The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Outdoor Navigation with a Spherical Amphibious Robot

Viktor Kaznov and Mattias Seeman

Abstract— **Traditionally, mobile robot design is based on wheels, tracks or legs with their respective advantages and disadvantages. Very few groups have explored designs with spherical morphology. During the past ten years, the number of robots with spherical shape and related studies has substantially increased, and a lot of work is done in this area of mobile robotics. Interest in robots with spherical morphology has also increased, in part due to NASA's search for an alternative design for a Mars rover since the wheel-based rover Spirit is now stuck for good in soft soil. This paper presents the spherical amphibious robot Groundbot, developed by Rotundus AB in Stockholm, Sweden, and describes in detail the navigation algorithm employed in this system.**

I. INTRODUCTION

Traditionally, mobile robot design is based on wheels, tracks or legs with their respective advantages and disadvantages. Very few groups have explored designs with spherical morphology. During the past ten years, the number of robots with spherical shape and related studies has substantially increased, and a lot of work is done in this area of mobile robotics [1], [2], [3], [4], [5], [6], [7], [8]. Interest in robots with spherical morphology has also increased, in part due to NASA's search for an alternative design for a Mars rover since the wheel-based rover Spirit is now stuck for good in soft soil. Among other, a spherical robot design with inflatable body for planetary exploration was proposed by [9] and [10].

This paper presents the spherical amphibious robot Groundbot, developed by Rotundus AB in Stockholm Sweden, and describes in detail the navigation algorithm employed in this system.

II. RELATED WORK

A. Spherical Robots

Spherical, or ball-shaped, robots possess a number of interesting properties, including but not limited to the following list.

- Cannot be overturned.
- Stability and robustness.
- Easy transformation into a sealed system.
- Option to use an inflatable body.
- Amphibious capabilities.
- Option to construct a spark-free robot for investigation of gas leaks or surveillance around oil tankers in a harbors.
- Appealing to the human eye.

Viktor Kaznov is with Rotundus AB, 11157 Stockholm, Sweden viktor@rotundus.se

Mattias Seeman is with Rotundus AB, 11157 Stockholm, Sweden mattias@rotundus.se

Fig. 1. The Groundbot at an airport.

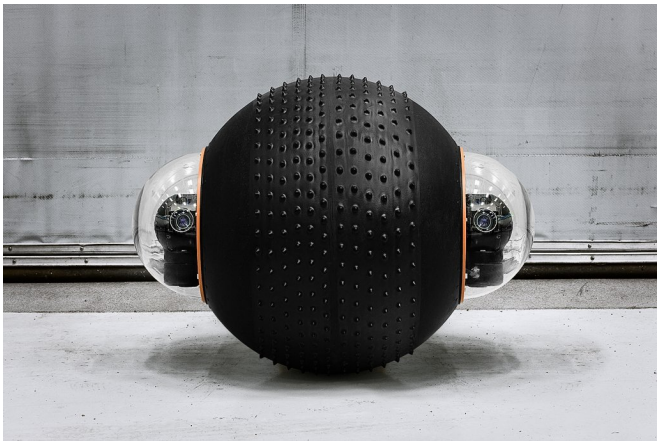
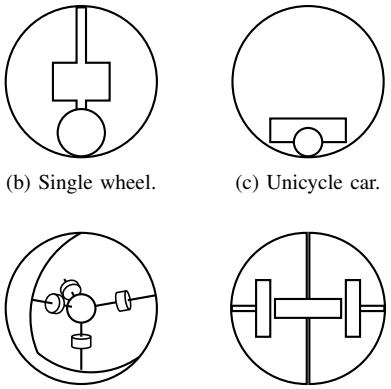


Fig. 2. The Groundbot inside a hangar.

Fig. 3. The Groundbot emerging from the sea.



(b) Single wheel. (c) Unicycle car.

(d) Reciprocating masses. (a) Attached rotors.

Fig. 4. Comparison of spherical robot designs.

- The ball shape invites humans to play.

What can be considered the intrinsic autonomy provided by a spherical morphology allows the robot to naturally negotiate rough terrain and sufficiently small obstacles, simply by rolling over them. Properties such as stability and robustness, make spherical robots especially suitable for surveillance and, to some extent, search and rescue tasks, as noted by [11] and [12]. Spherical robots have also been proposed for planetary exploration without using a dedicated lander [13]. One can see that many features and capabilities of ball-shaped robots are natural consequences of the spherical shape itself, and would require a substantial amount of work to accomplish on other types of robot designs.

However, there are also some technical difficulties with spherical robots. For example, ball-shaped robots present a number of challenges in terms of control and perception, such as camera image stabilization. Any change in motion induces unwanted oscillations that are hard to rectify; [1] mention the robot Rollo that controls oscillation, but only around the rolling axis.

Spherical robots offer limited freedom in placement of sensors; on the Groundbot for example, the most natural place for video cameras are at the point where the main axle meets the shell. Together with any oscillation, this placement easily disorientates operators, as established by [14]. However; since 2009, the current version of the Groundbot offers camera image stabilization about all three axes.

Apart from the Groundbot, which has a propulsion mechanism based on a controllable internal pendulum, a number of other spherical robot designs have been proposed and four of these are depicted in Fig. 4. Designs (a), (b), and (c) all have propulsion mechanisms based on displacing the system's center of mass, although they accomplish this in different ways. [5] describe design (a), with a single drive wheel pushed down onto the bottom of the sphere by a spring. Steering is accomplished by turning the drive wheel. [6] describe a similar design based on a car with unicycle kinematics resting on the bottom of the sphere. [7] propose a different approach to unsettle the balance of the sphere in design (c), based on four weights moving along spokes from the sphere's shell to the hub in its center. The propulsion system of design (d) [15] is based on conservation of angular momentum and has a set of perpendicularly mounted rotors.

Fig. 5. An early version of Rotundus' UGV, the Groundbot, in deep snow.

B. The Groundbot

The Groundbot has a diameter of 60 cm and is capable of navigating rough outdoor terrain at speeds up to 3 m/s (just under 7 mph). The Groundbot is designed to be rugged and durable, and, as depicted in Fig. 5, it is large enough to handle rough outdoor terrain, such as snow, mud, sand or water. As a spherical robot, both locomotion and steering of the Groundbot is accomplished by displacement of its center of mass. Almost all of the robot's weight is suspended on a rigid axle mounted through the shell. The distribution of this weight is managed by two perpendicular motors able to rotate the weight about the robot's center. A schematic illustration of the Groundbot, with axle and pendulum, is available in Fig. 7. The majority of the weight of the pendulum consists of a lithium-ion battery pack able to provide power for up to 12 hours of continuous operation. The Groundbot's hill-climbing abilities were documented in 2009 [16]; the tests show that the current design of the Groundbot can climb slopes up to 15–18 degrees.

An example application of the Groundbot is airport patrolling, depicted in Fig. 1. A number of Groundbots successfully participated in a project on autonomous cooperative

Fig. 6. The Groundbot running on water.

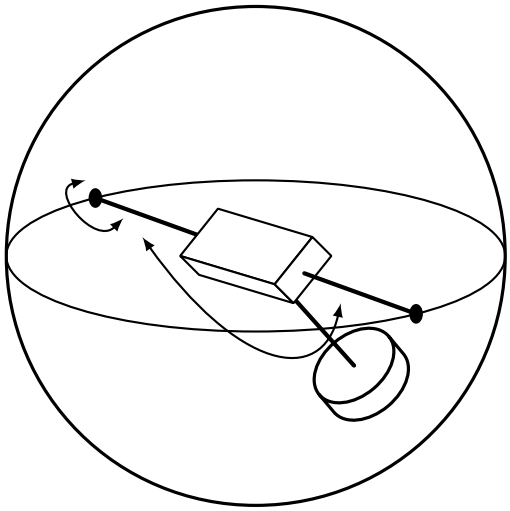


Fig. 7. The Groundbot's locomotion mechanism is based on displacement of the ball's center of mass.

robotics for reconnaissance and surveillance, lead by the Swedish Defence Research Agency [17].

As mentioned above, the Groundbot runs just as well in water as on land as it possesses strong amphibious capabilities. In Fig. 3, the Groundbot goes seamlessly from water to ground using the same locomotion principle, control algorithm and steering principle is both surroundings. There are many advantages to using spherical robots in water; for example, most technical problems related to balancing and video image stabilization disappear.

A number of tests to control the Groundbot's behavior in water have been conducted and show promising results [18], [19].

III. WAYPOINT FOLLOWING WITH THE GROUNDBOT

Upon receiving a list of waypoints from an external controller or operator, the Groundbot's internal *robot controller* goes from *ready mode* into *waypoint following mode*. Among other things, this transition involves converting the waypoint list from WGS 84 coordinates to a local planar metric reference frame with origin at a point in the Groundbot's operating area.

TABLE I

WAYPOINT PARAMETERS (EXCLUDING SENSOR PARAMETERS).

Latitude	WGS 84 latitude.
Longitude	WGS 84 longitude.
Time	Goal time.
Pass Radius	Required accuracy [m] of waypoint pass.
Type	STOP or PASS.

The Groundbot's waypoint following mode is based on the Pure Pursuit algorithm [20], [21]:

1) Determine the current location of the vehicle.
2) Find the path point closest to the vehicle.
3) Find the goal point.
4) Transform the goal point to vehicle coordinates.
5) Calculate the curvature and request the vehicle to set the steering to that curvature.

Each iteration of the algorithm generates low-level commands that set the speed and roll angle of the Groundbot.

A. Current Position

The current location of the Groundbot is readily available to the waypoint following algorithm. If the current leg of the path is cleared, the first waypoint is popped from the list of path waypoints. The current leg is considered cleared (or equivalently, the current waypoint is considered reached) if the line segment between current and previous position (from previous control loop iteration) of the Groundbot intersects the "acceptance circle" (that is, the circle of radius Pass Radius (from Table I) centered in the current waypoint).

B. Closest Path Point

The point on the the path that is the closest to the Groundbot is found in the following manner. Consider two lines, l_1 and l_2. l_1 is the line that passes through the waypoints that are the start and end points of the current leg. l_2 is perpendicular to l_1 and passes through the current vehicle position. The closest path point is found at the intersection between the lines l_1 and l_2. Note that this point can lie outside the current leg (or path segment).

C. Find the Goal Point

The goal point is a point on the line l_1 (from above) that is a distance l, or *lookahead*, away from the current position. If the distance between the closest path point and current position exceeds the lookahead distance, the closest path point is used as goal point. Otherwise, the goal point is found in the following manner. The three points current position (A), closest path point (C), and goal point (B) form a right triangle. The hypotenuse AB has length $c = l$, that is, the lookahead distance. The catheti have lengths a and b. The goal point lies at a distance a along the current leg from the closest path point. Note that this goal point can lie

outside the current leg still; the goal point is restricted to the current leg, by a clamped projection of the goal point onto the current leg, and henceforth used as goal point.

D. Transform the Goal Point to Vehicle Coordinates

The goal point in vehicle (SAE) coordinates, or the *local error*, is found by rotating the global error by the current heading, θ.

$$\mathbf{e}_{\text{local}} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} x_B - x_A \\ y_B - y_A \end{pmatrix}$$

The current heading of the Groundbot is, as the current location, readily available to the waypoint following algorithm.

E. Calculate Curvature and Set Steering

For the Groundbot, the turn radius

$$r_{\text{turn}} = \frac{l^2}{2y_{\mathbf{e}_{\text{local}}}},$$

as derived by Amidi [20], is related to the roll angle, ψ, as

$$\psi = \arctan \frac{R_{\text{shell}}}{r_{\text{turn}}}.$$

It should be noted that before the reference roll angle is calculated, the local error can be negated if its x component is negative to take advantage of the Groundbot's shape, that is, the fact that there is no front or rear to the robot.

At the end of the control loop, the reference speed is also calculated. Apart from distance and time to waypoint, the speed is modulated by the current roll angle since the turning performance of the Groundbot depends on the speed it is carrying.

IV. Waypoint Following Experiments

As [20] notes, Pure Pursuit is simply a proportional controller of the steering angle using the y displacement as the error and $\frac{l^2}{2}$ as the gain. Reasons for its reportedly high performance could be attributed to the ease of experimental tuning of the gain by simply changing the lookahead distance and to the absence of noisy derivative terms.

This section describes some experiments with tuning of the lookahead parameter and different waypoint paths.

A. Experiments in Simulation

Here, a simulation of the Groundbot, provided by a third party [17], is used. The simulated environment is flat and does not induce any perturbations to the Groundbot's motion. However, the temporal resolution of position updates from the simulation is only 5 Hz (per network protocol specification).

Fig. 8 shows a run where the Groundbot follows the lemniscate curve

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \frac{a\cos t}{1+\sin^2 t} \\ \frac{a\sin t\cos t}{1+\sin^2 t} \end{pmatrix}, t \in \left(\frac{-\pi}{2}, \frac{3\pi}{2}\right],$$

starting from the origin and heading to the northwest. The path includes 40 waypoints, and with a lookahead distance of $l = 1$ meter, a smooth path like this can be followed closely.

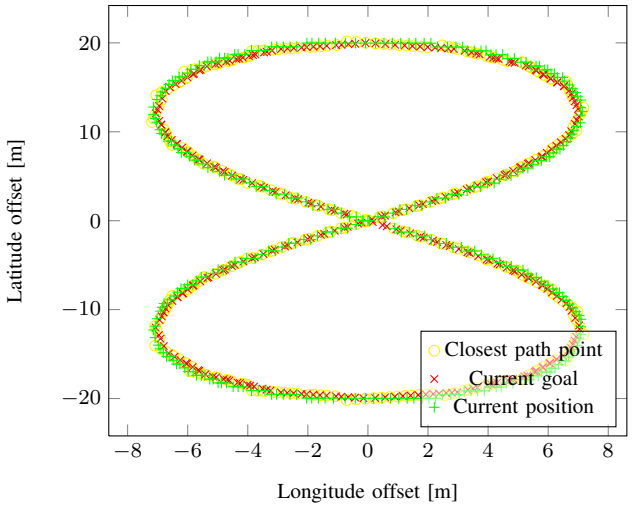


Fig. 8. Waypoint following in simulation; 1 meter lookahead distance. A detailed path with smooth curvature can be followed closely.

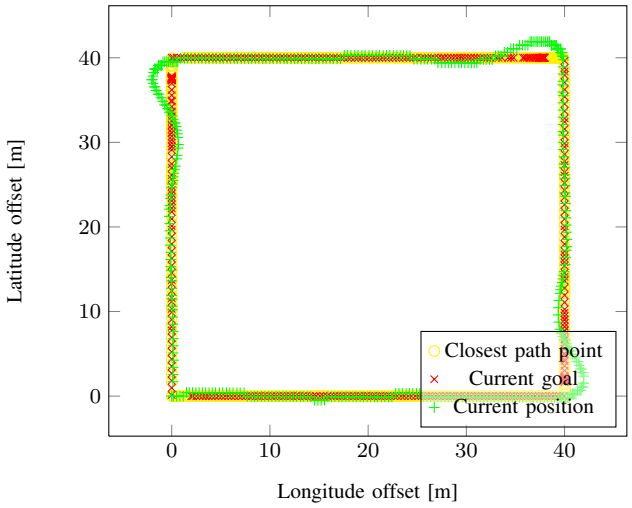


Fig. 9. Waypoint following in simulation; 2 meters lookahead distance. Coarse paths are followed less closely but robustly, even at right angled leg joints.

In Fig. 8, as well as in the following Fig. 9–11, for any given time during the path following run, a set of markers describe the current position, closest path point and current goal point, as defined in Sec. III. Since the closest path point and current goal point both lie on the line described by the current leg, together they resemble an interpolation between consecutive waypoints.

Fig. 9 shows a run where the Groundbot follows a square counter-clockwise path consisting of only four waypoints. A lookahead distance of $l = 2$ meters gives a good balance between overshoot and ringing as the robot turns onto a new leg, perpendicular to the previous. The robustness of the path following means that waypoint paths do not need to be very detailed as long as a path following error of about half the lookahead distance is acceptable. This also means that there is no need for smoothing of paths before attempting to follow them.

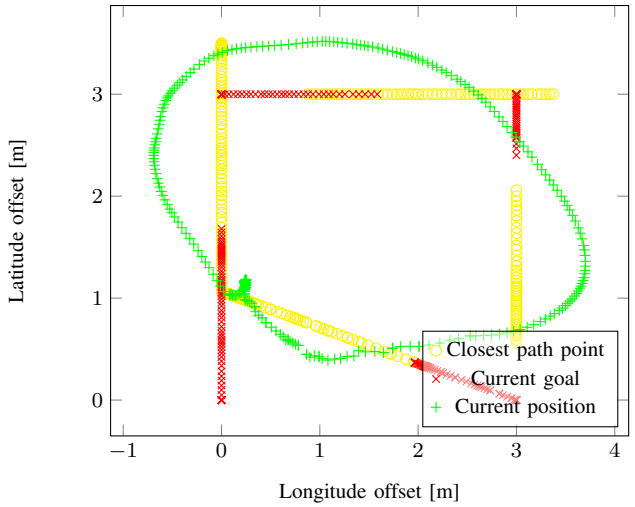


Fig. 10. Outdoor waypoint following; 2 meters lookahead distance. A small, angular path is followed with a tracking error that never exceeds half the lookahead distance.

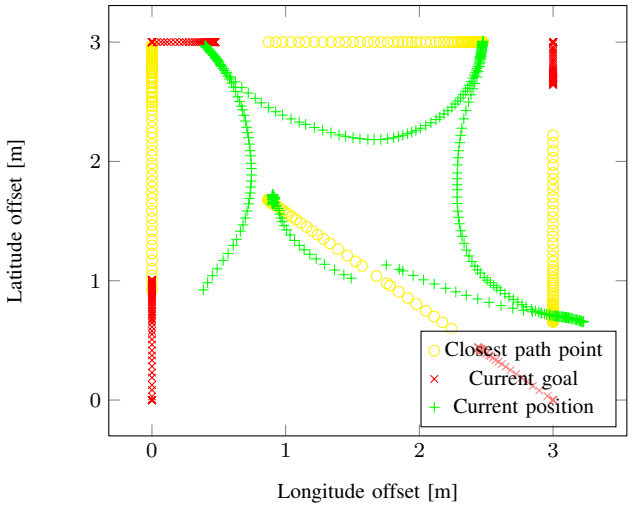


Fig. 11. Outdoor waypoint following; 2 meters lookahead distance. The path following takes advantage of the Groundbot's shape to minimize turning by opting to reverse when appropriate, as described in Sec. III.

B. Experiments in the Real World

As already outlined by [21], when using a short lookahead distance, any path tracking error will result in a more curvy, or snaking, path followed by the robot. With an uncertain position estimate (for example, a GPS reading in a forest or among buildings), a lookahead distance of $l = 1$ meter will produce an excessively curvy path followed by the Groundbot. However, using a slightly longer lookahead distance of $l = 2$ meters and a fairly liberal waypoint Pass Radius of 1 meter results in robust waypoint following behavior, even in the presence of an uncertain position estimate and an uneven surface.

Fig. 10 and 11 show two test runs of waypoint following performed on a small, tufty, horizontal field of grass. The field is surrounded by trees and flanked to the east by a two-story building. The Groundbot is configured with a lookahead distance of $l = 2$ meters and a Pass Radius of 1 meter; here, the maximum speed is also limited to 1 m/s. In both cases, the waypoint list is the same (specified in the local planar coordinate system mentioned in Sec. III): $(0, 3), (3, 3), (3, 0), (0, 0)$, a counter-clockwise square path. Remember that no smoothing of the path is done in the initial processing of the waypoint list. Also, note the different scale compared to the simulation experiments. As per previous remark, the markers for closest path point and current goal point essentially form an interpolation of the path between consecutive waypoints, and the sample points of the current position describe the path actually taken by the robot.

In Fig. 10, the Groundbot's start position is close to $(1, 0)$ and it can be seen to follow the waypoint list in a smooth, "circular" path which is an effect of the overshoot that results from a relatively long lookahead distance on such a tight waypoint path.

In Fig. 11, the start position is close to $(2, 1)$ and we can see the waypoint following algorithm take advantage of the fact that turning the Groundbot 180° is just a matter of reversing it. The result is that the robot follows the waypoint list in a "star-shaped" path.

It should be noted that the small scale of the waypoint path makes the absolute path tracking error clearly visible but still relatively small, always well under half the lookahead distance. In Fig. 11, there is an apparent discontinuity in the robot's path; this is a consequence of a current position estimate that relies heavily on GPS data and poor signal reception in the experiment area.

C. Conclusions

The Groundbot's adaptation of the Pure Pursuit algorithm provides robust waypoint following, even in the face of uncertain position estimates and bumpy surfaces that induce perturbations of the Groundbot's motion.

The choice of lookahead distance is a trade-off between stability and agility. A short lookahead distance enables tight following of high-curvature paths but exaggerates oscillations and snaking path following on uneven ground. A long lookahead distance enables stable following of straight paths but slows down rectifying of path following errors. It might be a good idea to use a variable lookahead distance depending on the waypoint path curvature and y displacement, but using a constant lookahead distance of $l = 2$ meters works well for the Groundbot.

V. FUTURE WORK

Waypoint following is of limited real-world use without the ability to detect and avoid obstacles in the path. However; the set of sensors available for this is limited for the Groundbot. Sonars would need to be placed outside of the sensor shell, and would compromise the water-tight encapsulation of the Groundbot. Laser rangers are either too big and heavy to be easily carried by the Groundbot, or have limited range; in either case, their measurements would suffer from the slight but ever-present oscillations in the motion of a spherical robot.

The obvious next step would be to use vision for obstacle detection, and using intermediary waypoints or prepending to the current waypoint list to avoid detected obstacles. The natural placement of two cameras on the Groundbot at a distance of 70 cm from each other gives a great opportunity to work with stereo vision for extracting 3D features for obstacle avoidance.

REFERENCES

[1] T. Ylikorpi and J. Suomela, "Ball-shaped Robots: A historical Overview and recent Developments at TKK," in *International Conference on Field and Service Robotics*, July 2005.

[2] M. Seeman, "Tele-operated remote inspection with a spherical robot," Licentiate Thesis, Örebro University, 2009.

[3] M. Seeman, M. Broxvall, and A. Saffiotti, "Virtual 360° panorama for remote inspection," in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, September 2007.

[4] M. Seeman, M. Broxvall, A. Saffiotti, and P. Wide, "An autonomous spherical robot for security tasks," in *Proceedings of the 2006 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety*, October 2006, pp. 51–55.

[5] A. Halme, T. Schönberg, and Y. Wang, "Motion Control of a Spherical Mobile Robot," in *4th International Workshop on Advanced Motion Control*, Tsu, Japan, 1999.

[6] A. Bicchi, A. Balluchi, D. Prattichizzo, and A. Gorelli, "Introducing the "SPHERICLE": an Experimental Testbed for Research and Teaching in Nonholonomy," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1997.

[7] R. Mukherjee, M. A. Minor, and J. T. Pukrushpan, "Simple Motion Planning Strategies for Spherobot: A Spherical Mobile Robot," in *Proceedings of the 38th Conference on Decision & Control*, Phoenix, Arizona, USA, December 1999.

[8] T. Salter, F. Michaud, D. Létourneau, D. C. Lee, and I. P. Werry, "Using proprioceptive sensors for categorizing human-robot interactions," in *HRI '07: Proceedings of the ACM/IEEE international conference on Human-robot interaction*. New York, NY, USA: ACM, 2007, pp. 105–112.

[9] J. Jones, "Inflatable robotics for planetary applications," in *In Proc. 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space (I-SAIRAS)*, June 2001.

[10] F. C. Bruhn, H. Kratz, J. Warell, C.-I. Lagerkvist, V. Kaznov, J. A. Jones, and L. Stenmark, "A preliminary design for a spherical inflatable microrover for planetary exploration," *Acta Astronautica*, vol. 63, no. 5-6, pp. 618 – 631, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/B6V1N-4SJ950N-2/2/56e68d710785d3277065821bcbe5f093

[11] S. H. Kenyon, D. Creary, D. Thi, and J. Maynard, "A small, cheap, and portable reconnaissance robot," in *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV*, E. M. Carapezza, Ed., vol. 5778. Society of Photo-Optical Instrumentation Engineers, 2005, pp. 434–443. [Online]. Available: http://link.aip.org/link/?PSI/5778/434/1

[12] B. Chemel, E. Mutschler, and H. Schempf, "Cyclops: Miniature Robotic Reconnaissance System," in *IEEE Int. Conf. on Robotics and Automation (ICRA '99)*, vol. 3, May 1999, pp. 2298–2302.

[13] F. Bruhn, K. Pauly, and V. Kaznov, "Extremely low mass spherical rovers for extreme environments and planetary exploration enabled with mems." in *Proc. of 'The 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space - iSAIRAS'*. ESA SP-603, August 2005, pp. 46–48.

[14] P. Johansson and M. Seeman, "Graphical user interface for control of a spherical robot," Master's thesis, Uppsala University, 2006.

[15] S. Bhattacharya and S. K. Agrawal, "Spherical rolling robot: Design and motion planning studies," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 835–839, December 2000.

[16] V. Kaznov and M. Seeman, "Rotundus groundbot is climbing slopes." Rotundus AB, Stockholm, Sweden, Videorecording, 2009. [Online]. Available: http://www.youtube.com/watch?v=ZcSb9F2Xn10

[17] P. Ögren, D. Anisi, D. Berglund, D. V. Dimarogonas, H. Gustavsson, L. Hedlin, J. Hedström, X. Hu, K. H. Johansson, F. Katsilieris, V. Kaznov, P. Lif, M. Lindhé, U. Nilsson, M. Persson, M. Seeman, P. Svenmarck, and J. Thunberg, "Results from the project AURES: Autonomous UGV-system for Reconnaissance and Surveillance," FOI, Stockholm, User Report FOI-R–2783–SE, 2009.

[18] V. Kaznov and M. Seeman, "Rotundus groundbot water tests." Rotundus AB, Stockholm, Sweden, Videorecording, 2009. [Online]. Available: http://www.youtube.com/watch?v=t_UIJ3d8XNI

[19] ——, "Rotundus groundbot jumps in water." Rotundus AB, Stockholm, Sweden, Videorecording, 2009. [Online]. Available: http://www.youtube.com/watch?v=jhpAzSQe3rg

[20] O. Amidi, "Integrated mobile robot control," Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-90-17, May 1990.

[21] R. C. Coulter, "Implementation of the pure pursuit path tracking algorithm," Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-92-01, January 1992.