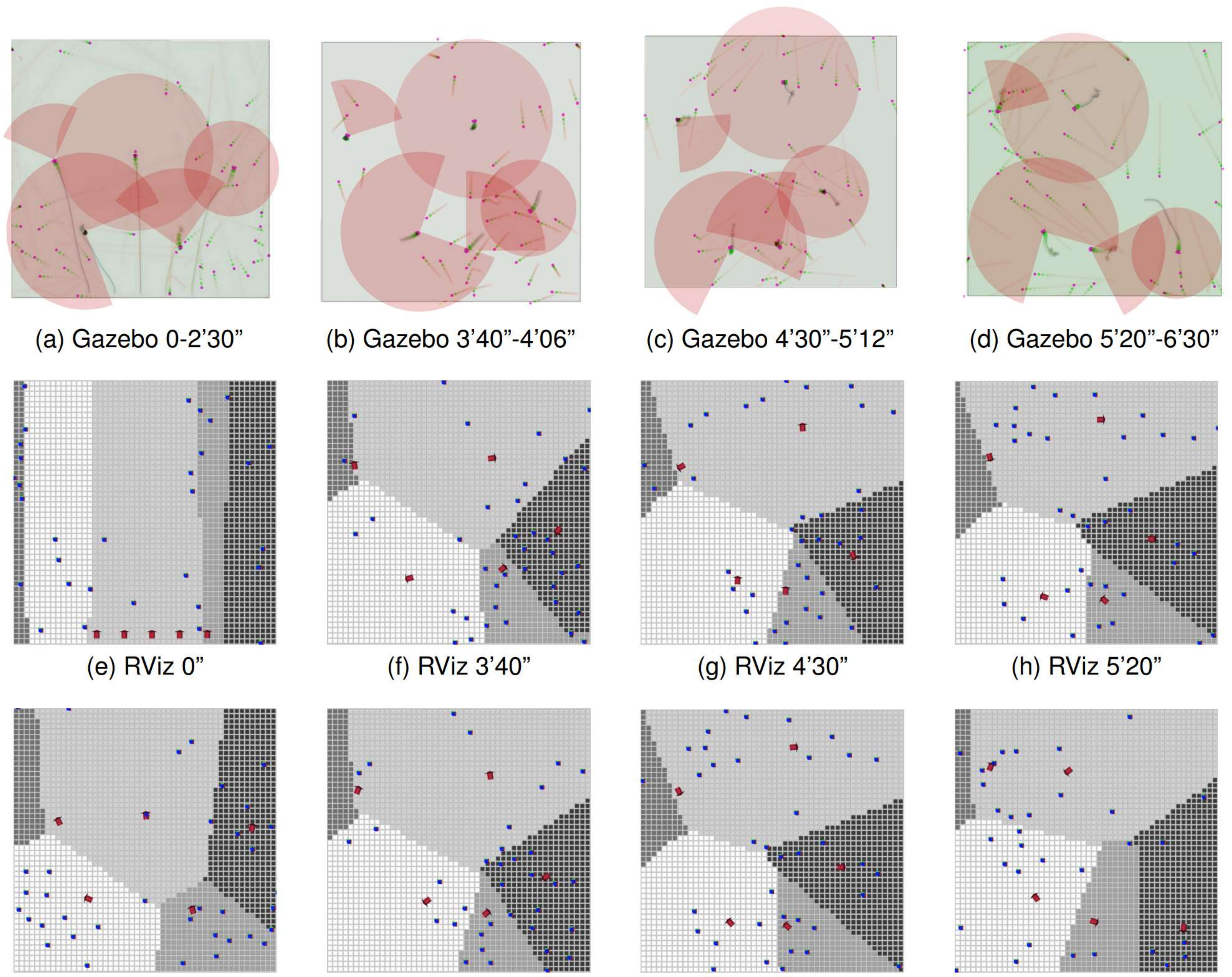


Fig. 4. Figures showing four clips during a single test using five TurtleBot3 robots, one of each type in Table I, to track forty moving targets, *i.e.*, from starting moment to 2 min 30 s (Figures 4a, 4e, 4i), from 3 min 40 s to 4 min 6 s (Figures 4b, 4f, 4g), from 4 min 30 s to 5 min 12 s (Figures 4c, 4g, 4h), and from 5 min 20 s to 6 min 30 s (Figures 4d, 4h, 4l), respectively. Figures on the first row are screenshot overlays of Gazebo GUI, showing the top-view of five robots with FoVs, marked in red, and targets marked in blue dots. Robot trajectories and target traces are also shown. Figures on the second and the third row show screenshot of RViz GUI at the beginning and the end of each clip, respectively. Red arrows show robot locations and orientations. Blue dots show target locations. Regions in different shades of grey show the assigned cells for each robot.

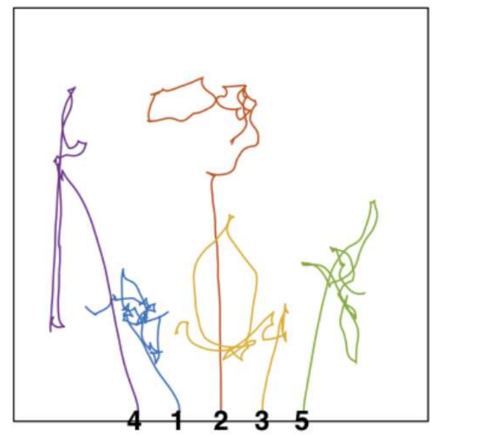


Fig. 5. Figure showing trajectories over 12 min 30 s testing time of 5 robots respectively in the squared open task space.The numbers indicate the IDs of the robots.

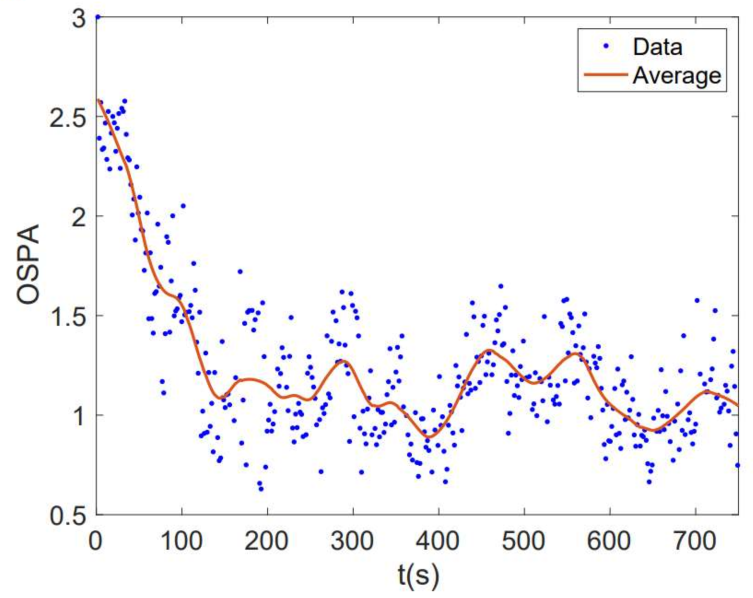


Fig. 6. Figure showing OSPA error at each discrete time step and its moving average during the entire simulation time.

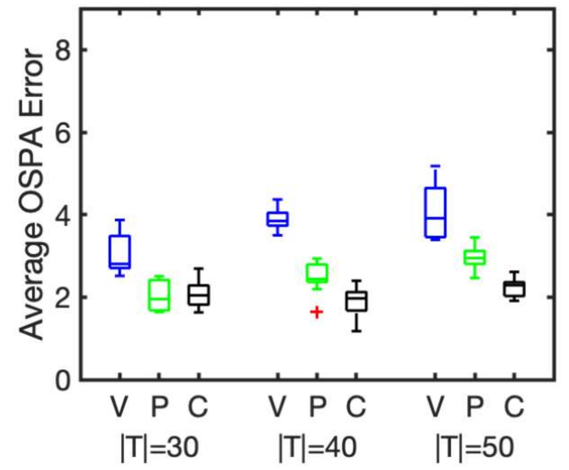


Fig. 7. The average OSPA error obtained from MATLAB tests. Blue, green, and red boxplots show results of using Voronoi diagram (V), power diagram (P), and CCVD (C) respectively

Conclusions

- We propose a distributed coverage control scheme for heterogeneous mobile robots with onboard sensors to track an unknown and time-varying number of targets.
- This novel strategy allows sensors to have arbitrary sensing models (with limited field of views) and dynamically optimizes the workload for each individual.
- Simulation results show the convergence of our proposed method in target tracking scenarios and indicates that our method yields better and more reliable tracking performance compared to a standard Voronoi diagram that does not account for heterogeneity.

References

[1] Dames, P. M. (2020). Distributed multi-target search and tracking using the PHD filter. Autonomous robots, 44(3), 673-689.

[2] Chen, J., Abugurain, M., Dames, P., & Park, S. (2023). Distributed Multi-Robot Multi-Target Tracking Using Heterogeneous Limited-Range Sensors. arXiv preprint arXiv:2311.01707.