BIO-INSPIRED MULTI-ROBOT BEHAVIOR FOR EXPLORATION IN LOW GRAVITY ENVIRONMENTS

Ruben Martinez-Cantin
Departamento de Informatica e Ingeniera de Sistemas,
University of Zaragoza, Spain
rmcantin@unizar.es

Abstract

In this paper, we present a new paradigm of biomorphic robot which is based on capabilities of animals such us grasshoppers or fiddler crabs. Biomorphic robots seem to be the future in exploration of hazardous environments. As has been proved in recent works, multi-robot platforms are the best solution for exploration of unstructured environments. In the forthcoming years, planetary missions must be optimized for low gravity environments. In this work, we use hopping robots to minimize energy cost in exploration of wide areas. Nature also use good sensors. For example, fiddler crabs have a pair of retractile onmidirectional eyes. Considering all these conditions (multi-robot, hop movement, omnidirectional views), data fusion is one of the most important aspects during autonomous navigation. In this paper, I present a comparative of the behaviour of most extended techniques of estimation and data fusion, specially oriented to localization and mapping.

1 INTRODUCTION

Recent missions to Mars have proved that mobile robots are the best way in low-cost planetary exploration. Spirit, Opportunity and Sojourner [13] are just a few example of this rising field.

The special capabilities of these robots result of the adaptation to the terrain that they are designed to explore. Searching for this "adaptation", some recent researches are focused on bio-inspired robots.

Looking into nature gives us the opportunity to discover excellent mechanisms of locomotion, perception and interaction between organisms [18]. Grasshoppers, crabs, ants or bees are some examples. In this paper, we present several new paradigms of biomorphic robots which are based on capabilities of these animals. These robots may be the future in exploration of hazardous environments. Several recent works consider multi-robot platforms a good solution for exploration of unstructured environments considering three aspects: efficiency, specialization and redundancy (resistance to local failures). Normally, those proposals present a combination of rovers or nanorovers equipped with different kinds of sensors (proximity, cameras, inertial, etc.). However, several bodies in the Solar System have lower gravity than Earth.

In fact, Moon and Mars, which are the next step in Solar System exploration, are in this group. So, planetary missions during the next years must be oriented for low gravity environments.

In this work, we use hopping robots to minimize energy cost during exploration of wide areas in low gravity. Hoppers have also the advantage to be able to jump over obstacles several times higher than the robot. On the other hand, fiddler crabs are a good example of sensor efficiency because they have a couple of 360 degrees FOV eyes, which can be folded and protected during dangerous manoeuvres. A colony of robotic crabs would mean a large amount of pictures, but also requires an accurate knowledge of the localization of the robots in a vast terrain.

We suppose autonomous navigation, which requires a complex algorithm to process all this information and give an accurate estimation. Section 3 presents several algorithms for this purpose.

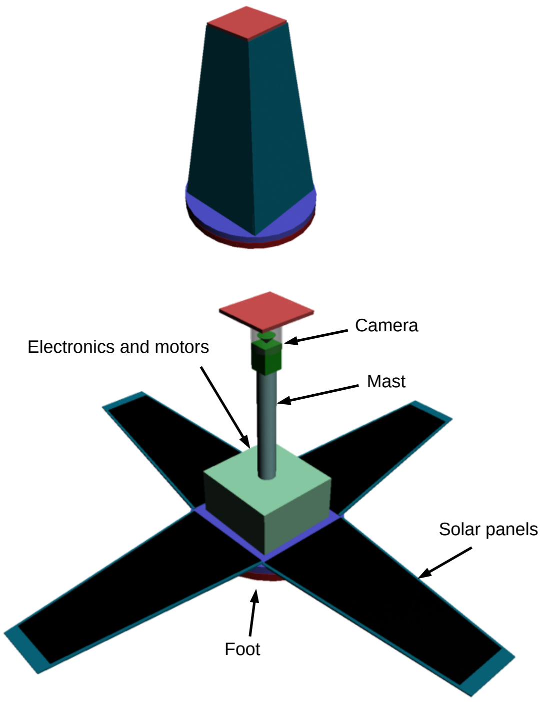

Figure 1: Scheme of the hopper with the side panels opened and closed.

2 HOPPING ROBOTS

The main paradigm of mobile robots is the wheeled vehicle because it is cheap and it is the most efficient locomotion system in typical scenarios on Earth. However, the low gravity of other celestial bodies such as Moon or Mars requires the development of new concepts of mobility systems which may seem inefficient in Earth.

Recent studies in the field of hopping robots had shown the potentials of this kind of locomotion in low gravity environments [20, 12]. Their efficiency comes from two facts: the low rate of energy cost per meter covered in low gravity and the capability to cross over high obstacles, i.e. big rocks, without needing to go around it.

A comparative between a wheeled nanorover and a hopper is presented in Schell et al [15]. The results show that, in Mars gravity, the energy for a jump is the same order in both platforms. On the other hand, if the nanorover needs to avoid some obstacles, the distances travelled increases. Then, the energy cost increases, too. Furthermore, in a lower gravity environment like Moon, the energy cost in the hopper decreases linearly since in the nanorover remains constant.

The design presented in this paper is based on other hoppers recently developed. A scheme of the robot can be viewed on figure 1.

The main parts of the robot are:

Omnidirectional camera: using a simple camera pointing to a parabolic mirror we

Non navigational sensors: underground sonar, microscopic camera, spectrometer, chemical sensors
Small tools for mining, ground prospecting and micro-manipulation
High performance CPUs for high level computing, real-time map building and data distribution

Table 1: Possible sensors and science instruments

can take omnidirectional images.

Side panels: they have a triple function. When closed, they are used as a shield for impacts and dust. When opened, they became solar panels for recharging the batteries. Finally, when opening, they are used as a self–righting system.

Hop mechanism: based on the hopper presented on [15]. It is formed on a platform, a spring, a little motor, a gear-box and, optionally, wheels for smooth displacements. Schell et al. [15] have considered also the possibility of using the foot as a scoop for collecting terrain samples. This can be useful for example, if we install a little spectrometer or analyzer inside the hopper .

Colored mast: it is used to sustain the camera and other sensors, but also to identify the hopper by other robots. Like fiddler crabs, this mast (and the camera also) is protected in hazardous situations by the side panels.

Electronics and Batteries: this is the brain of the robot, with the control, communications and processing unit.

Additional sensor and science instruments: it depends on the global mission and the role of the robot. Some examples are presented on table 1.

The robot can get three basic configurations depending on the position of the lateral panels: ready and shell with the panels closed and deployed with the panels opened. The *ready position* is the previous phase to the jump. The foot has achieved the take-off angle and the spring has been loaded according to the estimate distance of the jump. The *shell position* is used to protect the electronics, camera and solar panels at the landing. In addition, this configuration can be used during night to avoid dust to break or cover the camera, mirror or solar panels, leaving the robot in a sleeping state. Finally, the *deployed position* is used as a self–righting mechanism. But also, in this position, the camera can take images and the solar panels can recharge batteries.

Considering the weight and capabilities of other prototypes, the estimated weight of this hopper would be 2–3 kg. Which means that the hopper would achieve jumps about 30 cm. high and 60 cm. long in 1g.

3 MEASURING LOCALIZATION

The first step in robotic navigation, specially in multi–robot environments, is to get an accurate localization of each robot. On the other hand, the bad resolution in the distance of the hops, requires sophisticated systems of estimation and filtering to increase the accuracy of the measures. This is the focus of the second part of the paper. Our approach is based on a computation of the position and the heading in two different ways.

3.1 Measuring heading

The difference between the hoppers and other robots is the *discontinuity* of the movement. For example, knowing the initial orientation it is impossible to estimate his final heading of the hopper, because it can fall on a rock or slope and roll over.

Figure 2: Typical landscape of Mars taken by Mars Pathfinder [9].

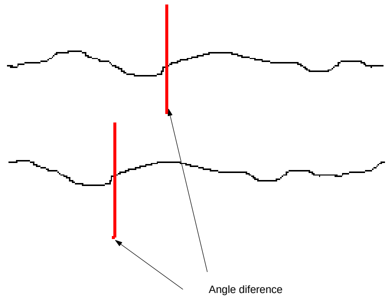

Figure 3: Heading computation using horizon lines. As panoramic views, we suppose that the whole line is equivalent to 360°

The solution is to estimate the orientation using an absolute reference invariant with big translations. In projective geometry, this features are the infinity points; in this case we will use the horizon line, which is a special case of infinity points.

The typical landscape in a planetary mission (like Moon or Mars) can be seen in figure 2. It is a flat surface, with a light slope and high density of small rocks with some big rocks. Normally, the horizon is a smooth curve of far away peaks and hills.

Thanks to the high contrast in the horizon it is easy to extract this curve with a simple edge detector. From this point we can match the whole horizon curve captured by the omnidirectional camera with previous curves. There are several algorithms developed for this propose, even if there is a projective transformation or partial occlusion [14]. Then we can compute the deviation between the curves and so, the relative orientation. In fact, a partial occlusion in the horizon means that there is an possible obstacle in this way.

Then, we can compute the heading of the robot directly as the displacement between the horizon lines as can be seen in figure 3. If we suppose that the horizon line is at infinitum, then the heading computed is absolute and precise.

3.2 Measuring positioning

The problem of computing localization and mapping using a single camera is that this can be extracted directly in one step, as the result of a bearing-only measure. However, several algorithms have been developed to deal with this fact [1, 3, 4]. Basically, the solution is to delay the computation to next steps.

Then we can get range information using basic triangulation or trilateration [1]. In this case, we need need to cope with symmetries, which complexity is increased in 3D.

If we are using a camera, we can also apply structure from motion [3, 4], which is a classical reference in computer vision. It uses properties from the projective geometry. In the 3D case, it has the advantage to use directly the information of both angles (azimuth and heading).

Finally, we can use optimization techniques like Bundle Adjustment [4] that uses directly the bearing measures.

4 COMPUTING LOCALIZATION

Once the position and heading is measured with the methods presented in the previous section, we need to compute the final localization. A filtering process is needed to cope with noise in sensors and data fusion.

On the other hand, since we are exploring the environment, a typical approach is to build a map (localize the environment features) as the same time we localize the robots. This is addressed in the literature as *Simultaneous Localization and Map-Building* (SLAM) or *Concurrent Mapping and Localization* (CML).

However, when we have multiple robot we can use other approaches based on cooperative localization without references of the environment. Then we do not have complete information of the environment, but we still have partial information direct from the current observation which is enough for autonomous navigation.

4.1 Simultaneous Localization and Map-Building

There is a lot of literature about SLAM. Since the problem was first addressed by Smith and Cheeseman [16], most solutions have been based on recursive bayesian algorithms, especially those based on Kalman Filters [5, 12].

However, Kalman Filter is the optimal estimator only in the linear-gaussian case and SLAM is a non-linear system due to the bearing component in the process and in the measures. Extended Kalman Filters solve the non-linear case in a suboptimal way. But, recent researches have proved this algorithms became inconsistent in large loops [10, 2].

Sequential Monte-Carlo algorithms (frequently called Particle Filters) also solve the non-linear and non-gaussian case, but with a higher computational cost which made it unfeasible in real-time for high-dimension systems like SLAM. Nevertheless, a raoblackwellised solution have been presented where the trajectory of the robot is computed using a particle filter and the environment features are processed with local EKFs [19]. This allows to build larger loops, but it is also suboptimal, inconsistent and it lose the information about uncertainty of the robot.

Finally, there is another approaches based on local maps [17, 6] which allows larger loops in a consistent way. The trick is to minimize local errors to linearize in a better point. This solution is faster than others but, again, it is suboptimal and with very large loops, it fails.

4.2 Cooperative Localization

As we have seen before, the best results are given in the Sequential Monte-Carlo techniques. However, as we have a system with an increasing number of dimensions, the problem becomes unfeasible.

On the other hand, multi-robot platforms allow to localize each robot in a way relative to the other robots. So, in this case, the state vector has a number of dimensions proportional to the number of robots, which is constant in time.

In this methods, the navigation is computed according to local maps based on current robot's observations. Then, each robot tries to localize and identify other robots in their local frame thanks to the colored mast [7, 8]. Finally, we can apply algorithms to combine information such as map merging [11].

Colored mast has a revolution symmetry that prevents to compute heading. However, since we are using an absolute common reference (the horizon line), we can compute relative headings in a separate way.

Furthermore, the methods commented

here has been tested on wheeled vehicles. But,hoppers, thanks to their locomotion system, remains static most of the time (the time flying only takes a few seconds every step). So, the estimation becomes much more easy when only a robot moves between observations.

5 Conclusion

We have present a new robotic system for planetary exploration in low gravity environments. We have consider bio-inspired architectures based on the principles of efficiency and redundancy to guarantee the state of the mission. The result is a prototype of hopper equipped with an omnidirectional camera for navigation tasks.

In addition, we have review the state of the art of multi-robot navigation adequate for such conditions. We have dedicate especial interest to localization because hopping is a very noise way of travelling and it is easy to get lost. Relative localization between robots using Particle Filters seems the most efficient way.

6 Acknowledgements

I wish to thank Carlos Compains for developing the 3D images and animations during this project. I would like to thank also Prof. José J. Guerrero and José A. Castellanos for their support and patience.

References

[1] K. Briechle and U. D. Hanebeck. Localization of a mobile robot using relative bearing measures. *IEEE Transactions on Robotics and Automation*, 20(1):36–44, 2004.

[2] J. A. Castellanos, J. Neira, and J. D. Tardos. Limits to the consistency of ekf-based slam. In *5th IFAC Symposium on Intelligent Autonomous Vehicles*, 2004.

[3] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proceedings of the 9th International Conference on Computer Vision, Nice*, 2003.

[4] M. Deans and M. Hebert. Experimental comparison of techniques for localization and mapping using a bearings-only sensor. In *International Symposium of Experimental Robotics*, 2000.

[5] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localisation and map building problem. *IEEE Transactions on Robotics and Automation*, 17(2):229–241, 2001.

[6] C. Estrada, J. Neira, and J.D. Tardos. Hierarchical slam: real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 2004. To appear.

[7] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Auonomous Robots*, 8(3), 2000.

[8] A. Howard, M. J. Mataric, and G. S. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation.

[9] JPL, CalTech, and NASA. Mars microrover photo gallery. http://mars.jpl.nasa.gov/MPF/rovercom/pixt.html.

[10] S. Julier and J. Uhlmann. A counter example to the theory of simultaneous localization and map building. In *Proceedings of the IEEE Internation Conference on Robotics and Automation*, pages 4238–4243, 2001.

[11] J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai. A practical, decision-theoretic approach to multi-robot mapping and exploration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-03)*, 2003.

[12] R. Martinez-Cantin. Moonhoppers colony. In *54th International Astronautical Congress, Bremen, Germany*, 2003.

[13] L. Matthies, E. Gat, R. Harrison, B. Wilcox, R. Volpe, and T. Litwin. Mars microrover navigation: Performance evaluation and enhancement. *Autonomous Robots Journal*, 2(4):291–312, 1995.

[14] C. Orrite and J. E. Herrero. Shape matching of partially occluded curves invariant under projective transformation. *Computer Vision and Image Understanding*, 93:34–64, 2004.

[15] S. Schell, A. Tretten, J. Burdick, S. B. Fuller, and P. Fiorini. Hopper on wheels: Evolving the hopping robot concept. In *Proceedings of Field and Service Robotics*, pages 379–384, 2001.

[16] R.C. Smith and P. Cheeseman. On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 5(4):56–68, 1986.

[17] J.D. Tardos, J. Neira, P. Newman, and J. Leonard. Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, 21(4):311–330, 2002.

[18] S. Thakoor. Bio-inspired engineering of exploration systems. *Journal of Space Mission Architecture*, 49–79, 2000.

[19] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot. Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association. *Journal of Machine Learning Research*, 2004. To appear.

[20] T. Yoshimitsu, T. Kubota, I. Nakatani, T. Adachi, and H. Saito. Micro-hopping robot for asteroid exploration. *Acta Astronautica*, 52:441–446, 2003.