Space technology

Sebastian Staffner
Mechatronics
HTL Saalfelden
Saalfelden/Austria
sebastian.staffner@htl-saalfelden.at

Finn Prostak
Mechatronics
HTL Saalfelden
Saalfelden/Austria
finn.prostak@htl-saalfelden.at

Oskar Mayr
Mechatronics
HTL Saalfelden
Saalfelden/Austria
oskar.mayr@htl-saalfeden.at

Elisa Gassner
Mechatronics
HTL Saalfelden
Saalfelden/Austria
elisa.gassner@htl-saalfelden.at

Sophie Resl
Mechatronics
HTL Saalfelden
Saalfelden/Austria
sophie.resl@htl-saalfelden.at

Simon Freiberger
Mechatronics
HTL Saalfelden
Saalfelden/Austria
simon.freiberger@htl-saalfeden.at

***Abstract*— This paper will clarify the knowledge the group acquired during the previous constructing and learning phase. To be more precise, it will compare problems of space technology and our own problems that occurred in some of our first robotic constructions because we found that there are many similarities in the way of constructing technology in space and in the European educational robotics competitions. Furthermore, the comparison to space technology helped us find new ways on looking onto our problems. Many similar problems had similar solutions and so we could us the solutions in our own constructions. In addition, we hope to get enough attention with this paper or at the competition itself or meet experts at the competition to even expand this applied knowledge. We hope that our questions will find answers in the future throw the competition and that this paper will find its way to the right person so that mistakes of it may be corrected. Our intension for Botball was to extend our knowledge. This are some points where we succeeded, and some were we hope to succeed in the future.**

Keywords – Hands-off, Space, Automation Process,

I. Introduction

In Botball it's all about a hands-off Mentality. To start the robots and then let them do their own thing without any outer force interfering. This is certainly a big challenge for the teams that are taking part in Botball. But sadly this Hands off Mentality isn't always a decision. Just think about how long it took for autonomous cars to get to the point where they are now. You just can´t control everything that happens in the real world, or even outer space. In outer space the issue gets more complex, the more you leave our earth. The delay of a signal to Mars is about 10 minutes. This hinders the engineers designing the robots massively. From the second the robots land on Mars, they have to do everything on their own. The challenges companies like NASA or ESA face when sending a robot into space are in some kind of way very similar to the challenges we face in Botball, so today we would like to show you how we used the discoveries from building robots for outer space, for building robots on the Botball table.

.

II. Concept

A. Navigation

One of the most important components of technology in space is its navigation. Without your valid position everything else is meaningless. We experienced the same kind of thinking in Botball constructions. In both fields, performing an action without the right position could cause that it will be effectless. Furthermore, there will always be the risk of a crash.

In space you want to know your position at any time. On the one hand it is important to compare the data sent from the spacecraft with the associated position. If not, it could not be interpreted. The whole mission would be a meaningless waste of money, if there was no new knowledge that can be accomplished or specific data that could be valued. On the other hand, there is the same need of navigation like in Botball, when their robots accomplish missions on the planar surface.

In the Botball tournament it is important to navigate the robot to the place of its task and to have inner navigation, so the task can be completed. With inner navigation we mean that any part of the robot must know which are required to get that cube of that tower or similar tasks.

Navigation in space engineering is parted into three segments:

- First there will be the need of a reference trajectory. This must be designed. Therefore there must be a working cooperation between the software engineers and the frontliners who will use it afterwards. A lot of mathematics, physics and experience is needed.
- Secondly the tracking of the position during the launch phase. Although the spacecraft follows the trajectory, there are several disturbances that cause that the trajectory will be left. Therefore the position must be track at any time.
- Finally, the flight path control. During the flight the flight path control evaluates the mistakes of the position to the trajectory. Only than the path can be corrected. Evaluating and correcting is a constant cycle during the whole flight.

To evaluate the position measurements are necessary. There will measurements of distance to earth, velocity components, optical navigation and the position in earth's sky. This last point will be accomplished by angular measurements of the spacecraft to stations on earth. [1]

As for Botball a kind of trajectory is necessary. Although we have the plan to keep everything on the ground, a path that the robot should follow is required. We have to plan in which order the tasks should be completed and plan at what time which command has to start. Consequently, like in space, we need to correct mistakes caused by disturbances. Those can be a unsymmetric drive of the robot or something it gets stuck into. The whole way from starting box to the finish, there has to be a path control. As for Botball this path control will be using lines, the border to center the robots and time measurement. Also, a camera may be used. This knowledge of space engineers helped us a lot. The thoughts needed to obtain a good path, the importance of lines and a self-correcting vehicle may be the most important wisdom we achieved during the entire building and constructing phase.

B. *Usage of sensors*

The Problem with space is that the robots must orient themselves and choose independently because in space there is no fixed point at which a robot could orient from.
In Botball, the same thing can be said because the robot must orient itself with just the railings and its environment, given that in Botball there are some things to orient from and in space, there is usually nothing.
To overcome this problem in space we can use specific equipment namely cameras, radar, lidar, and other Sensors to see and/or predict orientation and the way to go.
Solving the problem with hands-off engineering in Botball, we had to find ways to use the given sensors and equipment to drive and behave semi-autonomous. We implemented a way to use light sensors to detect the line on the floor to orient the robot in a more suitable position. Another way to overcome not being parallel to the boundaries of the game table is to drive the robot in reverse to the railing to sort of square it in a parallel position.
Another way to solve the orientation problem in space is to let Nanosatellites communicate with any autonomous systems in space, they were used by NASA. Nanosatellites observe the planet's surface and commune any suitable landing spots to Lander-Rockets.
Because the motors tend to deviate, we are using a self-written program to compensate for this offset to follow the line on the floor. Given the robot must follow a line in the first place. The distance sensor, Limit-switches or the bumper for that matter are used to indicate the offset from the boundary and the robot.
The translation from Imperial to Metric was another problem found in building the game table. Due to the inaccuracies of converting the Imperial to the metric system used by Europeans, we have slight deviations in the confinement. In extension to this, every game table has its own deviation, which causes some problems which we have solved by implementing the techniques listed above.
In summary making something autonomous is hard and it requires a lot of planning, structuring and most importantly testing. [2]

C. *Folding Satellites*

You might be asking yourself, " Why would someone fold satellites?". There is a pretty easy answer to this question. If you want to shoot a probe into space you want to get the most usage of the space you have. Getting the most practicality out of a small space is also a challenge we face with Botball. You want to build a big arm, but are then faced with the issue that it does not fit into the starting box or is to tall. [3]

D. *Testing*

Like it was said in the beginning, the rovers have to basically operate on their own up in space. The robots have to work independently from the moment they reach the planet. To prepare the rovers for the obstacles and challenges they will face on the other planets, as many test as possible are being done on earth.
It starts at the simplest things, like driving to a destination. The rover has to find a way to get there as quickly as possible and also dodge obstacles which could lead to damage or even the failure of the rover. For this NASA built their own, Mars like, landscape own earth. You could compare this to the table we built for the Botball competition. The smallest inconveniences, which probably would not bother us humans can be a big challenge for the rovers. A human can see a big rock, know it is too big and just walk around it. This is mostly through experience of the past or just logic. But how do you train a robot to know such things, as it was never on Mars. But a robot is not just on Mars for driving around. It has to complete tasks, like drilling for rock samples and analyze them. The drilling of the samples is also an issue, for example if the drill gets stuck because the robot slides around a gradient, which means game over. The drill could snap off, which means that the robot is basically useless. You can test for these challenges to a certain extent, but never be fully prepared. This is also, in a bit less risky, what is happening in Botball. You can prepare and test forever, but you are never sure if it is going to work in the end. [4] [5]

III. IMPLEMETATION

All those problems play a very big rule in the designing and building progress of a robot. Every single aspect needs to be deliberated and every possible error avoided. It all starts with the orientation of a robot on the game table. Every robot needs to know its exact position for grabbing objects, colleting poms and doing other tasks. Orientation is one of the most complex challenges for the programmers and the robots itself. If a robot is not oriented very well, it is doing its tasks not very reliable which leads to complications during the match. We solved the problem by using small light reflection sensors for the recognition of the black and grey lines on the playboard. These sensors helped us to drive straight lines by using a simple line follow algorithm. Stopping at a specific line enabled us to help the robots finding their exact positions on the game table, exactly knowing where they are at the time.
If the robots know their exact position there is only one problem in orientation left. The robots need to be calibrated in their own rotation axis. Solving this problem is not too hard. Around the game table there are all these boarders. These ones are perfectly rectangular or parallel to the black lines, the towers and cubes lying on the game table. All these boarders may be used to get the robots in the right rotation position by using the limit switches or bumpers like the one on the create robot. Another way to use the banners is to drive against them to perfectly align the robots.
The compact construction of our robots led to another problem. Grippers need to grab things like cubes in a very high height. Therefore you need big robots, but the robots are limited in their height, their length and width. A compromise in between force of the tools, height and a compact construction needs to be found. Driving arms have to expand

after the start signal, but they also have to be as lightweight as possible to ensure simple movements and they have to be very stable. The motors do not have that much force, which causes problems with long arms, heavy grippers or heavy objects, that need to be moved to some specific destinations.

The main thing of building and programming the robots is testing. Therefore, our team built the official 2023 game table for practice and testing. Because the competition follows the hands-off rule, it is even more important to test the hole process very often. The game table is the most important tool for testing. It is like the artificial Mars surface the NASA uses to test their robots and rovers. Without this facility a mission would not proceed successfully. And after a mission started, nothing can be fixed anymore. The rovers are thousands or millions of kilometers (1000m) away from the earth, far away from any civilization. A test is not done by one try, even if the one try works successful, many other test have to be done, to avoid any problem that may occur.

IV. Conclusion

You would think that space engineering is a very confusing subject, which it absolutely is. But the fundamentals were really helpful when solving the obstacles we faced when building our own robots. Sadly, our robots are not going to space anytime soon. But maybe the knowledge we gathered along the way can help some of us in our further life as engineers.

References

[1] https://solarsystem.nasa.gov/basics/chapter13-1/

[2] https://www.basicthinking.de/blog/2020/08/11/perseverance-mars-rover-2020/

[3] https://engineering.tamu.edu/news/2021/02/researchers-create-origami-inspired-satellite-antennas-that-can-self-fold.html

[4] https://www.esa.int/Enabling_Support/Space_Engineering_Technology/Rovers_drive_through_Tenerife_darkness

[5] https://mars.nasa.gov/mer/spotlight/marsrover01.html#:~:text=Rover%201%20underwent%20acoustic%20testing,and%20passed%20with%20flying%20colors.&text=JPL%27s%20Space%20Simulator%20Facility%20serves,undergo%20solar%20thermal%20vacuum%20testing