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

Teleoperation of AZIMUT-3, an Omnidirectional Non-Holonomic Platform with Steerable Wheels

François Ferland*, Lionel Clavien*, Julien Frémy*, Dominic Létourneau*, François Michaud*‡, Michel Lauria†

*Université de Sherbrooke, Sherbrooke, Québec – Canada
Email: {firstname.lastname}@USherbrooke.ca
†University of Applied Sciences Western Switzerland (HES-SO), Geneva – Switzerland
Email: michel.lauria@hesge.ch

I. Introduction

AZIMUT-3 is an omnidirectional non-holonomic (or pseudo-omnidirectional [1]) robotic platform intended for safe human-robot interaction. In its wheeled configuration, shown in Fig. 1, AZIMUT-3 uses eight actuators for locomotion: four for propulsion and four for steering the wheels, which can rotate 180 degrees around their steering axis. Propulsion is done using standard DC brushless motors (Bayside K064050-3Y) with optical encoders (US Digital E4-300-157-HUB, 0.3 deg of resolution), capable of reaching 1.47 m/s. The platform uses steerable wheels motorized using differential elastic actuators (DEA) [2], [3], which provide compliance, safety and torque control capabilities. AZIMUT-3's hardware architecture consists of distributed modules for sensing and low-level control, communicating with each other through a 1 Mbps CAN bus. A Mini-ITX computer equipped with a 2.0 GHz Core 2 duo processor running Linux with real-time patches (RT-PREEMPT) is used on-board for high-level control modules. Nickel-metal hydride batteries provide power to the platform for up to 3 hours of autonomy. A passive vertical suspension mechanism (Rosta springs) is used to connect the wheels to AZIMUT-3's chassis, allowing them to keep contact with the ground on uneven surfaces. The platform has a 34 kg payload capacity and weights 35 kg.

Compared to a two-wheel differential steering platform, commanding AZIMUT-3's wheels independently is a complex task, whether it is done through a teleoperation interface for the eight actuators or through individual motion controllers [4]. There are two standard ways to describe the velocity state of a robot chassis. The first one is by using its twist (linear and angular velocities) and is well adapted for holonomic robots. But for non-holonomic ones, changing the velocity state in the space of instantaneously accessible velocities is constrained and a representation using the rotation around the instantaneous centre of rotation (ICR) of its motion is more adapted [5], [6]. The ICR is defined as the point in the robot's frame that instantaneously does not move in relation to the robot. For a robot using steerable wheels, this corresponds to the point where the propulsion axis of each wheel intersect. This also means that the steering and propulsion axes must be precisely coordinated to enable motion and guarantee a safe and precise motion without generating high internal forces and slippage while changing the ICR's location.

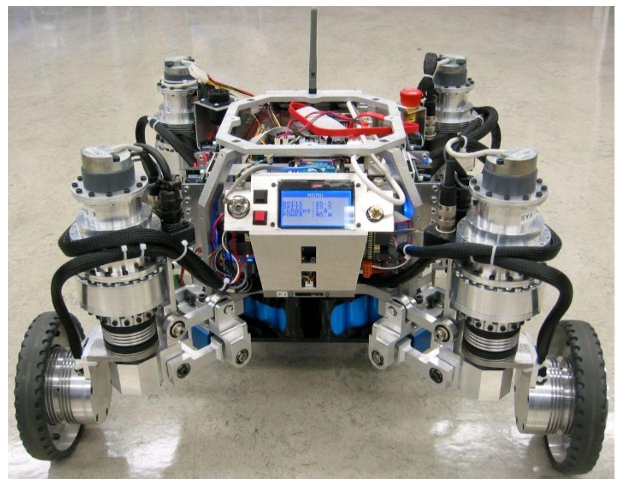

Fig. 1. AZIMUT-3 platform in its wheeled configuration

However, defining the robot's movement as a rotation around a single point is not intuitive. Twist commands are much simpler, allowing simple teleoperation or direct control commands, and most motion controllers use them as inputs. To use common trajectory control algorithms with the motion controller of AZIMUT-3, twist commands must be converted to an ICR representation. This presentation explains how we do so using the Robot Operating System (ROS) framework [7], and illustrates AZIMUT-3 controllability using a simple joystick.

II. Representation of the ICR

The ICR position in the robot frame can be parameterized using polar coordinates (ρ, γ). With this representation,

‡F. Michaud holds the Canada Research Chair in Mobile Robotics and Autonomous Intelligent Systems. Support for this work is provided by the Natural Sciences and Engineering Research Council of Canada, the Canada Research Chair program and the Canadian Foundation for Innovation.

978-1-4244-6676-4/10/$25.00 ©2010 IEEE

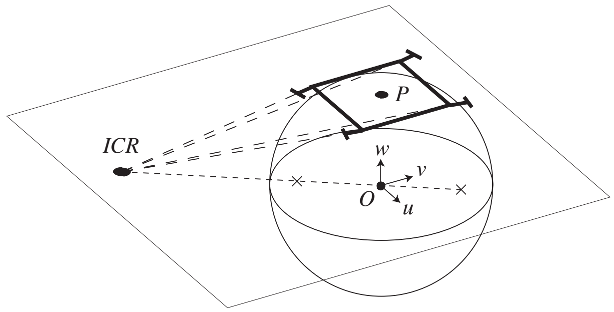

Fig. 2. ICR representation for AZIMUT-3. The chassis' center, P, is located at the zenith of the unit sphere, or $(0;0;1)$. The crosses represent the antipodal points which represents the ICR on the sphere.

singularities occur when the wheels are all parallel, making the ICR tend toward infinity [1]. An alternative is to represent the ICR by its inverse gnomonic projection [8] on a unit sphere tangent to the robot's frame at P, the center of the chassis. This is done by tracing a line between the ICR in the robot's frame and the sphere's centre. This produces a pair of antipodal points on the sphere's surface, as shown on Fig. 2. With this representation, an ICR at $\rho \to \infty$ is projected onto the sphere's equator. Going from one near-infinite position to another (e.g., when going as fast as possible from a slight left turn to a slight right turn) simply corresponds to an ICR moving near the equator of the sphere. An advantage of this representation is that all ICR can be represented with finite numbers.

Converting from a twist representation $\vec{t} = (\dot{x};\dot{y};\dot{\theta})$ to an ICR representation can be done using (1) and (2), with $\vec{\lambda}$ defining the ICR position on the sphere and μ the spin around it.

$$\vec{\lambda} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \frac{1}{\sqrt{\dot{x}^2 + \dot{y}^2 + \dot{\theta}^2}} \begin{pmatrix} -\dot{y} \\ \dot{x} \\ \dot{\theta} \end{pmatrix} \tag{1}$$

$$\mu = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{\theta}^2} \tag{2}$$

(1) describes one of the antipodal points. Note that $(\lambda;\mu)$ and $(-\lambda;-\mu)$ represent the same motion. From these equations, we can define a full ICR-based command as $\vec{\eta} = (\vec{\lambda};\mu)$. Unlike $\vec{t}$, $\vec{\eta}$ can also describe wheel configuration changes without actually moving the chassis (by setting $\mu = 0$). This means that by using a single control paradigm, the wheels can be reoriented before applying any speed to the robot, which can be useful in tight manoeuvring situations. Doing so is not possible with a pure twist-based control, since this command representation doesn't allow wheel steering without applying velocity to the chassis.

III. Integration with ROS

To use AZIMUT-3's motion controller with ROS, two nodes (processes) have been implemented. First, a converter node takes twist commands as input and produces $\vec{\eta}$ commands. The actual motion controller node only takes $\vec{\eta}$ as input. However, $\vec{\eta}$ production isn't exclusive to the converter node, i.e., both command modalities are available to ROS nodes. Because ROS packages usually send twist commands, it is therefore straightforward to use them to control AZIMUT-3. For instance, the motion capabilities demonstrated on the video are generated through a simple joystick interface developed with teleoperation ROS nodes. Twist commands are sent with the remote joystick using both analog sticks. The ICR motion controller changes the wheels' orientation automatically to best suit the twist command in accordance with the mechanical steering range constraints [4]: if a change in wheel orientation is required due to the discontinuities created by the limited steering range, the ICR motion controller stops the robot's motion to reorient the wheels and continue with the new command.

An interface has also been implemented to provide visual cues in *rviz* (ROS visualization tool), for real-time display of the commanded ICR position, both in the robot's frame and the sphere. This feature is shown in the video. We begin with the effect of various twist commands on the movement of the ICR on the sphere. Then, we show the visualization of an ICR fixed in front of the robot's chassis.

Overall, the flexibility and the numerous tools available with ROS helped us focus on the ICR motion controller itself. We are currently investigating the use of the path planning packages found in the full ROS navigation stack. Future work will focus on incremental integration of higher level functions (e.g., SLAM, vision, artificial audition, autonomous decisional architecture) on an interactive humanoid robot [9].

References

[1] C. P. Connette, A. Pott, M. Hagele, and A. Verl, "Control of an pseudo-omnidirectional, non-holonomic, mobile robot based on an ICM representation in spherical coordinates," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 9-11 Dec 2008, pp. 4976–4983.

[2] M. Lauria, F. Michaud, M.-A. Legault, D. Létourneau, P. Rétornaz, I. Nadeau, P. Lepage, Y. Morin, F. Gagnon, P. Giguère *et al.*, "Elastic locomotion of a four steered mobile robot," in *Proceedings of the 2008 IEEE International Conference on Intelligent Robots and Systems*, 2008, pp. 2721–2722.

[3] M. Lauria, M.-A. Legault, M.-A. Lavoie, and F. Michaud, "Differential elastic actuator for robotic interaction tasks," in *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 3606–3611.

[4] M. Lauria, I. Nadeau, P. Lepage, Y. Morin, P. Giguère, F. Gagnon, D. Létourneau, and F. Michaud, "Design and control of a four steered wheeled mobile robot," in *Proceedings of the 32nd Annual Conference of the IEEE Industrial Electronics*, 7-10 Nov 2006, pp. 4020–4025.

[5] G. Campion, G. Bastin, and B. d'Andréa-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.

[6] L. Clavien, M. Lauria, and F. Michaud, "Instantaneous centre of rotation estimation of an omnidirectional mobile robot," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (to appear)*, 2010.

[7] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source robot operating system," in *Open-Source Software workshop of the International Conference on Robotics and Automation*, 2009.

[8] J. P. Snyder, *Map Projections - A Working Manual.* United States Government Printing Office, Washington, 1987.

[9] F. Michaud, D. Létourneau, E. Beaudry, M. Fréchette, F. Kabanza, and M. Lauria, "Iterative design of advanced mobile robots," in *International Journal of Computing and Information Technology, Special Issue on Advanced Mobile Robotics*, 2009, pp. 4, 1–16.