Design of a Holonomic Ball Drive Robot for Greenhouses

B.J.M. Muilenburg (5499615), B.A. Kalkman (5646731),
H.J. Wassing (5588545) & M. Omara (5248450)

Technische Universiteit Delft
Faculteit ME
Mekelweg 2, 2628 CD Delft

Abstract

The TU Delft botanical garden is creating a smart-greenhouse. The desire is to eventually operate this greenhouse autonomously. For this, robots need to be created that have great manoeuvrability, such as holonomic drive robots. Many designs of holonomic drive robots have been created over the years, some including omni- or mecanum wheels, which have vibrations. Other designs might make use of a swerve system, where wheels can rotate up to 180 degrees. These designs however, are not as efficient as the wheels need to turn before moving holonomically, which results in a delay. In September 2014, G. Runge, G. Borchert and A. Raatz published an article about a possible holonomic ball drive robot, without omni- or mecanum wheels [4]. In our article, we present a holonomic ball drive robot that has implemented and improved their ideas. The first complete prototype was built from PLA and 3D-printed. The final design can be controlled with a PS4 controller. The robot has the ability to perform instant translations and a pure rotation. To further expand on this prototype, higher-quality ball rollers could be used, as could researching the difference between rubber and TPU balls. To autonomously control this robot, GPS and a lidar, could be integrated into the system.

1 Introduction

This research has been focused on applications in greenhouses. The Technical University of Delft recently started a project for autonomous greenhouses. As these greenhouses will not depend on any humans, robots will be doing all the work. This research is about designing a robot for this greenhouse, that can drive holonomicaly. The main focus was to improve the mechanical aspects of the earlier design from G. Runge, G. Borchert and A. Raatz, in order to see whether this holonomic technology would work once implemented into the smart-greenhouse. The robot will be used to perform tasks involved with the care of plants, while driving on hard, griddy cement.

Conventional wheels are used on almost every vehicle or robot. With the growing need for efficiency, holonomic drive robots could offer an improvement relative to conventional driven robots. Holonomic drive offers robots the ability to move instantaneously in any direction from any configuration, without needing to change its orientation. They simply operate as one compliant body, because they don't need an arch to turn, this creates an opportunity to further minimise the space needed to operate. Current designs feature either omni wheels or mecanum wheels. These are a step in the right direction, but they induce lots of vibrations. Vibrations can cause imprecise results in movements and sensor readings, and can cause the wheels to break. A holonomic ball drive could offer the solution, with much less vibrations, because of their continuous contact surface.

In an earlier paper about a potential design for holonomic ball drive robots, G. Runge, G. Borchert and A. Raatz (2014) proposed a theoretical design without the use of any omni- or mecanum wheels [4]. Their design, shown in figure 1, uses four normal wheels and a ball roller to constrain and drive the ball.

The research study behind our article is focused on answering the questions: what is the optimal configuration for a ball drive mechanism for use in greenhouses? What materials and parts should be used to get the best performance? Will a ball drive mechanism outperform omni- and mecanum wheels as the best holonomic drive mechanism?

The article will show the theory behind the research, before considering the used methodology.

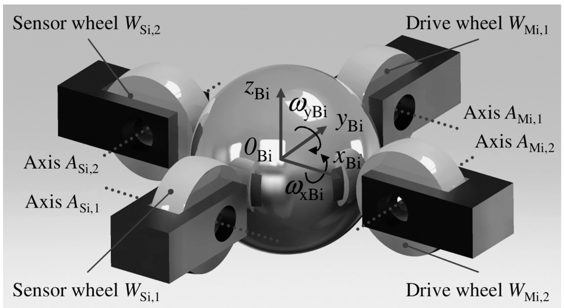

Figure 1. A ball drive mechanism from 2014, figure adapted from [4]

The results will then be shown and also discussed. Finally, conclusions will be drawn and recommendations will be done for further research.

This design study was conducted as part of the Bachelor End Project for mechanical engineering students at Delft University of Technology.

2 Literature research

Actuating the balls using magnetism was researched, and one of the most promising papers was written by (Yan, Chen, Yang, & Lee, 2006). In this paper, a mechanism for constraining and driving a ball is proposed, in which permanent magnets inside the ball and activating coils around said ball are used, see figure 2 [9]. The primary issue with this design, was the insufficient understanding of magnetism. To understand this subject, too much time would be spent on research and basic theory, instead of designing for the specific problem. Another problem was the tilting capability of this prototype. The prototype could generate 360 degrees of rotation around the z-axis, but only 45 degrees around the x and y axis. For full holonomic functionalities, this is not enough, as 360 degree rotation around every axis is required.

In 1995, Hideaki patented a design for a omnidirectional drive wheel [11]. Certain aspects from the final design are inspired by this patent. In this patent, ball rollers are used to constrain any translations of the ball in x, y and z directions, while still allowing rotation around these axes. This idea has been implemented into the final design.

An important insight on conventional wheel mechanisms was found from West and Asada from 1995 [12]. A wheel configuration would need steering links to rotate the wheels about the z-axis and be able to drive holonomically. Adding steering links creates a delay and makes the path planning of the robot harder, as the wiring to the motors of the wheels cannot be rotated infinitely.

Another engineer that brought inspiration was James Bruton. Bruton has several videos on his hobby YouTube channel where he creates holonomic drive mechanisms and explains how they work. In one of his videos he built the mechanism from figure 3 [7]. Unfortunately, with this design the robot still doesn't have a continuous contact surface with the ground, which causes unwanted vibrations that could harm the reliability of the mechanism. In his video about a swerve drive mechanism, he also created a clever mechanism, where four-wheel steering is used, to change the direction of driving force [8].

Another source of non-scientific, but useful inspiration, was a video from Brick Technology, a YouTube channel were various Lego constructions are made. Here, a Lego car is built, using balls as its driving mechanism [2]. A variable constraint is used to never over-constrain the balls. This idea has also been implemented in the form of small 'set rings', which will be discussed in the design section of this paper.

From the literature research, we learned about various actuation methods for ball drive robots. Magnetic actuation was too complex, but designs from Hideaki, James Bruton, and Brick Technology helped refine our design. These insights have helped to constrain the balls in a proficient way, using ball rollers and a variable constraint.

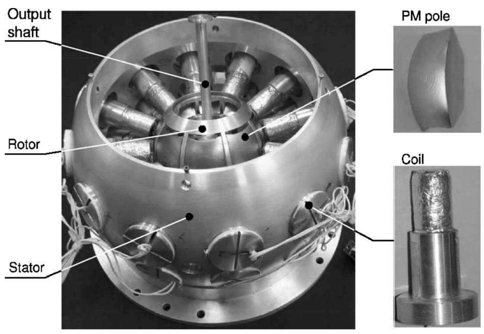

Figure 2. Prototype of a magnetic ball actuator, figure adapted from [9]

Figure 3. James Bruton's holonomic drive design, figure adapted from [7]

3 Methodology

Before designing the robot, the following list of requirements was created. The robot will be used in greenhouses, which changes what factors are important.

1. The robot must be able to drive in x and y directions without having to turn first.
2. The robot must be able to rotate about its own z-axis.
3. The robot cannot feature any mecanum or omniwheels.
4. The robot must be designed to use the minimum amount of material necessary without compromising structural integrity and functionality.
5. The robot needs to be compact enough to navigate through tight spaces within the greenhouse.
6. The robot must be designed to minimize internal friction to ensure smooth and efficient operation.
7. The robot must be portable and not attached to a wire.
8. The robot needs to exhibit minimal deviation from its path when driving in any direction.

A scientific approach was chosen to achieve the desired end result, with the design based on prior scientific literature.

A combination of three methods was used to create the final design: reference projects, calculations, and testing and tuning.

3.1 Reference Projects

For this report, reference projects were a method to build knowledge on the scope of holonomic drive robots. The proposed designs within these reference projects were then analysed and some were tested and improved for usability in our own design.

3.2 Calculations

Calculations were done to make sure certain mechanical and electrical parts met the design requirements. Using SolidWorks, Finite Element Method (FEM) analysis calculated the stiffness and strength of key parts. These simulations predicted how the robot would handle different loads and stresses, which helped improve the design for both performance and material use.

3.3 Testing and Tuning

The main method for improving the design was repeated testing, analyzing, and adjusting. This was mostly done for the mechanical parts of the robot. Full-size prototypes of key parts were 3D printed and tested. Performance was checked through observation and any design flaws or areas for improvement were noted. These insights led to new designs, which were tested again. This cycle continued until the parts worked well. Having many 3D printers made this process fast and was a big reason for choosing this method.

For the software, a similar repeated process was used. Initial code was written based on rigid-body dynamics principles and tested on the robot. Performance problems were found and fixed through several rounds of coding and testing, making sure the robot ran smoothly and efficiently.

4 Design and construction

4.1 Ball drive

There are a lot of pros and cons to balls if we compare them with wheels, see table 1.

In general, wheels are preferred: they are easier to constrain into any chassis or frame, have better traction, and have a bigger contact surface. A ball however, does not have the delay of having to turn, before being able to move holonomicaly, whereas standard wheels do. This way the robot has a smaller turn radius, as it can perform a pure rotation.

Besides that, a ball configuration would not require any servo motors, because the rotations generated by the DC motors are enough to acquire the three degrees of freedom needed. A wheel configuration however, would need them to rotate the wheels about the z-axis. Adding servo's makes the path planning of the robot harder, as the servo motors need to be related to the DC motors in the coding. The wheels can also not rotate infinitely, due to the wiring. And lastly, the delay of turning the servo motors before being able to move in a certain way is hard to compensate in the coding. With these reasons in mind, we chose to design a holonomic ball drive robot.

Table 1. Pro's of balls vs. wheels

Wheels	Balls
Better traction	Instant manoeuvrability
Bigger contact surface	One compliant body
Easier to constrain	No steering links needed

4.2 Ball specifications

For the robot to run smoothly, the material of the ball has to have high traction with the surface. Rubber has a very high coefficient of friction, as well as TPU, short for Thermoplastic polyurethane [13]. Rubber balls however, were very hard to obtain or make. TPU on the other hand can easily be 3d-printed and was therefore ultimately chosen.

To design the perfect ball out of TPU, a compromise between material efficiency and stiffness had to be made. If the ball is too flexible, it will deform under load. The pressure from the ball rollers will then cause them to 'sink' into the TPU balls. That would cause a lot of friction, as a result of which the robot will not run smoothly. Too much filament would have made the ball stiff enough, however that would be a waste of material. Balls with a progressively increasing percentage of filament were tested until enough stiffness was obtained. The balls were tested on the robot to analyse how smoothly they ran. Balls were tested with a filament of 15%, 25%, 35% and 40% respectively. The robot did not run smoothly until the 35% balls were tested. A 40% ball did not run smoother, therefore the 35% ball was chosen. With this percentage, both material efficiency and stiffness targets are met, making it suitable for the robot.

The ground in the greenhouse is made of cement, with some little bumps and holes, see figure 4. The robot, after measuring, does not have to overcome any bumps or holes higher or lower than 5mm. The first diameter of ball to generate at 5mm in clearance is 70mm, which leaves the robot with a ground clearance of 8mm. If the ball had a smaller diameter of 60 mm, then the clearance would be -2 mm. Testing showed that the robot could drive on the ground of the greenhouse with no issues with these balls. Bigger balls take up more space and use more material, hence they were not chosen.

Figure 4. Photo of the ground in the TU Delft greenhouse

4.3 Single ball mechanism

The mechanism of a single driven ball requires two degrees of freedom: rotation about the x-axis and rotation about the y-axis. To determine how to constrain the wheel in the most optimal way, minimising play and internal friction, we looked at the scientific paper, on which our final project is based. The 2014 study by G. Runge & G. Borchert & A. Raatz uses the advanced Grübler formula [4]. Through this formula, they arrived at a configuration shown in figure 1.

In this setup, the ball is clamped at 5 points. In the XY plane, the ball is clamped by 4 wheels, each separated by an angle of 90 degrees. In addition, the ball is also constrained by a ball roller from above.

They use two sensor wheels and two drive wheels. The design of our project does not require sensors, as it is operated using a controller. The two sensor wheels have therefore been exchanged for a single ball roller, see figure 5. This ball roller is attached to the frame with the option to put small 1mm rings in between, creating the option to press the ball roller against the ball with a desired pressure. This makes the ball constrained with limited play, and limited friction. In the same figure, it can be seen that the ball roller constrains the ball in y-direction. The wheels constrain the ball in x- and y-direction. These two wheels, together with the ball roller, have been placed 7.5 millimeters underneath the middle of the ball, to prevent it from falling out. A shorter distance from the middle does not constrain the ball enough to not fall out, where a bigger distance lowers the robot beyond the desired clearance. Finally, the ball roller at the top constrains the ball at the opposite end in the z direction. The result of these four constrains is that every translation is restricted, so that the ball will only be able to perform the wanted rotations.

The advantage of our configuration over the one from the previous article, is that by using a ball roller on the side, instead of two additional wheels, less internal friction occurs. This is because a ball roller has three degrees of freedom, where a wheel only has one. When the ball rolls perpendicular to the wheels, friction will be generated. With a ball roller, this is not the case, as it can roll with the ball in any given direction.

4.4 Layout

The full mechanism of the original authors, see figure 6, was analysed, to see whether the single driven ball mechanism they proposed, from figure 1, could be incorporated into our robot. However, after analysing the dynamics such a mechanism would have, two

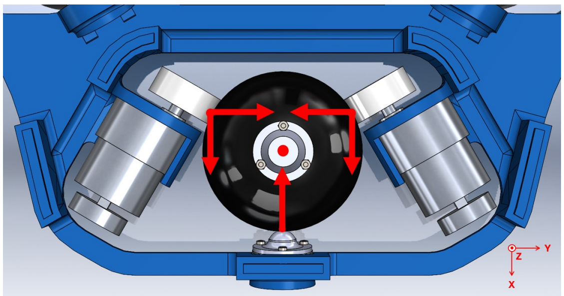

Figure 5. Constraints of a single ball drive mechanism, top view without roof

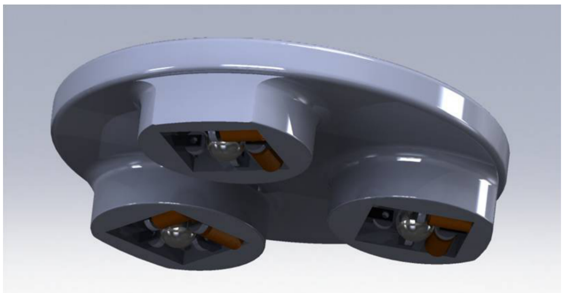

Figure 6. A complete ball drive robot from 2014, figure adapted from [4]

problems were spotted.

First, if only two driven balls were used instead of three, shown in figure 6, the center of rotation would naturally be around rotation point B, see figure 7. The forces of each driving mechanism would then have to be adjusted in order to rotate about rotation point A, which is the centre of gravity. Second, even if the rotation point is accounted for, in a realistic scenario, there is a frictional force present in the free ball mechanism. Meaning that the exact vector of the friction force would have to be known in order to include it in calculations to rotate about point A. Without some kind of sensor system, including the force vector would become very complex. Without it, deviations will occur when the mechanism is driven, and simple acts such as a pure rotation will not be achievable if the free ball undergoes friction.

The layout chosen, see figure 8, consists of four contact points with the ground, two of which are driven balls and two are free balls. The two driven balls are driven via the ball drive mechanisms, as mentioned previously, with their respective degrees of freedom.

In order to rotate and translate the mechanism, a minimum of two driving mechanisms is required. With one driving mechanism, the ball would need to rotate about the z-axis; however, it is currently designed to only rotate about x and y. Furthermore, using two driving balls allows the full mechanism to create a moment around the z-axis while still being free to translate at the same time.

The driven ball mechanisms take up more space than the non driven balls do, due to the addition of two DC motors. The wheels attached to these motors also induce extra internal friction. As per the set design requirements 4, 5 and 6, no more than two driven ball mechanism were therefore implemented into the final robot.

Figure 9 shows how the full mechanism will deform when the motors are burdened with a load of 0.0875 Nm each. This number was calculated by determining the weight supported by each ball. Given the total weight of the mechanism is around 20 Newtons and it rests on four balls, each ball supports

$$\frac{20\text{ N}}{4} = 5\text{ N}. \tag{1}$$

With a coefficient of friction (μ) of 1 (most extreme case), the frictional force per ball is

$$F_f = \mu \times F_N = 1 \times 5\text{ N} = 5\text{ N}. \tag{2}$$

Given the radius (r) of each ball is 0.035 meters, the torque (T) required to start rotating each driving ball is

$$T_{\text{ball}} = F_f \times r = 5\text{ N} \times 0.035\text{ m} = 0.175\text{ N·m}. \tag{3}$$

Since each driving ball is actuated by two motors, the torque required per motor is

$$T_{\text{motor}} = \frac{T_{\text{ball}}}{2} = \frac{0.175\text{ N·m}}{2} = 0.0875\text{ N·m}. \tag{4}$$

At this load, the maximum displacement that occurs is at the end of the motor holders, changing the geometry by only 0.08 mm, which is negligible.

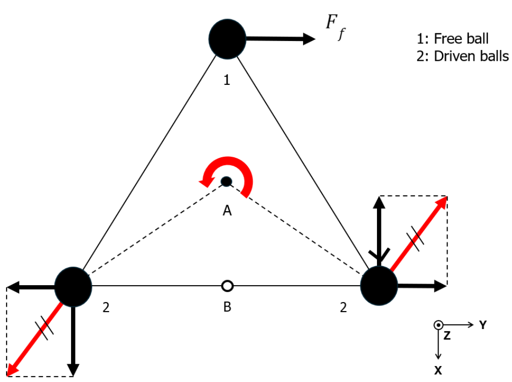

Figure 7. Schematic representation of a rotation for a triangular layout, top view

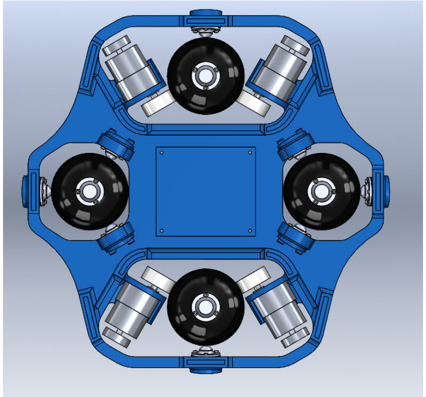

Figure 8. Final layout design, top view without roof

4.5 Main frame

The frame of the robot has been placed as close to the ground as possible. This creates a center of gravity very low to the ground, enhancing the stability of the robot, while also minimising the robot's height. The entire frame is made out of PLA, a stiff and lightweight material [10]. With this material, the frame will barely deform as shown in figure 9. Another reason for using PLA is the ease of manufacturing. PLA is a material that can be 3d-printed. The 'roof' of the frame has been made as a separate part. This is to reduce the amount of supports needed for 3d printing, not wasting any material. Mounting all the parts on the frame is also easier in this way. On the frame, every unnecessary moment arm has been removed and every beam has been shaped as tightly as possible, again to not waste any space and keep the robot as small as possible.

4.6 Wheel specifications

High friction between the wheels attached to the DC motors and the balls is important for transferring the rotational speed of the motors to the balls, without slip. Accordingly, rubber tires are used to drive the two ball mechanisms, as this material has a high coefficient of friction. Simple rubber tires were used in combination with custom 3d-printed PLA rims, to fit onto the DC motors. The outer layer of the rubber tires consist of a profile, that further enhances the traction. To constrain the wheels, they have been made to press fit onto the axes.The wheels have a diameter of 30mm. The dimension was chosen as small as possible, so that the wheel will just reach the ball, without the DC motor also hitting the ball. This can be seen in figure 10.

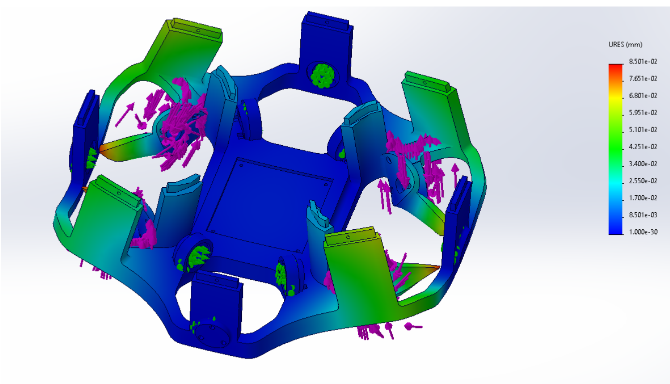

Figure 9. FEM analysis showing the displacement within the mechanism at full power

4.7 Hardware

The specific hardware used for the holonomic drive robot can be found in table 2. The holonomic drive robot makes use of the MIRTE software and hardware. MIRTE is an educational robot, used at the Technical University of Delft. To use this software, an SD card needed to be flashed with the correct file [6]. The second file for using the MIRTE software is a UF2 file that needs to be stored on the Raspberry Pi Pico for controlling the micro controller. At last, a diagram has been made showing the entire circuit. This diagram and other files used within the software can be found on GitHub [5].

4.8 Control

A PS4 controller is used to control the robot, as it allows for easy directional inputs and scalable power magnitudes. To steer the robot, the inputs from the controller must be converted into motor rotation speeds. The left joystick controls the direction and magnitude of the robots linear speed, where the right joystick controls the direction and magnitude of the robots rotation. An equation is needed to transform these inputs (linear speed x, linear speed y, linear speed z) from the body-fixed frame to the motor frame, allowing for the calculation of motor rotation speeds and rotation direction.

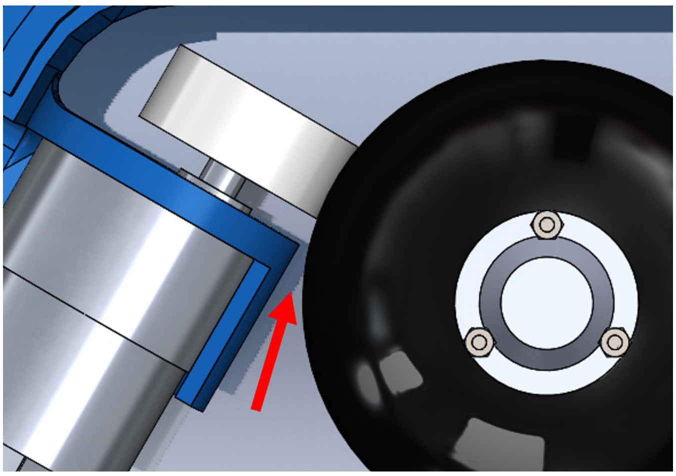

Figure 10. Clearance between DC motor and ball, top view without roof

Table 2. Hardware used

Component	Specifications
DC Motors	12V, 200 RPM, 6mm shaft
Micro controller	Raspberry Pi Pico H
Input device	PS4 controller
Lithium Battery	11-12.6V, 5000mAh, Li-Po battery
Single-board computer	Orange Pi Zero 2
SD Card	14GB, micro SD
PCB board	Mirte-master v0.1
Capacitor	1000 μF

The equation for converting the inputs to motor angular speed is:

$$u = \underbrace{\begin{bmatrix} -\frac{1}{R} & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \cos\beta & \sin\beta \\ -\sin\beta & \cos\beta \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} 1 & 0 & r_y \\ 0 & 1 & r_x \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix}}_{E} \tag{5}$$

$u_{1,2,3,4}$

The equation above provides the angular speed and direction for one motor. So for the holonomic robot, four different calculations need to be done, with the specific orientation per motor. The input vector for this equation, can be seen in equation 5, matrix E, where $\dot{x}$ and $\dot{y}$ are the desired translation speeds and $\dot{\phi}$ is the desired angular speed. Further, matrix D transfers the inputs on a global frame to a body-fixed frame and has the parameter ϕ which is the angle between the body-fixed frame and the global frame.

The translation matrix C that moves origin of the body-fixed frame to the center of the ball, has the parameters r_x and r_y. They represent the distances from the ball centre to the centre of the robot frame.

Matrix B converts the inputs in the body-fixed frame to a frame where the x-axis is always in the direction of the speed vector of the ball. Here β is the angle between the x-axis of the body-fixed frame and the ball speed vector created by the motor.

The matrix A that converts the inputs in the wheel frame to an angular momentum of the motor has the parameter R which is the diameter of the wheel.

In the equation 5, the angle from the fixed-frame to the body fixed frame ϕ is set to 0 for simplicity. This means that the inputs from the PS4 controller are represented in the body-fixed frame. The equation transforms the x, y speeds and z rotation speed, represented in the body-fixed frame into motor speeds.

In the case of an x translation, the ball drive mechanisms both need to rotate in the x direction. This can be realised by rotating the motors in the ball drive mechanism in opposite direction of each other, mirrored for the two ball drive mechanisms. This creates a net force in the x direction without any rotation. See figure 11, model A.

In the case of a y translation, the ball drive mechanisms both need rotate in the y direction. This can be realised by rotating the motors in the ball drive mechanisms in the same direction; either inwards or outwards. To result in a net y moment, the motor rotation per ball mechanism needs to be opposite. So if one side rotates inwards, the other side rotates outward. See figure 11, model B.

In case of the rotation about the z-axis, the driven balls will rotate in opposite direction to each other and perpendicular to the centre of the robot. This can be realised trough rotating the motors in the ball drive mechanism in the opposite direction and mirroring this motion in the other ball drive mechanism. In this way, the rotation of the robot will go exactly around the z-axis, making it a pure rotation. Figure 11, model C, illustrates what the resulting motor spin directions are, given a pure rotation input.

The joystick inputs received from the ps4 controller are never zero, even when the joysticks are not actuated. This can be a problem in two situations: When no translation or rotation is desired, and when only translation in one direction is desired. To prevent unwanted inputs, a code was developed to nullify any data between a certain input range. From input data analysis, the optimal range was determined to be 0.05.

The complete code that has been used can be found on GitHub [5].

4.9 Final design

The final version of the robot, named HoloMirte can be seen in figure 12. The robot has been printed in the colors of the TU Delft and the electrical components have been placed in such a way, that the centre of gravity is still in the middle.

5 Results

Once constructing the robot was complete, the robot was tested to check whether the list of requirements had been met.

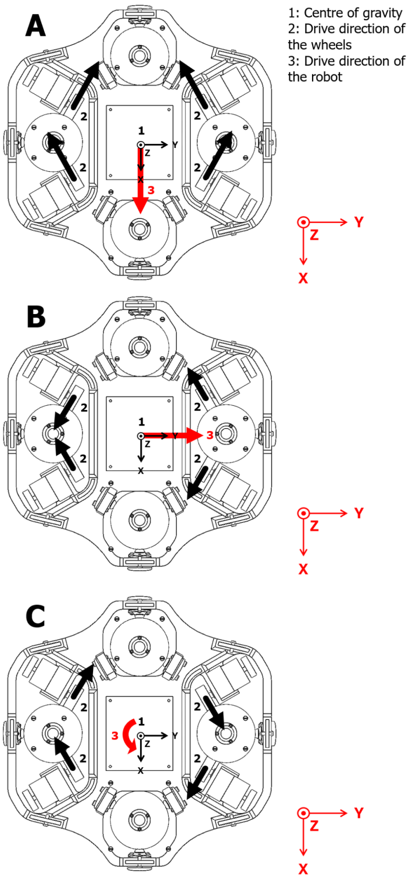

Figure 11. Drive directions of the wheels to realise each degree of freedom, top view without roof

To check whether the robot adhered to design requirement 1, a simple drive test was conducted where the robot would have to translate to location B from position A. From location B, the robot would then have to directly drive in a perpendicular direction relative to the direction it was driving in before, without needing to turn. The robot passed this test.

To pass requirement 2, a simple rotation test was conducted, which the robot passed.

The robot adhered to requirement number 3, and does not consist of any mechanum- or omni-wheels.

According to the FEM analysis carried out in Chapter 4.4, requirement 4 was not completely achieved. The

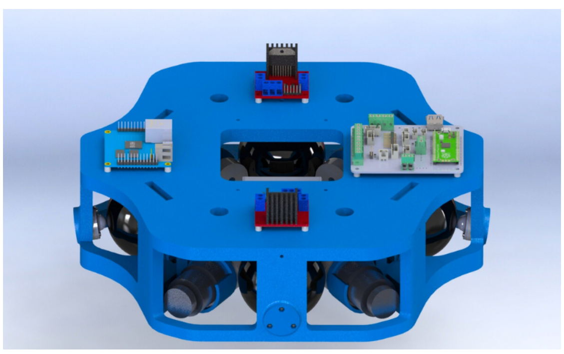

Figure 12. Final design of the robot

FEM analysis yielded results that showed a material displacement of 0.08 mm. Even though not much material is wasted and the structural integrity of the design is very high, the design still has the potential to be optimised to use less material.

The outer dimensions of the full mechanism are 320 x 320 mm. Typical greenhouses, with primary and smaller secondary paths, have secondary paths of a width ranging between 45 and 60 cm [3]. This means that the mechanism is satisfactory, leaving a minimum of 13 centimeters of space within the path. Requirement 5 has therefore been met.

To test whether or not the friction has been minimised, a direction change test was conducted to see whether any frictional forces within the mechanism hinder the mechanism from changing direction. The robot passed this test and changed directions at the same time as the joystick changed direction.

The robot is operated from a distance using a PS4 controller. This means that requirement 7 is met and that the robot is portable. Based on the specs of an Orange Pi Zero 2 and the fact that the standard connection uses a bandwidth of 2.4 GHz, it is known that inside a building, the Orange Pi to the laptop has a range up to 45 meters [1].

For requirement 8, the robot was made to drive in a straight line of 2m long. The final destination of the robot was then analysed to see at how much of an angle the robot deviated from the path. The robot deviated around 5 degrees, meaning that there are still some imperfections. These deviations occurred due to each motor being slightly different than the others. Adding encoders that read the motor speeds and correct them simultaneously through a PD-controller will fix this problem. However, due to the image of the MIRTE software, there is a problem in obtaining the "encoder-speed" values, only the encoder values can be obtained which can only read the absolute value of a PWM signal. This means that only when the motor is spinning in a positive direction, can it be adjusted using a PD-controller.

Videos for all of the experiments conducted can be found on GitHub. [5]

Next to the list of requirements, there were also design questions that needed to be answered.

The ball configuration HoloMirte uses, is great for applications in greenhouses. This is because the mechanism takes factors such as smooth riding, compactness and stability into account, which are desired qualities within a holonomic drive robot. Especially a robot that will have tools placed upon it. This report has indicated that, relative to the model of the original authors, this design uses less material, has constraints that consist of less friction, and uses four DC motors instead of six, meaning that it is more efficient. The robot only sometimes has a little knock, something that could be improved.

Seemingly, the materials used within HoloMirte work and do not break under stress. PLA should be used for the frame as it can easily be 3D printed. It is very stiff and, hence, will not deform. For the wheels, rubber is used as it allows for high efficiency in the transfer of moments from the wheel to the ball. For the balls, TPU is currently used; however, it is possible that better alternatives exist, such as rubber, that may allow more traction with the floor.

When comparing the HoloMirte mechanism to mecanum- or omni wheels and how effective the technologies may be for implementation in greenhouses, there are several factors to consider, such as: recallibration, navigation in tight spaces, cost, surface compatibility and load capacity. When it comes to recalibration, the wheel orientation of mecanum/omni wheels might need to be reestablished to ensure accurate movement, while the HoloMirte can drive away without the need for recalibration. Mecanum/omni wheels can handle rough surfaces better than HoloMirte, as these wheels have a profile due to their non-continuous contact surface. In greenhouses, however, the ground can be adjusted to fit HoloMirte better.

Mecanum/omni wheels also have a higher load capacity, which means that they outperform HoloMirte in carrying heavier tools around the greenhouse. Cost is a factor where HoloMirte excels in. HoloMirte makes use of a lot of 3D printed parts. These 3D-printed balls and the rubber with the rims are cheaper than mecanum/omni wheels (€13 vs. €25) [14]. Both technologies excel at navigating tight spaces; however, mecanum/omni wheels require simultaneous adjustment of the rotational speeds of the wheels, in order to move holonomically. HoloMirte, however, has truly continuous movement in any direction, without the need for coordinated adjustment of multiple actuated components.

HoloMirte was eventually tested in the greenhouse where it will be implemented, and it performed very well, carrying out rotation and translation as expected from the given inputs. However, when HoloMirte drove over the putt, it struggled due to the limited clearance. A solution to this could be to increase the ball size of the robots to 80 mm and to use stronger DC motors if necessary.

6 Discussion

6.1 Error analysis

Sometimes, the Orange Pi kept on disconnecting. The problem here was that the motors demanded too much current, so suddenly, that they would draw it from the current reserved for the Orange Pi. This caused the Orange Pi to momentarily disconnect. The solution is to add a 1000 μF capacitor to the PCB board at the ports of the 12 Volts, before it is regulated to 5 Volts. This causes the capacitor to store some energy, which it is then able to give back to the system, when a dip occurs.

Another effect that occurred, was the tilting of the robot when it drove in the y-direction. The robot tilted about the x-axis, creating an roll movement, resulting in unstable driving. The solution to this problem was to change the shape of the frame of the old mechanism, see figure 13, and to set the distance of the driving balls further away from each other to create a more stable robot.

The first TPU balls were printed as a single part. This caused one side of the ball to have a little flat spot. To fix this, the balls were printed as two halves and put together with a male, female construction.

6.2 Points of improvement

A minor flaw in the design is when the robot changes direction on the x- axis too fast, like in scenario B from figure 11, it has a little knock. Because of the sudden change in direction, one of the single ball drive mechanism shortly becomes over constrained, clamping the ball too tightly. The resulting forces, illustrated in figure 14, cause the wheels to move the ball downwards and the frame upwards, after which the frame falls back again, which causes the knock. A solution to this could be to place the constraints in the XY plane above the middle of the ball. This would

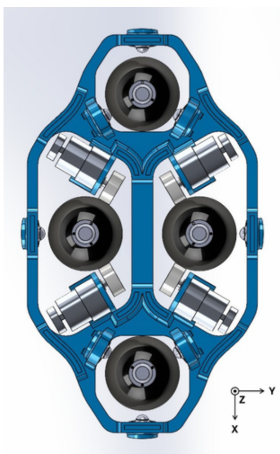

Figure 13. Earlier version of the robot design

significantly decrease the force that tries to push the ball out of the frame. However, a different solution to preventing the ball from falling out of the frame when lifted up should then be considered.

This would require something like a spring that continuously pushes the ball into the other constraints. In this case, a force, from the robot changing direction, will be 'caught' by the spring and the ball will not be constrained looser or tighter.

Secondly, the quality of the ball rollers that are used is bad. They do not run entirely smoothly and cause a bit of friction within the robot. This could be improved by utilising ball rollers of better quality, that naturally run smoother.

The robot receives inputs, which the motors translate into PWM signals that control the motor speed. However, due to deviations within each individual motor, problems occurred that caused a deviation in speed per motor, which then caused directional deviations for the full mechanism. a way to fix this is to activate the encoders within each motor and to integrate them into the code to actuate the motors. This way, the values of the encoders can be used in combination with a PD-controller to ensure that all motors spin at their desired speeds and that the mechanism drives accordingly.

To improve the design of the control system that manages direction (enabling both rotate and translate), the current limitation need to be addressed. the existing system processes inputs from an body-fixed frame perspective which makes it difficult to achieve translation and rotation simultaneous. to overcome this, it is necessary to determine the angle ϕ at each instance in time so it can be used in for equation 6. hereby the control system processes the input in a room-fixed frame which makes translating and rotation simultaneously possible. One method to determine the angle ϕ involves calculating the angle by integrating the robots angular speed, which can be calculated using the data from the motors encoders. However a more accurate approach is to directly measure the angle ϕ. This however would acquire different sensor for the robot.

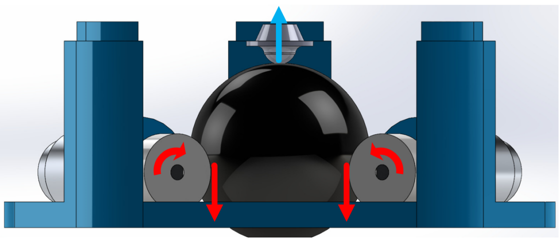

Figure 14. Force analysis on single ball drive mechanism, side view

7 Conclusions and recommendations

Even though mecanum/omni wheels are better at handling rough surfaces and carrying heavier loads, HoloMirte is a leap forwards for greenhouse applications. The design from 2014 has been improved by utilizing fewer motors and materials. The well-constrained ball drives lead to lower frictional losses and therefore improves operational efficiency. Because of the continuous contact surface from the balls, vibrations are minimised and reliability is optimised.

To further improve HoloMirte, the motor control can be made more precise by using advanced encoder integration with a PD-controller. To improve traction, perhaps a different ball material can be researched. If more powerful smaller DC motors exist, they could be implemented into HoloMirte to scale down the full mechanism while maintaining its power. Implementing these improvements will allow HoloMirte to be even more successful in performing tasks in an autonomous greenhouse.

Acknowledgements

During their work on this paper, the authors were financially supported by the Technical University of Delft. They received help and feedback from Martijn Wisse, Arend-Jan van Hilten and their mentors Martin Klomp and Chris Pek. The authors had access to a variety of tools and machines, like 3d printers, provided by the Makerspace in the TU Delft Science Center.

References

[1] B. Mitchell. (2020, 6 november). *What is the Range of a Typical Wi-Fi Network? Lifewire* https://www.lifewire.com/range-of-typical-wifi-network-816564

[2] Brick Technology. (2022, February 2). *Lego Car with Ball Wheels* [Video]. YouTube. https://www.youtube.com/watch?v=5p-B51FKEps

[3] G. Growning. (2021, July 27). Paths, Access and Greenhouse Flooring - Greenhouse Gardening. *Greenhouse Gardening*. https://www.greenhousegrowing.co.uk/around-the-greenhouse/pathsaccessgreenhouseflooring/

[4] G. Runge, G. Borchert, & A. Raatz. (2014). *Design of a Holonomic Ball Drive for Mobile Robots - A potential alternative to a Mecanum wheel.* https://doi.org/10.1109/MESA.2014.6935568

[5] HoloMirte. (2024). *BEP*. GitHub repository. https://github.com/HoloMirte/BEP

[6] Install Mirte Software — Mirte Documentation v0.1 documentation. (n.d.). https://docs.mirte.org/doc/install-mirte-software.html

[7] J. Bruton. (2021, May 18). *Are ball shaped wheels practical?* [Video]. YouTube. https://www.youtube.com/watch?v=rHDCIngOWak

[8] J. Bruton. (2021, November 16). *How many ways can this robot steer?* [Video]. YouTube. https://www.youtube.com/watch?v=My4OxxNe\Rbc

[9] L. Yan, I. Chen, G. Yang, & K. Lee (2006). Analytical and experimental investigation on the magnetic field and torque of a permanent magnet spherical actuator. *IEEE/ASME Transactions on Mechatronics, 11*, 409-419. https://doi.org/10.1109/TMECH.2006.878545

[10] M. Ferdinánd, R. Várdai, J. Móczó, & B. Pukánszky, (2023). Poly(lactic acid) reinforced with synthetic polymer fibers: interactions, structure and properties. *Composites. Part A, Applied Science And Manufacturing, 164*, 107318. https://doi.org/10.1016/j.compositesa.2022.107318

[11] M. Hideaki. (1995). *JPH07257422A*. Espacenet. https://worldwide.espacenet.com/patent/search/family/013907105/publication/JPH07257422A?q=JPH07257422A

[12] M. West, & H. Asada. (1995). Design and control of ball wheel omnidirectional vehicles. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*. https://doi.org/10.1109/ROBOT.1995.525547

[13] Thermoplastic Polyurethane (TPU) material: Properties & structure. (n.d.). https://omnexus.specialchem.com/selection-guide/thermoplastic-polyurethanes-tpu

[14] V. Putkaradze, & S. Rogers. (2018). On the dynamics of a rolling ball actuated by internal point masses. *Meccanica, 53*(15), 3839–3868. https://%doi.org/10.1007/s11012-018-0904-5

[15] 123materialen. (n.d.). *4 stuks 60 mm macanum omni wielen voor DHZ rc robot auto.* Retrieved June 14, 2024, from https://shorturl.at/SLaRK

8 Appendices

8.1 Calculations for torque on motor holders

The input vector for this equation is:

$$\mathbf{v} = \begin{bmatrix} \dot{\phi} \\ \dot{x} \\ \dot{y} \end{bmatrix} \tag{5E}$$

Rotation matrix for room frame to robot body-fixed frame:

$${}^{B}C_S = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5D}$$

Rearrange the rotation matrix so it matches our input vector:

$${}^{B}C_S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{6}$$

The second matrix is a translation matrix so the origin of the body-fixed frame is in the center of the ball:

$$^{B}C_S = \begin{bmatrix} y & 1 & 0 \\ x & 0 & 1 \end{bmatrix} \tag{5C}$$

Rotation matrix for robot body-fixed frame to a wheel frame:

$$^{B}C_S = \begin{bmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

The rotation matrix that converts the inputs in the wheel frame to an angular momentum of the motor:

$$^{B}C_S = \begin{bmatrix} -\frac{1}{r} & 0 \end{bmatrix} \tag{5A}$$

The equation for converting the inputs to motor angular speed looks like:

$$u = \begin{bmatrix} -\frac{1}{r} & 0 \end{bmatrix} \begin{bmatrix} \cos\beta & \sin\beta \\ -\sin\beta & \cos\beta \end{bmatrix} \begin{bmatrix} y & 1 & 0 \\ x & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{x} \\ \dot{y} \end{bmatrix} \tag{5}$$

working out the equation for each motor. r is for every motor the same: 15 mm.
for motor 1: $\phi = 0$, $\beta = 30°$, $x = 0$, $y = 105$ mm

$$u_1 = -6.0621\dot{\phi} - 57.7350\dot{x} + 33.3333\dot{y} \tag{8}$$

for motor 2: $\phi = 0$, $\beta = -150°$, $x = 0$mm, $y = 105$mm:

$$u_2 = 6.0621\dot{\phi} + 57.7350\dot{x} + 33.3333\dot{y} \tag{9}$$

for motor 3: $\phi = 0$, $\beta = 150$, $x = 0$mm, $y = -105$mm.

$$u_3 = -6.0621\dot{\phi} + 57.7350\dot{x} - 33.3333\dot{y} \tag{10}$$

for motor 4: $\phi = 0$, $\beta = 30$, $x = 0$mm, $y = -105$mm.

$$u_4 = 6.0621\dot{\phi} + 57.7350\dot{x} - 33.3333\dot{y} \tag{11}$$