A unified benchmark framework for autonomous Mobile robots and Vehicles Motion Algorithms (MoVeMA benchmarks)

Daniele Calisi, Luca Iocchi, Daniele Nardi
Department of Computer and Systems Science
"Sapienza" University of Rome, Italy
Email: *lastname*@dis.uniroma1.it

***Abstract*— In this paper we present a framework of benchmarks for evaluating and comparing motion algorithms for autonomous mobile robots and vehicles.**

These benchmarks focus on real-world issues, such as uncertainty and nonholonomic constraints, and on those situations that have been proven to be critical by the past research.

I. Introduction and related work

Performance metrics, benchmark databases and widely-accepted comparison methodologies are very important tools for all scientific, industrial and commercial products. Robotics research, as a special case, often lacks of these kinds of studies, thus making it difficult to understand the quantitative improvements in specific issues and problems, as well as failing in providing methodologies to compare different methods in different environments and scenarios.

In this paper we will concentrate in a particular field of robotics: motion algorithms for autonomous mobile robots and vehicles. This is indeed a very important area in robotics, since, as stated by Latombe [6], robot motion is "eminently necessary since, by definition, a robot accomplishes its tasks by moving in the real world".

Other research communities in robotics and artificial intelligence have found benefits by the definition of open and standardized benchmarks and performance measurements. In fact, all improvements and new methods can have immediate verification and can be compared with existing researches, understanding their weaknesses and strenghts, with respect to specific settings or scenarios.

Some examples of these benchmarks are the following: Radish[1] (Robotic Data Set Repository) and Rawseeds[2] (developed by the Politecnico of Milano), a collection of robot data logs used to test localization and SLAM[3] methods; the UCI Machine Learning repository[4]; the PASCAL Collection[5] for vision-based object recognition and many others.

Also in the motion planning community there has been some effort in this direction: for example the Motion Planning Puzzles of the Parasol Lab at A&M University of Texas[6] (which includes the famous "alpha puzzle") and the MOVIE Project[7] (aimed at motion planning in virtual environments). Moreover, motion planning is also one of the main areas of the EURON Benchmarking Initiative[8]. The aims of these benchmarks are strongly focussed in motion planning with many degrees of freedom and off-line computation; there is indeed a very weak attention to issues such as uncertainty, information gathering by sensors, execution control, etc. However, in the case of autonomous vehicles or mobile robots (usually involving a few degrees of freedom), difficulties arise because of real-world unpredictability and real-time constraints. Moreover, planning algorithms usually assume to be able to access the whole world description, that is not realistic when dealing with real autonomous mobile robots and vehicles.

Another issue when dealing with motion algorithms is that it is an "active task": it is not possible to collect logs and then run algorithms off-line, since each algorithm choice determines a new situation and thus different sensor readings and action possibilities. Data for performance metrics in this kind of algorithms can only be collected real world benchmarks or simulators that are able to model the real world responses to robot actions.

The NIST (National Institute of Standards and Technology) is currently conducting some projects for performance metrics and evaluation of autonomous vehicles in real world scenarios (e.g., the Mobile Autonomous Robot Software Project[9]): its effort is very important in order to detect methodologies in performance measurements, but it still does not provide a widely accepted and used set of benchmarks.

A research group at University of Zaragoza [9] has developed an Automatic Evaluation Framework for obstacle avoidance algorithms. With respect to their work, besides a careful attention to the real-world issues such as uncertainty, we want to develop a more general benchmark system for motion algorithms, including, but not limited to, obstacle avoidance techniques.

[1] http://radish.sourceforge.net
[2] http://rawseeds.elet.polimi.it/home
[3] simultaneous localization and mapping
[4] http://mlearn.ics.uci.edu/MLRepository.html
[5] http://www.pascal-network.org/challenges/VOC/databases.html
[6] http://parasol-www.cs.tamu.edu/groups/amatogroup/benchmarks/mp/
[7] http://www.give.nl/movie/
[8] http://www.euron.org/activities/benchmarks/index.html
[9] http://www.isd.mel.nist.gov/projects/darpa_mars/index.html

II. Definitions and objectives

The aim of this work is to provide an extensive benchmark for algorithms that can be used for mobile robot and vehicle autonomous navigation. In the following we will use the terms “robot” and “vehicle” interchangeably; in particular, in this first version of the benchmark, we are interested in omni-directional robots, skid steered and Ackerman steered vehicles. Our main goal is to test and compare “motion algorithms”. This *can* involve some limited mapping and localization skill, in order to understand the goal and being able to plan the path, but the main areas that we want to cover are: path and motion planning, obstacle avoidance, and control. Moreover, since we are interested on real systems, we strongly encourage the algorithms to be tested in situations in which critical real-world issues arise, e.g.:

- the environment in which the robot acts should not be modified in order to make its tasks easier;
- uncertainty, both on sensor readings and on action results, should be taken into account;
- nonholonomic constraints and dynamics need to be considered when they become relevant (e.g., at high speeds);
- only a partial knowledge of the environment is accessible by the robot, acquired by sensor readings.

With respect to the last item, it is commonly accepted that a global knowledge of the environment can be provided to the robot beforehand (e.g., a map); anyway, it is realistic that this information can be partial and includes some uncertainty.

Every evaluation framework should consist of at least two parts: a performance evaluation methodology and a set of benchmark problems, environments or situations where the whole systems or the single algorithms can be tested and compared with each other. In the following we will first describe a possible set of performance metrics and then we will explain our proposed set of benchmarks for robot motion algorithms.

III. Performance metrics and constraints

We identified some performance metrics in this kind of application, but it is important to underline that no measure can be considered more significant than the other without a specification of the particular application goals. The most used performance metrics to compare autonomous motion algorithms are:

- the time to reach the goal, that is often the only performance metrics used in this kind of application;
- the precision at the target, i.e., how far is the robot from the target position: this can be important if the robot motion is performed in order to accomplish some higher level task that involves other actions.

In addition to these, it can be important to identify another measure that can be requested in order to maintain the vehicle stability and integrity: the *smoothness of the trajectories followed by the robot*. Other interesting measures can be found in [8], for example *security metrics*, i.e., the distance from obstacles along the trajectories.

Moreover, in real-world and industrial applications, it is often necessary to set some constraints that the algorithm has to satisfy in order to be used. The concept of constraints is different from the performance metrics described above: while the latter can be used to optimize the performance or to measure the differences of two algorithms, the former have to be considered strict needs of the application at hand: if the robot motion does not satisfy one of these constraints, the mission is declared failed. Some of these constraints can be:

- maximum speed allowed;
- maximum acceleration;
- minimum distance from the obstacles.

IV. The MoVeMA benchmark database

The final goal of every robotic research should be the application to real world robots and environments. However, since the direct development in real world scenarios is not feasible, simulation devices are often needed. Moreover, it is much more easier to build scenarios using a simulator: this can help in revealing critical situations that need more attention and further analysis also in the real world. Anyway, there are real-world issues that cannot be handled by simulators and the exclusive use of synthetic benchmarks can lead to unexpected failures when applying methods on real robots. For these reasons, the proposed benchmark database is divided into two parts: simulated benchmarks and real world benchmarks.

A. Simulated benckmarks

Simulation is one of the most important tools in robotic development. On the one hand, it enables the evaluation of different alternatives during the design phase of robot systems and may therefore lead to more general solutions. On the other hand, it supports the process of software development by providing a replacement for robots that are currently not available (e.g. broken or used by another person) or not able to endure long running experiments (e.g. learning tasks). Furthermore, the execution of robot programs inside a simulator offers the possibility to perform an easier and faster debugging phase before the first real experiment.

The creation of our set of simulated benchmarks begins with the choice of the specific simulation software to use. In particular, we are interested in simulators that are able to deal with different robot models, kinematics and uncertainty. There is a big amount of free and commercial robotic simulators that corresponds to these features, most of them being long-standing projects and extensively used by the scientific community. Currently, the MoVeMA simulated benchmarks are developed for the Stage simulator of the Player/Stage package[10]. This choice is due to many reasons: first of all, this simulator is widely used by the scientific community and is currently also included in some Linux distributions; thanks to the use of the Player library, client software is presented with the same interface both to real robots and to the Stage simulator; since a world is defined by a bitmap and a text file,

[10] http://playerstage.sf.net

Fig. 1. Three maps taken from the benchmark database, showing also the robot starting pose

it is indeed very easy to generate worlds for benchmarking; finally, it is free and open source. Anyway, other choices can be added in the future, e.g., USARSim[11], Webots[12] or the Microsoft Robotics Studio Simulation Environment[13].

The set of the environments includes all the examples given in past research papers (e.g., [1, 3, 7, 10, 12]), especially those that involve an analysis of one algorithm or a comparison between different methods (e.g., [5, 2, 11]). Moreover, we extend this initial collection with other critical situations found through our past research and with many other scenes, using office-like maps, more unstructured environments and so on. For each of these maps a set of pairs starting poses/target poses are given. In Figure 1 some samples from the the benchmarks are depicted.

Simulated environments and situations are considered an important tool in developing new algorithms and analyze the behavior of existing methods with respect to a great spectrum of situations. Automatic testing is very easy to perform, and thus collecting big amount of data for statistical analysis of complex scenarios. Moreover, since the simulator allows to introduce uncertainty in sensor readings, this can be considered the first step towards the application in the real world.

B. Real world benchmarks

The use of real-world benchmarks, that can be shared among many research groups, presents some practical problems. In order to overcome these difficulties, two solutions have been presented in the past. Scientific competitions, such as AAAI mobile robot competitions[14] and DARPA Grand Challenge[15] are becoming benchmark de-facto, but cannot be used for extensive and continuous testing, due to their high costs and the fact that they take place only few times per year. Another approach is to model standard environments using common elements and materials (e.g., NIST's test arenas for autonomous mobile robots[4], used also in RoboCupRescue[16] competitions), but the performance evaluation, in this case, includes not only the algorithms and software, but also the robot hardware.

Although these two approach are effective, we propose a third approach, that can make it easier to compare various motion algorithms using the very same scenarios. We suggest that every research group arrange its own reproducible experimental conditions (environment and robots) and test foreign (as well as its own) software, collecting and reporting back the results.

To be effective, this kind of shared benchmarks needs a set of rules and policies.

- First of all, from the implementation point of view, a commonly accepted interface to sensors and actuators is needed, as well as a protection system that prevents the foreign code to harm the robot (e.g., hard-limiting the controlling speed and providing an emergency break that has a higher priority with respect to foreign commands). This is currently achieved thanks to the use of Player drivers.
- Moreover, a precise policy about how often and when the experiments are conducted is necessary, in order not to overload the host research group. The policy should be accepted by all research groups that want to join the MoVeMA benchmarks network.
- Real world experiments pose also other kinds of problems: parameters of motion algorithms are usually tuned by trial-and-error sessions, and this is not feasible for remote experimentation. Parameter tuning is actually one of the main obstacles that prevents one method to be widely used. Indeed, the remote experimentation encourages to limit parameters to the minimum, and to prefer *meaningful* parameters (e.g., robot size) with respect to unmeaningful (e.g., magic numbers for obscure coefficients). Moreover, developers can provide tuning software that can automatically set or learn parameter values. We think that providing easy mechanisms for

[11] http://usarsim.sf.net
[12] http://www.cyberbotics.com/products/webots/index.html
[13] http://msdn.microsoft.com/en-us/robotics/
[14] http://www.aaai.org/Conferences/AAAI/2008/aaai08robot.php
[15] http://www.darpa.mil/grandchallenge
[16] http://www.robocuprescue.org

parameter tuning, being them manual or automatic, is an important issue for motion algorithms to be widely used also outside the academic and research world: a step further in this direction can be to give much more importance to experiments performed in remote laboratories by foreign research groups.

Actually, the same policies can be used to share simulated scenarios built in commercial simulators.

V. CONCLUSIONS AND ON-GOING WORK

This work presented a framework of benchmarks for motion algorithm experimentation and performance evaluation. The aim is to provide a common testbed for quantitative evaluation and comparison of different motion strategies for autonomous vehicles and mobile robots.

Many issues need further development. For example, the development of a common software interface that is not dependent to Player drivers can be useful to include more robots and sensors. Moreover, it is necessary to define a common, fixed and well-defined set of performance metrics, that have to include, as much as possible, all the needs of mobile robot applications.

In order to be effective, all evaluation frameworks need to be widely accepted and used. For this reason, our database of simulated benchmarks and our own real-world scenarios are available for download and use at `www.dis.uniroma1.it/~calisi/movema`: we think that the scientific community can help in improving and enhancing this database from its very beginning.

REFERENCES

[1] J. Borenstein. The vector field histogram-fast obstacle avoidance for mobile robots. *Robotics and Automation, IEEE Transactions on*, Volume: 7 Issue: 3:278 –288, 1991.

[2] J.L. Fernández, R. Sanz, J.A. Benayas, and A.R. Diéguez. Improving collision avoidance for mobile robots in partially known environments: the beam curvature method. *Robotics and Autonomous Systems*, 46(4):205–219, April 2004.

[3] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, March 1997.

[4] A. Jacoff and E. Messina. DHS/NIST response robot evaluation exercises. In *IEEE International Workshop on Safety Security and Rescue Robots*, Gaithersburg, MD, USA, 2006.

[5] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings of the IEEE Conference on Robotics and Automation (ICRA)*, pages 1398–1404, 1991.

[6] J. C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[7] J. Minguez and L. Montano. Nearness diagram (ND) navigation: Collision avoidance in troublesome scenarios. *IEEE Transactions on Robotics and Automations*, 20(1):45–59, 2004.

[8] N. Muñoz, J. Valencia, and N. Londoño. Evaluation of navigation of an autonomous mobile robot. In *Proceedings of Performance Metrics for Intelligent Systems Workshop (PerMIS)*, pages 15–21, August 2007.

[9] I. Raño and J. Minguez. Steps towards the automatic evaluation of robot obstacle avoidance algorithms. In *Proc. of Workshop of Benchmarking in Robotics, in the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[10] R. Simmons. The curvature-velocity method for local obstacle avoidance. In *Proceedings of IEEE International Conference on Robots and Automation*, volume 4, pages 3375–3382, April 1996.

[11] C. Stachniss and W. Burgard. An integrated approach to goal-directed obstacle avoidance under dynamic constraints for dynamic environments. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[12] I. Ulrich and J. Borenstein. VFH+: Reliable obstacle avoidance for fast mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1572–1577, May 1998.