MengeROS: A Crowd Simulation Tool for Autonomous Robot Navigation

Anoop Aroor[1], Susan L. Epstein[1,2] and Raj Korpan[1]
Department of Computer Science
The Graduate Center[1] and Hunter College[2] of The City University of New York
New York, NY 10065
aaroor@gradcenter.cuny.edu, susan.epstein@hunter.cuny.edu, rkorpan@gradcenter.cuny.edu

Abstract

While effective navigation in large, crowded environments is essential for an autonomous robot, preliminary testing to support it requires simulation across a broad range of crowd scenarios. Most available simulation tools provide either realistic crowds without robots or realistic robots without realistic crowds. This paper introduces MengeROS, a flexible 2-dimensional simulator that realistically integrates multiple robots and crowds. MengeROS provides a broad range of settings in which to test the capabilities and performance of navigation algorithms designed for large crowded environments.

Robots are increasingly deployed in crowded indoor environments, such as museums, shopping malls, and conference centers (Tsui et al. 2011). It is challenging, however, to design and execute appropriate, large-scale, real-world testing for robots across the broad range of crowd conditions that arise there. Because each hypothetical instance defines a different test case, a flexible, accurate crowd simulator is essential before a robot is deployed in a crowded environment. Such a simulator can also support the evaluation of different navigation algorithms under comparable crowd conditions. This paper introduces a novel tool, *MengeROS*, that integrates a flexible, open-source crowd simulator called *Menge* (Curtis, Best, and Manocha 2016) with *ROS*, the standard operating system for robots that navigate.

Specialized robot simulators do not simulate realistic crowds (e.g., Gazebo (Koenig and Howard 2004) or Stage (Gerkey, Vaughan, and Howard 2003)). Moreover, most crowd simulators do not simulate robots (e.g., PedSim[1], OpenSteer (Reynolds 1999), Menge, and Continuum (Treuille, Cooper, and Popović 2006)). The two exceptions are PedSim_ros[2] and a ROS version of Continuum. The former, however, is restricted to one collision avoidance model, and the latter is not freely available for research.

MengeROS is an open-source two-dimensional crowd simulator that facilitates research on multi-robot path planning in large crowded environments. MengeROS inherits Menge's ability to simulate a broad variety of human crowd behaviors, as described in the next section. MengeROS integrates the robot into the crowd, so that the crowd must avoid the robot through its Menge-specified behavior while, in turn, the robot's ROS controller must avoid the simulated individuals in the crowd. To the best of our knowledge, MengeROS is the only open-source simulator that supports the movement of multiple robots through a broad range of crowd scenarios.

The Menge Crowd Simulator

Within a map of the environment's static elements (e.g., walls and furniture), a *location* in Menge is either a pair (*x,y*) of coordinates or a delineated area (e.g., the kitchen). Each *pedestrian* (simulated person) in a Menge crowd begins at some initial location, selects a *target sequence* of locations to visit with a finite state machine (*goal selection*). and moves toward its next target's location. To reach its current target, each pedestrian uses the same *plan computation* strategy to select its next location. Menge implements two plan computation methods, A* and potential fields. Both generate and assign a *velocity vector* (direction and distance) to each pedestrian. A* pursues an optimal shortest path; potential fields uses an attractor mechanism.

Pedestrians must avoid collisions both with the static features in the environment and with one another. In Menge, each pedestrian uses the same *plan adaptation* method, a collision-avoidance strategy that adjusts its intended velocity vector. Menge implements six plan adaptation methods. Four are based on the *social force model* (Helbing and Molnar 1995), which revises each pedestrian's velocity vector based on the attractive or repulsive forces of nearby objects and pedestrians. The other two methods, ORCA (van den Berg et al. 2011) and PedVO (Curtis and Manocha 2012), are based on velocity obstacles. The velocity obstacle (*VO*) of a pedestrian is the set of all velocity vectors that will produce a collision. Collision-free motion requires that every pedestrian have a velocity vector outside its VO. To prevent livelock and find an optimal solution, ORCA shares this responsibility equally among all pedestrians. PedVO adapts ORCA to behave more similarly to people.

A *crowd scenario* in Menge specifies the number of pedestrians with their initial locations and target selection mechanisms, plus a uniform plan computation and a uniform plan adaptation strategy for all pedestrians. Users can select from among pre-coded options for goal selection, plan computation, and plan adaptation, or implement their own.

[1] http://pedsim.silmaril.org

[2] https://github.com/srl-freiburg/pedsim_ros

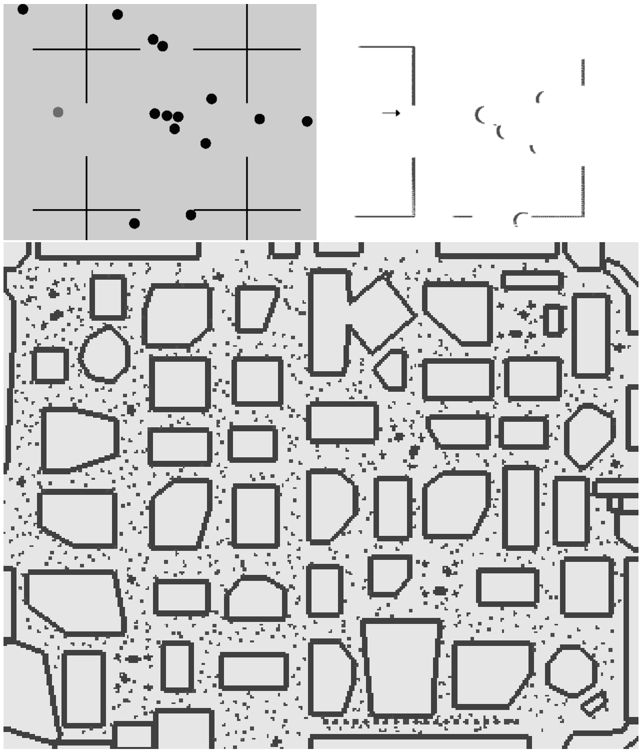

Figure 1: (top left) Aerial view of a simple world with a robot and 14 pedestrians. (top right) Robot range scanner readings (bottom) A trade show world with 1000 pedestrians and 20 robots in a row at the lower right

Menge's ability to simulate many different crowd scenarios is a significant improvement over earlier crowd simulators, which hardcoded a single approach. Nonetheless, Menge is not available through ROS and does not simulate robots.

MengeROS

A typical ROS-based robot navigation framework uses a *simulator node*. This node accepts as input a velocity command in a ROS-specified format, and returns simulated sensor readings (e.g., laser range scans) at a specified frequency. To determine the robot's motion, a controller node generates velocity commands based on the most recent sensor reading it has received from the simulator node.

MengeROS simulates both robots and pedestrians in a single node. MengeROS controls pedestrian behavior just as Menge does. It also allows multiple robots to be introduced, each with its own external controller. A robot in MengeROS executes the velocity commands received from its external ROS controller, similar to the way other robot simulators (e.g., Gazebo and Stage) interface with ROS. Pedestrians avoid the robot and one another with the plan adaptation option specified in the Menge control files. A robot, however, is completely dependent on the external commands from its own controller for collision avoidance.

MengeROS can simulate a laser scanner mounted on a robot. At the top of Figure 1 are two aerial views of a simple world. On the left is the ground truth, with a gray robot and 14 black pedestrians. On the right is the robot's view when it is located at the arrow's tail and oriented toward its head. Distances to obstacles are reported by a simulated laser with a (configurable) 220° field of view with maximum (configurable) range of 25 meters. MengeROS returns the positions of all pedestrians and robots in ROS-compatible format, for use by all other ROS nodes.

Results and Discussion

MengeROS readily simulates large crowds, including 1000 pedestrians that move simultaneously in Menge's complex trade show environment, shown at the bottom of Figure 1. Each decision cycle computes and assigns a new velocity vector to every pedestrian. On an 8-core, 1.2 GHz workstation, 100 decision cycles without robots average 51ms each. Because each robot's range sensors must be processed separately, more robots slow performance. This slowdown appears to be linear in the number of robots. Average decision cycle times with 5, 10, 15, and 20 robots were 437ms, 824ms, 1204ms, and 1568ms, respectively.

Recent work to improve navigation in a crowded environment used MengeROS to simulate robot movement through as many as 90 pedestrians (Aroor and Epstein 2017). In this work, a robot learns a *crowd density map* for pedestrian movement from simulated 2-D laser scan data. Multiple simulations with MengeROS also allowed for easy comparison of traditional A* with CSA*, a new, crowd-sensitive planner that improves navigation performance.

Code, documentation, and video examples for MengeROS are available on GitHub.[3] This version assumes circular robots with a configurable radius, and noise-free laser scans and movements controlled by velocity vectors. Future work will introduce noise, simulate robots of different shapes, process data from sensors other than range finders (e.g., odometry and cameras), and refine the code to reduce decision cycle time.

Human behavior with robots is based on demographics (Brscić et al. 2015; May et al. 2017) and how each individual perceives, experiences, and interacts with the robot (Nomura et al. 2008; Walters et al. 2005; Takayama and Pantofaru 2009; Mumm and Mutlu 2011; Butler and Agah 2001; Dautenhahn et al. 2006; Rios-Martinez, Spalanzani, and Laugier 2015). Individualized pedestrian reactions and sophisticated, full-body humanoid robots with gestures and expressions, while possible on this scale, are beyond the scope of our current work. MengeROS focuses only on safe, nonthreatening navigation, rather than interaction.

In summary, robots in crowded indoor environments experience new challenges as their navigation algorithms confront dynamic obstacles. Research costs to develop algorithms in realistic scenarios can be significantly reduced by simulation. MengeROS is an efficient, flexible, and extensible new tool for such work. It builds upon the Menge crowd simulator, and allows robotics researchers to test their algorithms in realistic crowds before deployment.

[3] https://github.com/ml-lab-cuny/menge_ros/

References

Aroor, A., and Epstein, S. L. 2017. Toward Crowd-Sensitive Path Planning. In *AAAI 2017 Fall Symposium on Human-Agent Groups*.

Brscić, D.; Kidokoro, H.; Suehiro, Y.; and Kanda, T. 2015. Escaping from Children's Abuse of Social Robots. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, 59–66. ACM.

Butler, J. T., and Agah, A. 2001. Psychological Effects of Behavior Patterns of a Mobile Personal Robot. *Autonomous Robots* 10(2):185–202.

Curtis, S., and Manocha, D. 2012. Pedestrian simulation using geometric reasoning in velocity space. In Weidmann, U.; Kirsch, U.; and Schreckenberg, M., eds., *Pedestrian and Evacuation Dynamics 2012*. Cham: Springer International Publishing. 875–890.

Curtis, S.; Best, A.; and Manocha, D. 2016. Menge: A Modular Framework for Simulating Crowd Movement. *Collective Dynamics* 1:1–40.

Dautenhahn, K.; Walters, M.; Woods, S.; Koay, K. L.; Nehaniv, C. L.; Sisbot, A.; Alami, R.; and Siméon, T. 2006. How May I Serve You?: A Robot Companion Approaching a Seated Person in a Helping Context. In *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, 172–179. ACM.

Gerkey, B.; Vaughan, R. T.; and Howard, A. 2003. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In *Proceedings of the 11th International Conference on Advanced Robotics*, volume 1, 317–323.

Helbing, D., and Molnar, P. 1995. Social Force Model for Pedestrian Dynamics. *Physical Review E* 51(5):4282.

Koenig, N., and Howard, A. 2004. Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004*, volume 3, 2149–2154.

May, D. C.; Holler, K. J.; Bethel, C. L.; Strawderman, L.; Carruth, D. W.; and Usher, J. M. 2017. Survey of Factors for the Prediction of Human Comfort with a Non-anthropomorphic Robot in Public Spaces. *International Journal of Social Robotics* 9(2):165–180.

Mumm, J., and Mutlu, B. 2011. Human-Robot Proxemics: Physical and Psychological Distancing in Human-robot Interaction. In *Proceedings of the 6th International Conference on Human-Robot Interaction*, 331–338. ACM.

Nomura, T.; Kanda, T.; Suzuki, T.; and Kato, K. 2008. Prediction of Human Behavior in Human-Robot Interaction Using Psychological Scales for Anxiety and Negative Attitudes Toward Robots. *IEEE Transactions on Robotics* 24(2):442–451.

Reynolds, C. W. 1999. Steering Behaviors for Autonomous Characters. In *Game Developers Conference*, 763–782.

Rios-Martinez, J.; Spalanzani, A.; and Laugier, C. 2015. From Proxemics Theory to Socially-Aware Navigation: A Survey. *International Journal of Social Robotics* 7(2):137–153.

Takayama, L., and Pantofaru, C. 2009. Influences on Proxemic Behaviors in Human-Robot Interaction. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009.*, 5495–5502. IEEE.

Treuille, A.; Cooper, S.; and Popović, Z. 2006. Continuum Crowds. *ACM Transactions on Graphics (TOG)* 25(3):1160–1168.

Tsui, K. M.; Desai, M.; Yanco, H. A.; and Uhlik, C. 2011. Exploring Use Cases for Telepresence Robots. In *Proceedings of the 6th International Conference on Human-Robot Interaction*, HRI '11, 11–18.

van den Berg, J.; Guy, S.; Lin, M.; and Manocha, D. 2011. Reciprocal n-body collision avoidance. *Robotics Research* 3–19.

Walters, M. L.; Dautenhahn, K.; Te Boekhorst, R.; Koay, K. L.; Kaouri, C.; Woods, S.; Nehaniv, C.; Lee, D.; and Werry, I. 2005. The Influence of Subjects' Personality Traits on Personal Spatial Zones in a Human-Robot Interaction Experiment. In *IEEE International Workshop on Robot and Human Interactive Communication, 2005. ROMAN 2005.*, 347–352. IEEE.