

# Heavy-Duty Omni-Directional Mecanum-Wheeled Robot for Autonomous Navigation

## System Development and Simulation Realization

Li Xie

Department of  
Mechanical Engineering  
University of Auckland  
Auckland  
New Zealand  
lxie021@aucklanduni.ac.nz

Christian Scheifele

Institute for Control  
Engineering of Machine Tools  
and Manufacturing Units  
University of Stuttgart  
Stuttgart  
Germany  
cs@christian-scheifele.de

Weiliang Xu

Department of  
Mechanical Engineering  
University of Auckland  
Auckland  
New Zealand  
p.xu@auckland.ac.nz

Karl A. Stol

Department of  
Mechanical Engineering  
University of Auckland  
Auckland  
New Zealand  
k.stol@auckland.ac.nz

**Abstract**—A Mecanum-wheeled robot benefits from great omni-direction maneuverability. However it suffers from random slippage and high-speed vibration, which creates electric power safety, uncertain position errors and energy waste problems for heavy-duty tasks. A lack of Mecanum research on heavy-duty autonomous navigation demands a robot platform to conduct experiments in the future. This paper introduces AuckBot, a heavy-duty omni-directional Mecanum robot platform developed at the University of Auckland, including its hardware overview, the control system architecture and the simulation design. In particular the control system, synergistically combining the Beckhoff system as the Controller-PC to serve low-level motion execution and ROS as the Navigation-PC to accomplish high-level intelligent navigation tasks, is developed. In addition, a computer virtual simulation based on ISG-virtuos for virtual AuckBot has been validated. The present status and future work of AuckBot are described at the end.

**Keywords**—Mecanum; omni-direction; heavy-duty; AuckBot; control system architecture; simulation

### I. INTRODUCTION

Omni-directional mobility is an outstanding ability for the mobile robots to conveniently transport in the congested, confined or highly dynamic environments. An omni-directional wheel provides a complete maneuverability that a conventional wheel does not [1]. The Mecanum wheel, as an omni-directional wheel, was invented in 1975 by Ilon in Sweden [2]. The Mecanum wheel is described as an active hub with a designed number of passive rollers attached around its circumference at a conventional  $45^\circ$  angle as shown in Fig. 1. The geometry of the rollers is specially curved so that the side shape of the Mecanum wheel keeps circular. While one Mecanum wheel is rotating, the angled roller, which is in contact with the ground, provides one angled drive force to the Mecanum robot. The sum drive force with any desired directions on the Mecanum robot can be produced by combining all the angled drive forces from the Mecanum wheels. Thus, the Mecanum robot is capable of omni-directional motion without wheel steering by controlling

different velocity combinations of the Mecanum wheels. The holonomic mobile robots based on Mecanum wheels have been applied in AGV for many decades, mainly on account of its great omni-direction maneuverability, which prioritizes itself over conventional locomotion solutions in confined working environment such as warehouses. However, its inherent problems, including random wheel slippage and high-speed vibration, cause uncertain position errors and very low efficiency. These disadvantages hinder the Mecanum wheel's further autonomous application. Mecanum wheel mobile vehicles are expected to meet the challenging AGV criteria, in particular, but not limited to high speed, high accuracy and high stability.



Fig. 1. The heavy-duty Mecanum wheel assembled in AuckBot

A lack of heavy-duty Mecanum robot research on autonomous navigation demands a robot platform like AuckBot to conduct experiments. AuckBot is a robot platform designed and manufactured at the University of Auckland. This heavy-duty omni-directional four-Mecanum mobile robot will be utilized as a testing platform in an industrial environment for autonomous navigation research of the future at the University of Auckland. Our intelligent navigation methods developed in [3-5] will be incorporated into the AuckBot control architecture to be developed in this study. The rest of the paper is organized as follows. Section 2 describes the characteristics overview of the Mecanum wheel including

inherent disadvantages, mathematical model, position rectification, vibration, energy efficiency and redundant motion system. In section 3, related work and research motivation of AuckBot is introduced. Section 4-6 describes the hardware overview, control system architecture and simulation design. Discussion and future work are shown in Section 7. Finally this paper is concluded in Section 8.

## II. MECANUM WHEEL

### A. Inherent Disadvantages

A working Mecanum wheel involves a very complicated mechanism. Each Mecanum wheel has only one roller contact most of the time, but occasionally two roller contacts with the ground. Due to the roller's placement angle and curved surface mentioned earlier, a point contact or a tiny area contact exists between the ground and roller. The tiny contact causes the first typical Mecanum wheel's fundamental problem - wheel slippage. Additionally, the contact point continuously moves from one side of the roller to the other side and then discretely jumps to the next coming roller. The same procedure repeats. Also, the radius of the wheel changes within a 5% range as the roller contact point is moving [6]. When the Mecanum wheel is rotating at high speed the continuously and discretely moving contact causes another fundamental problem - wheel vibration. Both problems are on account of the rolling rollers. This is why both problems happen more obviously when the same sides of the wheels (1-4 or 2-3 wheels in Fig. 4) are rotating in opposite directions, e.g. moving sidewise. Roller rolling is more engaged in such cases than in a moving forward case, in which the same sides of the wheels are rotating in the same directions.

### B. Four-Mecanum Robot Mathematical Model

The moving and tiny contact from the rolling roller leads to a complex mechanism involved in the Mecanum robot model. Based on many simplifications such as free-rolling roller, fixed central contact point and so on, researchers derived the Mecanum wheel's kinematics and dynamics model. Equation (1) is the forward kinematics equation, which is most commonly used kinematics model for Four-Mecanum robots and from which many control equations are derived [7-11].

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \frac{R}{4} \begin{bmatrix} -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ l_1+l_2 & l_1+l_2 & l_1+l_2 & l_1+l_2 \end{bmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \end{pmatrix} \quad (1)$$

Where  $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4$  are actuation rotations of all Mecanum wheels and  $\dot{x}, \dot{y}, \dot{z}$  are the velocities of the Mecanum robot in the three dimensions. The rest of the symbols are the geometrical parameters of the Mecanum robot. The model errors from the simplifications are dealt with as control uncertainties by control methods. In recent years, some researchers have analyzed the position error sources such as slippage, bearing and/or axle friction and point contact friction. This is to refine the kinematics model [12] and the parameter error source, such as moving roller contacts, and to refine the dynamics model [6] by considering fewer simplifications in the complex mechanism of the Mecanum wheel. More

improvements in the mathematical models for the Mecanum wheel can be expected in the future.

### C. Position Rectification Control

The random wheel slippage introduces a severe position error in the optical encoder dead-reckoning method, which is very commonly applied in the position control of wheel mobile robots. An accurate position control is very necessary for an autonomous mobile robot. There is some research work on the position control of Mecanum robots. Most of them solve the position error by applying a visual sensor to detect the absolute position to correct the position error [9,10,13]. ITM University proposed a successful localization technique using an encoder, gyro and accelerometer for the Mecanum robot at very low speed [14].

### D. Vibration

Vibration is an inherent disadvantage of the Mecanum wheel, especially at high speed. However, not much research focuses on the high-speed vibration of the Mecanum wheel yet. There are a few comments about Mecanum wheel vibration in the research work including vibration excitation mechanism analysis from the kinematics model, vibration avoidance by improving roller geometry, and vibration reduction by robot suspension system [15-17].

### E. Energy Efficiency

The Mecanum wheel trades off maneuverability against energy efficiency. The drive force due to the motor torque is divided into effective wheel drive force and ineffective roller rolling force by the angled roller. The angled effective wheel drive force contributes to the complete maneuverability of the Mecanum robot. The ineffective roller rolling force is wasted because roller rolling does not contribute to any movement at all. Diegel proposed to lock the rollers for forward/backward motion. In addition, there is a dynamically adjustable roller angle to best fit the required motion direction [18]. This actually increases the gaps between rollers when they are 45° or 135° degrees. The increased gaps between rollers cause more obvious vibration or even affect the functionality of the Mecanum wheel. So far such improvement in the Mecanum wheel has not been used yet.

### F. Redundant Motion System

A Mecanum wheel has 3 degrees-of-freedom (DOF), which includes wheel rotation, roller rotation and rotational slip about the yaw axis passing through the contact point [19]. According to the forward kinematics equation (1), the controllable DOF (four input independent motor rotations  $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4$ ) is more than the total motion DOF (three output robot velocities  $\dot{x}, \dot{y}, \dot{z}$ ). The Four-Mecanum robot is also a redundant motion system. Many different combinations of the four motor rotations are available to perform one same motion task. From a starting position to a goal position with a desired posture, there are too many possibilities of motion plans. The Mecanum wheel itself has unsatisfying energy efficiency. A trajectory optimization is necessary but has challenges because the Mecanum holonomic robot provides many motion forms.

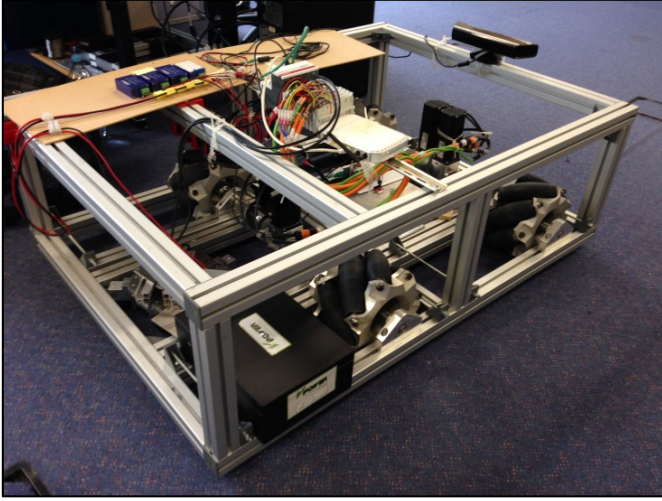


Fig. 2. AuckBot: heavy-duty omni-directional Mecanum-wheeled robot

### III. RELATED WORK AND RESEARCH MOTIVATION

There are many practical Mecanum applications in various fields [19], e.g. NASA OmniBot, Navy Mecanum wheel vehicle and MarsCruiserOne in the military field; Airtrax forklifts in the industrial fields; OMNI, the CIIPS wheelchair and iRW in the medical fields; and Uranus, Omni-1, Omni-2 and MEGAN in the academic field [20-25]. Even though there are heavy-duty Mecanum applications in the military and industrial fields, the academic field is lacking in heavy-duty Mecanum robot research especially on autonomous navigation. Many research interests in heavy-duty Mecanum robots are to be explored. The high-speed vibration of the Mecanum wheel becomes severer in the heavy-duty case. The heavy-duty Mecanum wheel vibration may create unacceptable noises and even affect the stability of the motor currents and the power system, which relates to safety issues. Heavy-duty tasks consume huge amounts of energy. Besides, the Mecanum wheel trades off maneuverability against energy efficiency. It is very necessary to optimize the trajectory planning and trajectory following for heavy-duty cases to achieve energy optimization. Especially for the holonomic and redundant Mecanum system, the trajectory planning and following have way more options than the conventional wheel system. High working speed and stability are also very important for autonomous application. AuckBot in Fig. 2 is designed and manufactured for the research in the future.

### IV. HARDWARE OVERVIEW

#### A. Mechanical Design Overview

The Mecanum wheel in AuckBot has a diameter of 0.2m and a width of 0.15m, with seven rollers placed at a 45° angle around the wheel circumference. Each Mecanum wheel is mounted by a wheel hub on a motor shaft, which is supported by two SKF 30304JQ2 tapered roller bearings on both sides as shown in Fig. 3. Both wheel hubs and motor shafts are made of stainless steel. Bearing houses are made of aluminum. Four Mecanum wheels are assembled inside a 1.2m×0.9m×0.4m aluminum chassis. The total weight of the robot is around

100kg and it has a 300kg loading ability, which is mostly limited by the Mecanum wheel's loading ability.

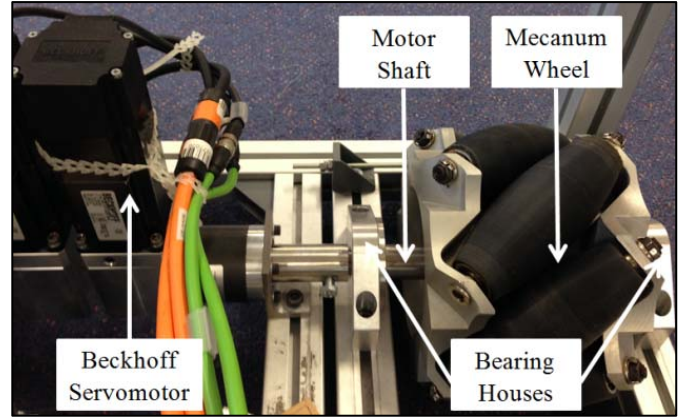


Fig. 3. Mechanical design of AuckBot

#### B. Motion Control System Overview

The whole packages of motion control system are based on the Beckhoff compact technology. According to the schematic diagram in Fig. 4, four AM3121 servomotors with AG2250 gear units are linked to the CX5020 embedded PC via EL7201 motor drive terminals. The embedded PC, which acts as the controller in the system, controls the motor axis positioning.

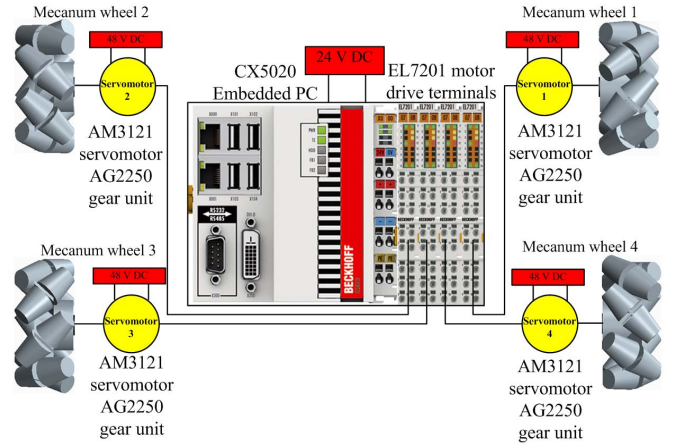


Fig. 4. Beckhoff motion control system schematics

#### C. Existing Sensors Overview

The optical encoder integrated in the servomotor provides the rotation information of the servomotor shaft. Two spherical ball encoders provide the robot pose information to generate the robot's localization. Both sensory data are sent to the embedded PC and then fused to proper information. Microsoft Kinect collects environment information to the Robot Operating System (ROS) for navigation purpose.



## V. CONTROL SYSTEM ARCHITECTURE

The motion control system is based on the Beckhoff technology. However, the provided system is not computationally capable of autonomous navigation tasks including perception, localization, cognition and motion control. A control system combining the Beckhoff motion control system and an autonomous navigation system is desired. ROS, on which leading research in the robotics field is working and whose active community offers free open-source code, is chosen as the intelligent robot framework responsible for autonomous navigation tasks. The combination of Beckhoff motion control system and the ROS robot framework contributes to upgrading AuckBot to state-of-the-art.

### A. Beckhoff Controller-PC

The Beckhoff embedded PC acts as the Controller-PC in the combined system and governs the locomotion of the robot platform system. The PC-based industrial automation technology from Beckhoff certainly provides accurate and stable motor controllability. Real-time and precise motor tasks are easily achieved. The Controller-PC runs a TwinCAT PLC program that applies NC-PTP, a numerical controller point-to-point function to operate the motor axis positioning. The TwinCAT NC motion control PLC library provides a function block to access NC-PTP. However, it is beyond the functionality of the provided Beckhoff system to perform advanced autonomous navigation tasks such as visual perception, SLAM and so on. It is also both time-consuming and costly to create a new navigation system based on the advanced Beckhoff system.

### B. ROS Navigation-PC

To achieve autonomous navigation, a navigation system has to accomplish environment perception, localization, obstacle avoidance, trajectory planning and trajectory following. So in a Navigation-PC, a high-level intelligent navigation system, which deals with navigation tasks and sends motion command to the Beckhoff motion control system to execute, in parallel with Controller-PC is a wise solution. ROS, which offers collections of libraries, tools and packages shared from its community, provides convenient, advanced and robust robot framework to AuckBot. A laptop PC, in which Linux Ubuntu OS is installed, runs ROS. Within the ROS structured communication layer, data are exchanged between different ROS *nodes*, which can *publish* to *topic* to send information and can *subscribe* to *topic* to receive information [26].

### C. 2D navigation stack

The ROS community offers an open-source 2D navigation stack, which has been developed for autonomous navigation purposes. The frame of the 2D navigation stack has been established. In order to apply the 2D navigation stack to a specific robot, configuration requires to be adjusted by the robot's attribution and dynamics by creating proper platform

specific nodes including odometry information, base controller and sensor sources to publish and subscribe to the proper topics.

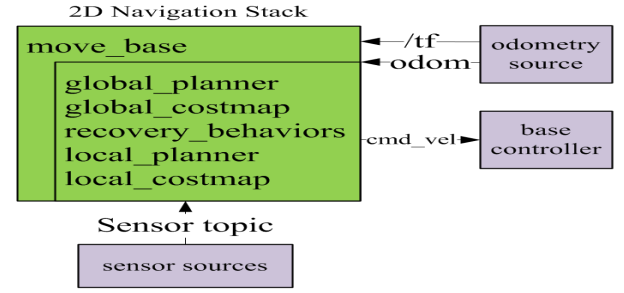


Fig. 5. Odometry, motion command and sensory information flow

In Fig. 5, the information flow involved with 2D navigation stack is shown. An odometry information node is created to publish odometry data on the odometry topic (/odom) in the 2D navigation stack. In the meanwhile, the odometry information node publishes transformed odometry data to transform odometry frame (start position of the robot) to robot frame (present position of the robot) using tf. The 2D navigation stack publishes motion commands using a geometry\_msgs/Twist message in robot frame on the cmd\_vel topic. This message has to be transformed into wheel coordinates and sent to TwinCAT. The environment data are received from the exteroceptive sensor Microsoft Kinect and are published to sensor topics via the Kinect connection node.

### D. Task allocation

The autonomous navigation performance can be realized by accomplishing both Controller-PC tasks and Navigation-PC tasks. Linking Controller-PC and Navigation-PC, control system architecture is shown in the schematics in Fig. 6.

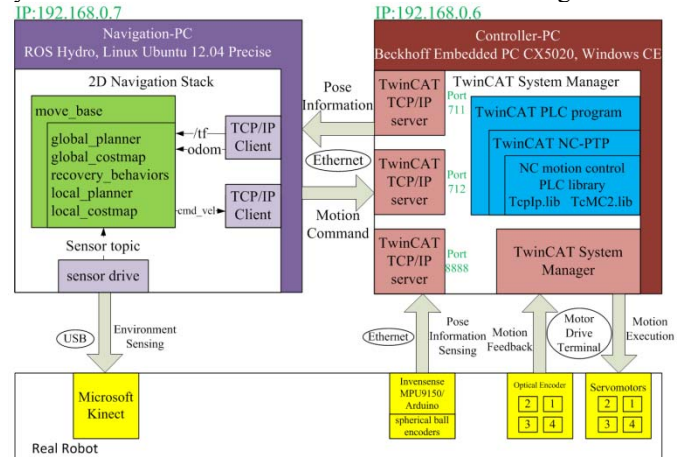


Fig. 6. Control system architecture

The Beckhoff system serves the robot platform as low-level motion control system. The Controller-PC is responsible for collecting and fusing the sensory information from the proprioceptive sensors. Optical encoders integrated in the servomotors collect motor shaft velocities as feedback signal to close-loop control motor motions in the TwinCAT NC-PTP. The Controller-PC is also responsible for collecting the

odometry data from the exteroceptive sensor, the spherical ball encoders and sending to the Navigation-PC to fuse. In addition, the Controller-PC receives the motor motion commands from Navigation-PC and executes the commands in the TwinCAT PLC program by accessing the MC MoveVelocity function block in TwinCAT PLC library.

ROS serves the robot platform as a high-level intelligent navigation system. First, perception must be done to be aware of the surrounding environment. The Navigation-PC collects and fuses the exteroceptive sensory information. The Microsoft Kinect collects environment visual data to the Navigation-PC to be analyzed. The Navigation-PC receives the pose sensory data from the Controller-PC and fuses to generate robot pose estimation. Then complex navigation tasks such as localization, path planning, path following and obstacle avoidance are to be done based on the environment information. The navigation algorithm inside ROS calculates the motion planning in term of robot dynamics and then transforms to motor motion command, ready to be sent to the Controller-PC to execute.

### E. Data communication via TCP/IP

Navigation-PC with ROS has a wide range of connectivity options, while the Controller-PC with TwinCAT limits the options. TCP/IP is the connectivity solution in AuckBot. The TwinCAT TCP/IP Server offers convenient realization of TCP/IP clients or server application in the Controller-PC TwinCAT PLC. The TCP/IP client is easily integrated into a ROS node on Navigation-PC with the help of Berkeley sockets API. Two TwinCAT TCP/IP servers addressed with the Controller-PC IP address and port numbers are created by the SPS library `TcpIp.Lib`, which permits full access to TCP and UDP ports. The servers are ready for sending odometry data to the Navigation-PC and receiving motion commands from the Navigation-PC. The odometry data are received from the Controller-PC. In the case of AuckBot, the odometry data shown in Fig. 7 consist of the  $x$ ,  $y$  and  $\theta$  position of the robot in relation to the world odometry frame and the velocity in  $x$ ,  $y$  and  $\theta$  direction in relation to the robot frame. The motion commands using a `geometry_msgs/Twist` consist of a linear velocity in the  $x$ - and  $y$ -direction and an angular velocity in the  $z$ -direction.

PosX	PosY	Pos Theta	VelX	VelY	Vel Theta	sign1	sign2	sign3	sign4	sign5	sign6
[0.5]	[6.11]	[12.17]	[18.23]	[24.29]	[30.35]	[36]	[37]	[38]	[39]	[40]	[41]

Fig. 7. The odometry data in TCP/IP communication

## VI. SIMULATION DESIGN

A computer virtual simulation is important in engineering design. First, safety should be guaranteed in the simulation before conducting experiments especially on the heavy-duty AuckBot. A reduction of experiment time can be expected in simulation. In addition, an arbitrary testing environment is easily set up in the simulation. The behavior of the real robot AuckBot is simulated to be a virtual robot in the simulation system, which offers virtual experiments for the virtual robot to be tested. The simulation system is supported by ISG-

virtuos because this is able to fully integrate into the TwinCAT system manager. ISG-virtuos can be installed and optionally integrated into the TwinCAT. ISG-virtuos software consists of Virtuos M for simulation model, Virtuos S for simulation solving and Virtuos V for simulation visualization.

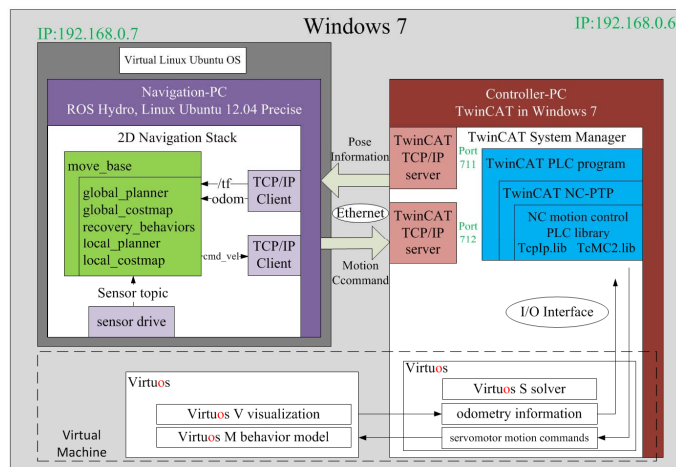


Fig. 8. Simulation system architecture in one-PC solution

The simulation system in AuckBot has three function aspects: behavior model, visualization and monitoring. A behavior model is created based on the forward kinematic equation in the Virtuos M. To set up the simulation, the virtual robot, rather than the real robot, is connected to the TwinCAT PLC via the TwinCAT system manager. One-PC solution of simulation system architecture is shown in Fig. 8. The motion command, which was sent to servomotors, is now sent to the ISG-virtuos. Given the motion command from the Controller-PC, ISG-virtuos can calculate the robot's dynamics and odometry information in the Virtuos S. The robot's dynamics are sent to Virtuos V to generate the visual motion of the virtual robot. Odometry information is returned to the Controller-PC. During the procedure, all the values in the simulation are available to be displayed in Virtuos M.

## VII. DISCUSSION AND FUTURE WORK

Currently AuckBot is able to perform continuous-curvature omni-directional motion very smoothly. The total weight of AuckBot is 100kg and it is able to carry a 300kg load. Simulation based on ISG-virtuos has been realized. The simulation system has been working in the one-PC solution. Motion behavior of the virtual robot in any designed virtual environment can be simulated by giving any motion commands from TwinCAT and ROS, with the visualization of the movement. The control system architecture, combining the Beckhoff control system and ROS framework, has been validated. The tasks allocated for both the Controller-PC and Navigation-PC have been verified successfully in the simulation system.

The research challenges of the Mecanum robot are caused by the random wheel slippage and the high-speed wheel vibration. The near future research will focus on issues due to the slippage. Firstly the random wheel slippage causes noticeable position error in the commonly used optical

encoder dead-reckoning method. An absolute-position odometry system by utilizing IMU, gyroscope, vision sensor, laser sensor, optical flow sensor and/or GPS will be properly designed and applied to implement the precise position rectification for the robot platform. Secondly the low energy efficiency caused by Mecanum wheel slippage is the inherent problem of the Mecanum wheel design. The Mecanum wheel trades off maneuverability against energy efficiency. However, the Mecanum robot is a holonomic and redundant system. Moving from a start position with any orientation to a goal position with any orientation, it has many more motion execution options than the conventional wheel system. Research interest to optimize the motion execution can achieve energy optimization. Energy cost function of the Mecanum robot in terms of both electric consumption energy and kinetic energy will be quantified. Then the cost function will be applied to calculate energy-optimal motion execution.

After that, AuckBot is ready to carry out the SLAM research and autonomous navigation research. Robust odometry system and precise position control are essential for the upcoming SLAM research. Energy-optimal motion execution will be initial step of the upcoming energy-optimal autonomous navigation research, including trajectory planning, trajectory following, obstacle avoidance and even behavior-based navigation, all based on the energy-optimal criterion.

## VIII. CONCLUSION

The first introduction of AuckBot with hardware overview, control system architecture and simulation design is presented in this paper. The Beckhoff-based locomotion hardware performs a smooth motion. The integration of ROS with the Beckhoff system in a synergistic way upgrades AuckBot to an intelligent autonomous robot platform that is state-of-the-art and ready for research in the future. The one-PC solution of virtual simulation based on ISG-virtuos has been validated. Some future work needs to be done to have AuckBot ready to meet the research criteria. More research such as SLAM research, energy-optimization navigation and behavior-based navigation is to be developed and completed afterwards.

## REFERENCES

- [1] R. Siegwart, I.R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. 2nd ed, The MIT Press; Cambridge, Massachusetts; London, England, ISBN 978-0-262-01535-6, 2004.
- [2] B. E. Ilon, "Wheels for a course stable self-propelling vehicle movable in any desired direction on the ground or some other base," US Patent 3876255, Apr. 8, 1975.
- [3] W. L. Xu and S. K. Tso, "Real-time self-reaction of a mobile robot in unstructured environment through fuzzy reasoning," *Eng. Applicat. Artif. Intell.*, vol. 9, no. 5, pp. 475-485, 1996.
- [4] W. L. Xu, S. K. Tso, and Y. H. Fung, "Fuzzy reactive control of a mobile robot incorporating a real/virtual target switching strategy," *Robot. Auton. Syst.*, vol. 23, no. 3, pp. 171-186, 1998.
- [5] W.L. Xu, "A virtual target approach for resolving the limit cycle problem in navigation of a fuzzy behavior-based mobile robot," *Robotics and Autonomous Systems*, vol. 30, pp. 315-324, 2000.
- [6] M. De Villiers, and N.S. Tlale, "Development of a control model for a four wheel Mecanum vehicle," *Journal of Dynamic Systems Measurement and Control-transactions of the ASME*, vol. 134, pp. 1-6, 2012.
- [7] P. F. Muir, and C.P. Neuman, "Kinematic modeling for feedback control of an omnidirectional wheeled mobile robot," *Robotics and Automation. Proceedings*, vol. 4, pp. 1772-1778, 1987.
- [8] N. Tlale and M.D. Villiers, "Kinematics and dynamics modeling of a Mecanum wheeled mobile platform," *15th International Conference on Mechatronics and Machine Vision in Practice (M2VIP08)*, pp. 657-662, 2008.
- [9] P. Viboonchaicheep, A. Shimada, and Y. Kosaka, "Position rectification control for Mecanum wheeled omni-directional vehicles," *Proceedings Of the 1998 IEEE International Conference on Vehicles & Automation*, pp. 854-859, 2003.
- [10] A. Shimada, S. Yajima, P. Viboonchaicheep, and K. Samura, "Mecanum-wheel vehicle systems based on position corrective control," *Annual Conference of IEEE Industrial Electronics Society*, pp. 2077-2082, November 2005.
- [11] S. Kumra, R. Saxena, and S. Mehta, "Navigation system for omni-directional automatic guided vehicle with Mecanum wheel," *IOSR Journal of Electrical and Electronics Engineering (IOSRJEET) ISSN: 2278-1676* vol. 2, issue 3, pp 35-39, 2012.
- [12] K.L. Han, H. Kim, and J. S. Lee "The sources of position errors of omnidirectional mobile robot with Mecanum wheel," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 581-586, 2010.
- [13] J. A. Cooney, W. L. Xu, and G. Bright, "Visual dead-reckoning for motion control of a Mecanum-wheeled mobile robot," *Mechatronics*, vol. 14, no. 6, pp. 623-637, 2004.
- [14] J. Kim, J. Park, and S. Kim, "Inertial navigation system for omni-directional AGV with Mecanum wheel," *MIME 201*, pp. 207-273, 2011.
- [15] G. Wampfler, M. Salecker, and J. Wittenburg, "Kinematics, dynamics and control of omnidirectional vehicles with Mecanum wheels," *Mech. Struct. Mach.*, vol. 17, pp. 165-177, 1989.
- [16] A. Gfrerrer, "Geometry and kinematics of the Mecanum wheel," *Computer Aided Geometric Design*, vol. 25, no. 9, pp. 784-791, 2008.
- [17] Y. Xie, S.H. Kang, A.R. Wang, and C.Y. Yang, "Derivation and simulation on the academic profile of the Mecanum wheel," *Advanced Materials Research*, vol. 546, pp. 125-130, 2012.
- [18] O. Diegel, A. Badve, G. Bright, and J. Potgieter, "Improved Mecanum wheel design for omni-directional robots," in *Proc. 2002 Australian Conf. Robot. Autom.* Auckland, New Zealand, 2002.
- [19] F. Adăscăliței and I. Doroftei, "Practical applications for mobile robots based on Mecanum wheels - a systematic survey," *The Romanian Review Precision Mechanics, Optics and Mechatronics*, no. 40, pp. 21-29, 2011.
- [20] N. Houshang and T. Lippitt, "Omnibot mobile base for hazardous environment," in *Electrical and Computer Engineering, 1999 IEEE Canadian Conference on*, vol. 3, pp. 1357-1361, 1999.
- [21] H. Hoyer, U. Borgolte, and A. Jochheim, "The OMNI-wheelchair-state of the art," *Proceedings of Conference on Technology and Persons with Disabilities*, Los Angeles, 1999.
- [22] T.V. How, "Development of an anti-collision and navigation system for powered wheelchairs," PhD Diss, University of Toronto, 2011.
- [23] P.E. Hsu, Y.L. Hsu, and J.M. Lu, "iRW: an intelligent robotic wheelchair integrated with advanced robotic and telehealth solutions," in *1st Asia Pacific eCare and TeleCare Congress*, Hong Kong, 2011.
- [24] P. Muir and C. P. Newman, "Kinematic modeling of wheeled mobile robots," *Journal of Robotic Systems*, vol. 4, no. 2, pp. 281-340, 1987.
- [25] T. Braeunl, *Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems*, 3<sup>rd</sup>, Springer Berlin Heidelberg, ISBN-13 978-3540705338, 2008.
- [26] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. B. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.