An Intelligent Human Avatar to Debug and Challenge Human-aware Robot Navigation Systems

Anthony Favier
LAAS-CNRS
*ANITI**
Toulouse, France
anthony.favier@laas.fr

Phani Teja Singamaneni
LAAS-CNRS
Toulouse, France
ptsingaman@laas.fr

Rachid Alami
LAAS-CNRS
*ANITI**
Toulouse, France
rachid.alami@laas.fr

2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI) | 978-1-6654-0731-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/HRI53351.2022.9889651

***Abstract*—Experimenting, testing, and debugging robot social navigation systems is a challenging task. While simulation is generally well suited for a first level of debugging and evaluation of robotics controllers and planners, the social navigation field lacks satisfactory simulators of humans which act, react and interact rationally and naturally. To facilitate the development of human-aware navigation systems, we propose a system to simulate an autonomous human agent that is both reactive and rational, specifically designed to act and interact with a robot for navigation problems and potential conflicts. Besides, it also provides some metrics to partially evaluate such interactions and data logs for further analysis. We show the limitations of over-used reactive-only approaches. Then, thanks to two different human-aware navigation planners, we show how our system can help answer the lack of intelligent human avatars for tuning and debugging social navigation systems before their final evaluation with real humans.**

***Index Terms*—Human-Robot Interaction; Human Simulation; Social Navigation**

I. Introduction

The development of robots that interact, assist, or work among humans is an active field of research. However, people working in this field struggle to test and evaluate their systems. Real-life tests are slow, hardly repeatable, and expensive. Such tests are mandatory to verify the final system. However, the system needs to also be tested before reaching maturity, and real-life tests become very tiresome for debugging. Simulation could be an efficient way to achieve these preliminary tests. Yet, simulating realistic interactions is difficult which makes simulation unreliable and limited. This is why there is a need for an "intelligent artificial human" which would challenge the robot's interactive and decisional abilities.

The social robot navigation field faces the same issues. Currently, researchers conduct their preliminary tests in simulation using reactive models for the human agents. Such models, like social force or optimal reciprocal collision avoidance (ORCA), are easy to use and very efficient for crowd simulation. Nevertheless, as individuals, such agents tend to act unnaturally and to even completely fail in solving conflicts in intricate scenarios.

Our contribution is an architecture to simulate an intelligent autonomous human agent that will challenge social navigation of robots in intricate scenarios. Our avatar is able, to a certain extent, to act, react to robot behavior and interact with the robot. First, it exhibits a goal-oriented behavior and adapts to changes in the environment (path blocked, ..) with goal persistence. Also, we can add some usual human attitudes or behaviors like a change of mind: stop the current goal and start achieving a new one. Finally, we can tune the human reaction to some situations: the cautious human who stops far from a robot to facilitate its actions, or, on the contrary, a human who is in a hurry or who assumes that the robot should give priority to her/him and take the load of solving conflicts. We also propose a way to visualize logged metrics and execution data through plots and graphs to help evaluate the interaction.

We use here below the term 'rational' to refer to a notion close to Goal Reasoning [1], [2], i.e. the ability of autonomous agents which can dynamically reason about and adjust their goals. Doing so, enables agents to adapt intelligently to changing conditions and unexpected events, allowing them to address a wider variety of complex problems.

The paper is structured as follows. Section II briefly discusses related work. Section III presents the current implementation of the system and its functionalities. Then, section IV exposes the results we obtained using different social navigation robot systems. Finally, section V presents conclusions and future work.

II. Related Works

Human-aware navigation (or social navigation) has been a topic of research for a long time [3], [4], but, recently it is gaining more attention due to the increasing number of robots navigating in the human environments. Numerous approaches were proposed [5]–[12] to safely navigate the robot around the humans. Although most of these methods were largely tested using simulation during their development, the model of the human remained very simple and some times unrealistic.

Simulating perfectly natural and realistic human behavior is currently impossible. But with some efforts and depending on the context, we can make intelligent enough agents able to challenge the robotics systems. Many social navigation works focus on crowd navigation and such contexts can now be simulated efficiently with reactive models. Works

*This work has been partially funded by the Agence Nationale de la Recherche through the ANITI ANR-19-PI3A-0004 grant

like MengeROS [13] or PedSim_ROS[1] use the social force model [14] and offer a scalable and efficient way to simulate crowds of human agents. Other crowd simulators using the ORCA method can be found like the work in [15]. Reactive-only methods are clearly useful for crowds but provide very limited possibilities when trying to simulate individual agents, particularly in intricate scenarios and narrow environments. Simulating more intelligent human agents is a novel field and only a few very recent contributions address this topic. VirtualHome [16] appears as very interesting but interactions between agents are still limited. Another recent work [17] presents a learning-based local planner to generate a natural pedestrian motion. This ongoing work shows an interesting navigation behavior like waiting and letting the other agent pass embedded in iGibson [18]. However, it is not clear if it is possible to attribute dynamically new goals to the agents and if they can solve more intricate navigation conflicts. Also, as of today, it is not publicly available.

III. Presentation of the System

In this section, we start by presenting the main functionalities of the system which generate rational behavior. It is followed by details on the usability of the system. And finally, a description of the logs and metrics is given. The whole system is open-source and already publicly available[2].

A. Major functionalities

The system has a goal management process that includes choosing a goal (or waiting for one from the human operator), generating a plan to achieve the given goal, and supervising the execution of the plan, step by step, until the goal is reached. The system is aware of its own goal and it can reason and be persistent about the goal. The plan execution can be suspended and later resumed, which is useful to resolve conflicts or for specific reactions of the avatar.

During the execution of navigation actions, the system can detect and handle navigation conflicts. Currently, the kind of navigation conflict handled is path blockage (e.g. another agent standing in a doorway). To detect such conflict we periodically compute a path to the current goal position using Dijkstra's algorithm. Then, by tracking the length of the computed path, a significant increase of the path length or no path at all means that another agent is blocking either the shortest or the only possible way to the goal. When such situations are detected, the conflict is resolved by suspending the plan execution to perform an approach in order to show the agent's intention. Eventually, once close enough and still blocked, the agent stops and actively waits for the path to be cleared.

As navigation planner, the system uses a publicly available human-aware navigation planner called CoHAN [19]. It has been slightly modified to remove some conflicting features. Thus, we benefit from the high-level decision-making of the system and the enhanced local navigation with trajectory

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

[2]https://github.com/AnthonyFavier/InHuS_Social_Navigation

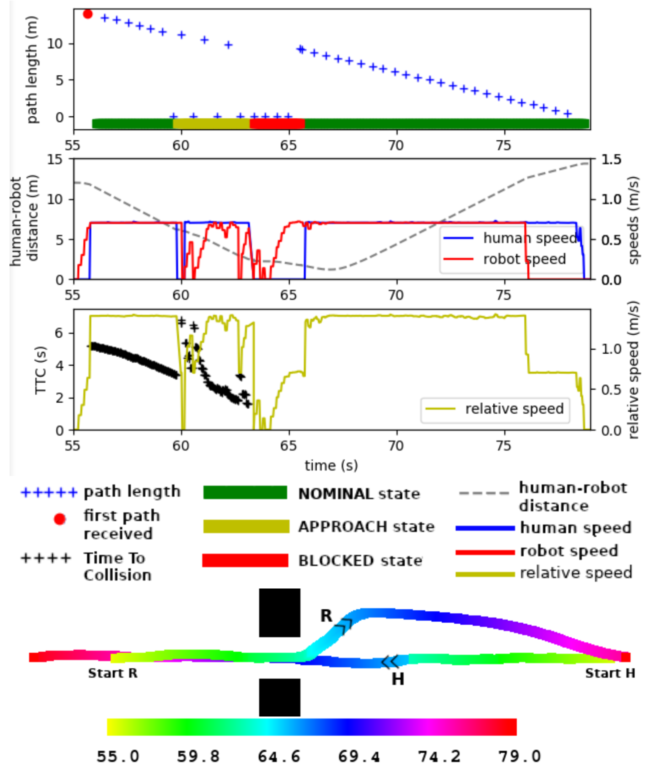

Fig. 1. Plots for the CoHAN planner on the robot in the doorway scenario. The robot blocks the human's path while crossing, and it makes the human agent go into the approach and then blocked state. The velocity of the robot decreases as it approaches the human, and the TTC value does not fall below a certain value, showing some intended human-aware behavior.

predictions. Thanks to its high tunable property, this planner also helps to generate different behaviors.

To generate a variety of specific challenging situations, we created what we call *Attitudes*. They are modes that affect both goal decisions and reactions regarding other agents. Some of the currently implemented *Attitudes* consist of: randomly picking a new goal like someone changing its mind, harassing the robot by constantly going in front of it like a child could do [20], and stopping the agent close to the robot and making it look at the robot for a few seconds before resuming its goal to emulate a curious behavior.

B. Usability

A simple graphical user interface component allows the human operator to control all the agents by sending goals and starting scenarios to repeatedly generate the same situations. All goals and scenarios are predefined in an XML file that can be easily modified. Specifying a radius for a goal makes the agent receive a goal randomly picked within the defined circle. *Attitudes* and an "endless" mode can also be activated through this component. The used simulator can be easily switched to match the one used by the robot controller to be challenged. Manual velocity commands can be sent to use scripted trajectories or motion capture to control the human agent.

C. Logs and Metrics

As mentioned earlier, the system logs the execution data and computed metrics that might be useful later to evaluate the interactions, and thus, the performance of the tested robot planning system. We provide two kinds of visualizations using these data. The first visualization, at the bottom of Fig. 1, is the paths taken by each agent and colored over time according to a corresponding legend that helps estimate an agent's position at a specific moment. The other visualization shown at the top of Fig. 1 is composed of several plots showing some metrics over time. The legend for these plots is shown just above the time-colored paths. These plots help to identify conflicts, analyze speed variations, and show a metric called time to collision (TTC). It estimates the time remaining before the agents collide. We can argue that TTC can correspond to a "threat feeling" since a low TTC value corresponds to a high threat of collision. Hence, the social robot navigation planners can be tuned to maintain a certain TTC value to make humans more comfortable.

IV. Results

In this section, we first present some results about the limits of reactive-only agents, and secondly, about the performance of our system.

A. Limits of reactive-only agents

In order to highlight the limitations of overused reactive-only approaches, we present results obtained with a PedSim_ROS (or simply PedSim) agent. Pedsim is a pedestrian simulator that uses the social force model. It is very efficient for generating crowds to test robot navigation. However, at the individual level, the simulated agents are purely reactive and have no decisional abilities like most pedestrian simulators.

Consider the doorway scene shown in the upper part of Fig. 2. The robot is blocking the way which the human agent intends to cross. The Pedsim agent approaches the robot and tries to push itself through, but it fails due to a very high value of social force. The agent never stops moving and tends to go right or left along the wall before wiggling again just in front of the robot. This kind of behavior is confusing and makes its intentions unclear to the robot planner. The narrow corridor scenario shown in the lower part of Fig. 2 also exposes some limits. Here the path is blocked by the static robot, and the Pedsim agent slowly gets closer and closer to the robot before squeezing itself between the wall and the robot. For some reason here the social forces allowed the agent to pass, unlike the previous example. But it highlights that the Pedsim agent doesn't use a defined hitbox or footprint for the agent and relies only on the repulsive social forces to prevent the collisions. The lack of properly defined collision shapes makes the agent temporarily go through walls and other agents. As a consequence, it breaks many intricate scenarios where a rational decision should be taken and results in unrealistic situations like the above.

Based on the above observations, we can infer that such approaches, despite being efficient for large spaces or crowds, could lead to confusing behaviors in narrow environments and intricate scenarios, where conflict resolution is required.

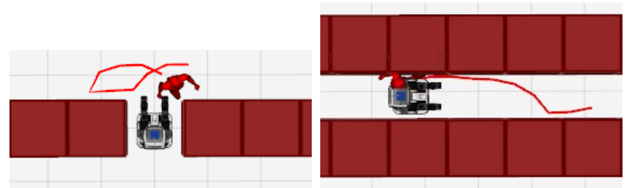

Fig. 2. In the doorway scenario at the top, the reactive-only (Pedsim) agent never stops moving and trying to go through the robot even though its path is blocked. Moreover, sometimes the agent squeezes itself between the wall and the robot colliding with both, like with the narrow corridor scenario at the bottom. Not having collisions is a big limitation for Pedsim since it means it can't realistically react in intricate conflicts.

B. Analysis for a human-aware navigation system

To expose how both the challenging situations created and the data recorded by our system could help evaluate social robot navigation, we present detailed results and their interpretation with a robot running the original CoHAN system. In the doorway scenario, the navigating robot blocks the way before the human can cross. Since the human agent stops and leaves enough space for the robot to cross, CoHAN eventually makes the robot cross and move to the left to clear the way and avoid the human. The time-colored paths in Fig. 1 illustrate this execution well. Furthermore, the plots in the same figure give additional information about the execution. First, we notice the conflict detection (path length equals zero) that makes the human switch to the approach and then to the blocked state. These plots also show that the robot slowed down before entering the doorway and also when the human-robot distance decreases while crossing. After a certain value of TTC, the robot's speed drops to almost zero, and then it increases again when the robot changes its direction to continue its navigation. All these observations can be useful for improving or evaluating the social robot planner's performance: finding ways to decrease the blocked state time for the human, maintaining a particular threshold for TTC, or waiting for the human to cross the door without blocking.

C. Different planning systems and metrics

We chose three robot navigation planners. The first planner is called SMB, which stands for Simple Move Base. It uses *teb_local_planner* and the ROS navigation stack with almost every default parameter and an additional process to consider the human agent as a static obstacle to avoid it. Thus, it is not a human-aware planner. The second one is a human-aware robot navigation planner from Kollmitz et. al. [10]. It is referred to here as TDP. The third one is the already mentioned CoHAN planner. TABLE I presents some metrics calculated from three runs with the doorway scenario. It shows the human's time to reach the goal, minimum TTC, the duration for which the human is blocked, and the time spent by the robot at a distance less than $2m$ from the human. Note that these metrics are not automatically computed but can be easily extracted from the log data. Thus, other metrics can be added without any

	Time to goal (human)	Min TTC	Time blocked	Time below 2m
SMB	23.6	1.5	1.1	3.1
TDP	27.5	2.4	7.1	5.8
CoHAN	22.9	1.6	2.3	3.0

TABLE I
DIFFERENT METRICS COMPUTED FROM LOG DATA CORRESPONDING TO RUNS OF DIFFERENT ROBOT PLANNERS IN THE DOORWAY SCENARIO.

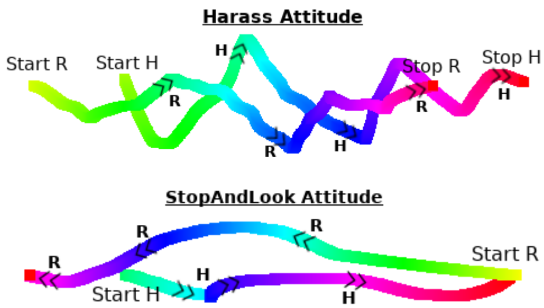

Fig. 3. Behaviors obtained by activating the *Harass* and *StopAndLook Attitudes*. With *Harass* (top), the human always goes in front of the robot. By observing the colors, one can that the human is always ahead of the robot. With *StopAndLook* (bottom), the human stops to look at the robot for a few seconds when close to it. This can be seen in the plot as a sudden change of color from green to dark blue in the human's path.

modifications as long as they are based on the data shown in the plots. Then, more data can be logged to be able to compute even more metrics. The interpretation of the results in TABLE I depends on how the corresponding system has been designed to behave and on what needs to be improved.

D. Generating different behaviors

Our system is capable of generating different agent behaviors in order to diversify situations and conflicts to decently challenge the robot system. One way to do this is by tuning parameters about the navigation conflicts and the ones related to the geometric planner. Changing the velocity and path planning of the agent has a lot of influence on the produced behavior. Besides parameter tuning, activating *Attitudes* produces complex behaviors and reactions. The execution of the mentioned *Harass* and *StopAndLook Attitudes* can be seen in Fig. 3 which shows the time-colored paths of the agents. Concerning the *Harass Attitude*, by paying attention to the colors, we see that the human is always in front of the robot that continuously tries to avoid the harassing agent causing erratic movements. In the same figure, at the bottom, we see thanks to the color discontinuity how the *StopAndLook Attitude* makes the human suspend its goal to stop and briefly stare at the robot before moving again.

E. Long runs and scenarios

Finally, the system is capable of conducting long runs, thanks to the Boss interface that autonomously sends goals to the agents. Such a feature is interesting because it helps to test the challenged system's stability and robustness. Moreover, when randomness is added to the goals of the long run, unexpected situations and conflicts of interest might be

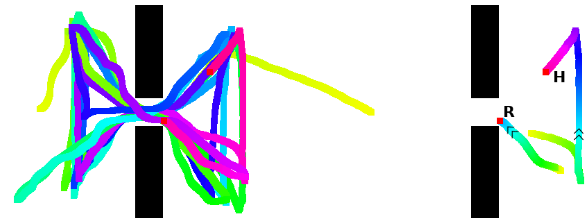

Fig. 4. Execution of the long run scenario using the TDP planner and our system. We see the complete set of time-colored paths on the left. On the right, the same path is cut around the moment when the robot got stuck in the wall. Long runs help to debug such unexpected issues.

generated. For instance, Fig. 4 depicts a long run executed with our system and the TDP planner. The agents were made to endlessly loop over four goal positions (each with a $1m$ radius) but in reverse order to create as many conflicts as possible. After 3 minutes, the robot got stuck in the wall of the doorway, indefinitely blocking the path for the human, which could be an issue of interest. In addition to highlighting problematic situations where the robot doesn't act as expected, long runs can expose low-level issues like unexpected crashes or memory leaks.

V. CONCLUSION AND FUTURE WORK

Human-aware robot navigation is rapidly growing, but the community lacks good human agent simulations to test and debug their systems before real-life experiments. Reactive-only approaches exist, but we have shown that they are limited. Through this system, we propose a pertinent approach to address this issue. Then, by focusing on a specific human-aware robot planner, we showed that our system generates conflicting situations that need to be resolved by making rational choices. Moreover, all the metrics and data recorded during execution allow us to evaluate the interaction and behavior of the robot. Our system can also generate different and tunable behaviors that diversify the situations and conflicts imposed on the robot, and thus, it helps to debug and tune the system. The long runs provide additional potential ways to improve the system.

Our work obviously has limitations. First, we are looking forward to testing other similar systems and to comparing them with our system. Secondly, we claim to generate only reactive and some rational behavior, which is still far from natural or realistic human behavior. Finally, we currently handle scenarios with two agents only, the human and the robot. We can run scenarios with other human agents, but they will be treated like robots.

We already use effectively this system to test our own human-aware motion planners and we will refine it over time thanks to the tests conducted. In future work, we plan to handle scenarios with more than one intelligent human agent, with groups and maybe even with crowds. We also intend to enrich the set of available metrics and of available conflicting scenarios that could help evaluate social robot navigation.

References

[1] S. Vattam, "Breadth of Approaches to Goal Reasoning: A Research Survey," *Goal Reasoning: Papers from the ACS Workshop*, p. 17, 2013.

[2] B. Johnson, M. W. Floyd, A. Coman, M. A. Wilson, and D. W. Aha, "Goal Reasoning and Trusted Autonomy," in *Foundations of Trusted Autonomy*, 2018, vol. 117.

[3] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726–1743, Dec. 2013. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0921889013001048

[4] J. Rios-Martinez, A. Spalanzani, and C. Laugier, "From proxemics theory to socially-aware navigation: A survey," *International Journal of Social Robotics*, 2015.

[5] R. Guldenring, M. Görner, N. Hendrich *et al.*, "Learning local planners for human-aware navigation in indoor environments," in *IEEE/RSJ IROS*, 2020.

[6] S. B. Banisetty, V. Rajamohan, F. Vega, and D. Feil-Seifer, "A deep learning approach to multi-context socially-aware navigation," in *2021 30th IEEE International Conference on Robot Human Interactive Communication (RO-MAN)*, 2021, pp. 23–30.

[7] S. B. Banisetty, S. Forer *et al.*, "Socially-aware navigation: A non-linear multi-objective optimization approach," *arXiv preprint arXiv:1911.04037*, 2019.

[8] P.-T. Singamaneni and R. Alami, "Hateb-2: Reactive planning and decision making in human-robot co-navigation," in *International Conference on Robot & Human Interactive Communication*, 2020.

[9] E. A. Sisbot, L. F. Marin-Urias *et al.*, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, 2007.

[10] M. Kollmitz, K. Hsiao, J. Gaa, and W. Burgard, "Time dependent planning on a layered social cost map for human-aware robot navigation," in *Proc. of the IEEE Eur. Conf. on Mobile Robotics (ECMR)*, 2015. [Online]. Available: http://ais.informatik.uni-freiburg.de/publications/papers/kollmitz15ecmr.pdf

[11] G. Ferrer, A. Garrell *et al.*, "Social-aware robot navigation in urban environments," in *IEEE ECMR*, 2013.

[12] X.-T. Truong and T. D. Ngo, "Toward socially aware robot navigation in dynamic and crowded environments: A proactive social motion model," *IEEE Transactions on Automation Science and Engineering*, no. 4, 2017.

[13] A. Aroor, S. L. Epstein, and R. Korpan, "MengeROS: a Crowd Simulation Tool for Autonomous Robot Navigation," *arXiv:1801.08823 [cs]*, Jan. 2018, arXiv: 1801.08823. [Online]. Available: http://arxiv.org/abs/1801.08823

[14] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[15] W. van Toll, F. Grzeskowiak, A. L. Gandía, J. Amirian, F. Berton, J. Bruneau, B. C. Daniel, A. Jovane, and J. Pettré, "Generalized microscropic crowd simulation using costs in velocity space," in *Symposium on Interactive 3D Graphics and Games*. ACM, 2020, pp. 1–9. [Online]. Available: https://dl.acm.org/doi/10.1145/3384382.3384532

[16] X. Puig, K. Ra, M. Boben, J. Li, T. Wang, S. Fidler, and A. Torralba, "VirtualHome: Simulating Household Activities Via Programs," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT: IEEE, Jun. 2018, pp. 8494–8502. [Online]. Available: https://ieeexplore.ieee.org/document/8578984/

[17] L. Yige, L. Siyun, L. Chengshu, P.-D. Claudia, and S. Silvio, "Interactive pedestrian simulation in igibson," *RSS Workshop on Social Robot Navigation*, 2021.

[18] B. Shen, F. Xia, C. Li, R. Martın-Martın, L. Fan, G. Wang, S. Buch, C. D'Arpino, S. Srivastava, L. P. Tchapmi, K. Vainio, L. Fei-Fei, and S. Savarese, "igibson 1.0: a simulation environment for interactive tasks in large realistic scenes," *arXiv preprint*, 2020.

[19] P. T. Singamaneni, A. Favier, and R. Alami, "Human-aware navigation planner for diverse human-robot ineraction contexts," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[20] T. Nomura, T. Kanda, H. Kidokoro, Y. Suehiro, and S. Yamada, "Why do children abuse robots?" *Interaction Studies*, vol. 17, no. 3, pp. 347–369, 2016.