

8 dof quadruped robot.

8 dof quadruped robot

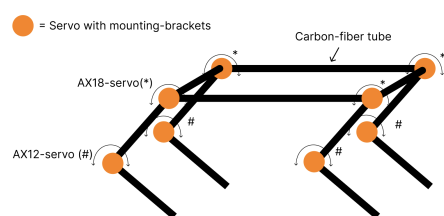
A project in IN5590 by Samuel Valland Lyngset and Hallvard Innset Hurum, E-mail: samuelvl@uio.no and hallvaih@uio.no

1 Intro

We chose to make a quadruped robot because it delivers the right mix of ambition and practicality. By staying with four legs, we keep the hardware scope manageable while still tackling interesting locomotion and optimization challenges. This path let us operate without relying on the ROBIN walking hub. the goal if this project is as follows:

- build a 3D-printed quadruped robot that can move as fast as possible without falling.

We started by scetching out an idea of how this could be done. There are a lot of 12 DOF "robot-dogs" on the internet, but we implemented it with only 8 DOF. This makes the robot a lot simpler, but it means the turning-performance is poor. Hence why we choose the goal to be to walk as fast as possible.

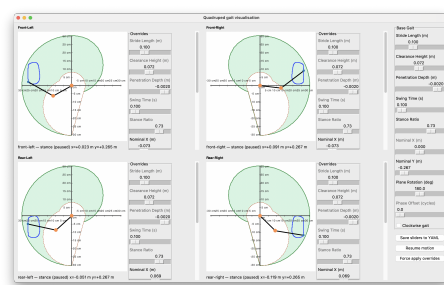


2 Tools

We wanted a modular design-philocopy. Most of the parts is used four to eighth times making it easy to replace and refine parts. This modular approach gives us a lot of flexibility, but it makes the robot heavier than what is absolutly necessary. We have also encountered some problems with slipping in the joints between the tubes and the 3d-printed clamps. The BOM for the most important components are as follows:

- 8 Dynamixel AX18
- 12mm Carbon Tubes
- 3D-printed PLA components
- Moulded Silicone "socks" for the feet

Our robot control was optimized for speed by adjusting the gait parameters. We created a graphical user interface to adjust the robots gait parameters and found what we think ar good settings.



We used inverse kinematics to calculate the correct angles for the dynamixels located in the hip and knee of the four legs. In addition, we based our gait bezier curve on a paper from Rahme et. al (2010)[1].

3 Conclusion

We are quite pleased with how fast the robot is walking now. In reference to the current physical design we do not think it can become a lot quicker due to the strength of the 3d printed parts, overheating of the servos and traction of the feet.

We optimized the robot by marking a set distance and mesuring the time it took for the robot to cross the distance. We started with a grid-search changing only two parameters, but quickly realized that quite a lot of parameters needed to be tuned to find a good combination. As such we resorted to a more manual approach testing a lot of different combinations.

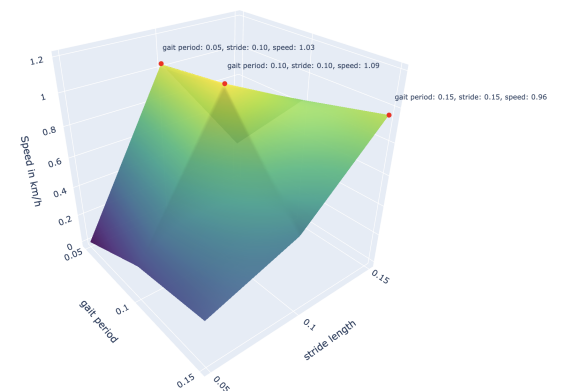
For the future work of the robot, there are several things that can be done. Here is a list of the most pressing issues:

- Optimize the gait further in simulation, maybe using RL
- Controll the gait based on feedback from a gyro for stability
- Adding a battery and an arduino or Raspberry Pi for controll

Extra figures



picture from training



plot of initial grid-optimization approach



screenshot of assembly in solidworks

References

- [1] Maurice Rahme et al. *Dynamics and Domain Randomized Gait Modulation with Bezier Curves for Sim-to-Real Legged Locomotion*. 2020. arXiv: 2010.12070 [cs.RD]. URL: <https://arxiv.org/abs/2010.12070>.



UNIVERSITY
OF OSLO

GitHub repo →

