

POLITECNICO DI TORINO

Corso di Laurea Magistrale in Mechatronic Engineering



Mechanical modularity for e-puck3 education robot

Student:
Marco Trovato

Professors:
A. Rizzo
F. Mondada

October, 2019
Torino, Italy

Acknowledgements

I would like to firstly thank my project assistants Norbert Crot and Daniel Burnier for the continuous availability and assistance they gave me during these months. Furthermore, important advices have been received after consulting professor P. Müllhaupt and Y.Piguet, thus my gratitude goes to them as well.

Finally, a sincere thanks to professor A. Rizzo who helped me remotely with formal aspects concerning the thesis drafting.

Abstract

This master project had as main goal the research of mechanical solutions to provide modularity of mechanical components mounted on e-puck3 mobile robot, an educational tool used for future generations of students. To this aim, several gearbox designs were made by providing flexibility in terms of: reduction ratio, exploited materials and occupied volume. After that, an deep analysis of one of the possible robot configurations (2-wheeled inverted pendulum) led to characterize the robot dynamics by means of non-linear equations found both through Lagrangian and Newton approaches. The dynamic was simulated on Simulink implementing different control techniques which gave good performances letting the user to easily play with some control parameters to obtain the desired robot behavior. In addition, to prevent slipping conditions during motion, an accurate analysis was made by looking at the physical correlations established among mechanical aspects such as: distance of pendulum body center of mass (CoM), friction coefficient between wheel and ground, weight ratio between wheel and pendulum body. Finally a gearbox was designed to give e-puck3 performances similar to EduMip robot^[1], simulating its behaviour on Simulink and validating the motor choice (Maxon Brushless DC motor) made before the start of this project.

Table of contents

List of figures	vii
List of tables	x
Nomenclature	xi
1 Project introduction	1
1.1 Thesis contribution	4
1.2 Educational mobile robots: State of the art	5
2 Mechanical design	7
2.1 Design of magnetic encoder support	7
2.2 First gearbox design: misaligned spur gears	10
2.3 Second gearbox design: aligned spur gears	12
2.3.1 Gears with 0.8 module	12
2.3.2 Gears with 0.5 module	13
2.3.3 Plastic gears	14
2.3.4 Housing design for motor module	15
2.4 Third gearbox design: worm drive	16
2.5 Omniwheel design	18
3 Two-wheeled inverted pendulum configuration	22
3.1 Modeling the 2WIP	23
3.1.1 Lagrangian approach	24
3.1.2 Newton-Euler approach	27
4 Friction analysis	31
4.1 System parameters correlation	31
4.2 Gearbox variation	33

Table of contents	vi
4.3 Slipping evaluation	36
5 Controller design and simulation	38
5.1 LQR design	38
5.2 Model linearization	39
5.2.1 State-space representation	39
5.3 First simulation: PID and LQR control	41
5.4 Second simulation: LQR control	46
5.5 Third simulation: including BLDC motor	48
5.5.1 Introduction to simulation	48
5.5.2 Simulation including the motor	52
5.5.3 Motor operating conditions	53
5.5.4 Motor efficiency	54
6 Conclusions and future improvements	55
Appendix A Simulink Model blocks	56
Appendix B Motor data-sheet	63

List of figures

1.1	e-puck1. ^[2]	1
1.2	Main electronic board for e-puck3 ^[2]	2
1.3	Inverted pendulum ^[2]	2
1.4	Caterpillar ^[2]	2
1.5	Omnidirectional robot ^[2]	3
1.6	Delta robot ^[2]	3
1.7	Drone ^[2]	4
2.1	Operating principle of a magnetic encoder ^[3]	7
2.2	Disc magnet glued on the rear part of the rotor.	8
2.3	Support plates on stator and sensor.	9
2.4	Final housing of the motor.	9
2.5	First gearbox conception.	10
2.6	Axial width of the motor block.	11
2.7	Axes misalignment and motor height.	11
2.8	Second gearbox design.	12
2.9	Geometric constraints for gears below the motor.	14
2.10	Complete design exploiting a worm gear.	15
2.11	Complete motor block attached to the wheel.	16
2.12	Single enveloping gear drive.	17
2.13	Complete motor block of third design.	18
2.14	Main types of omniwheel ^[14]	19
2.15	Rezero project by ETH ^[15]	19
2.16	CAW geometric features ^[16]	20
2.17	Radii as function of θ_i , for 6 pairs of rollers.	20
2.18	Radii as function of number of rollers ^[16] .	21
2.19	Manufacturing problem for inner roller.	21

3.1	e-puck3 horizontal configuration.	22
3.2	Schematic of the 2D inverted pendulum model.	23
3.3	Schematic of the system.	27
3.4	Tangential forces on teeth contact point.	28
3.5	Pendulum free body diagram.	29
3.6	Wheel free body diagram.	30
4.1	Static forces on e-puck3.	31
4.2	Difference between e-puck2 and e-puck3 motor.	34
4.3	Minimum mass ratio for different μ_s	36
4.4	Minimum mass ratio variation for different d	36
5.1	Simulink schematic with PID and LQR control.	42
5.2	Simulation with PID and LQR control.	43
5.3	Simulation of the first part of motion.	44
5.4	Zoom on simulation with bad LQR tuning.	44
5.5	Switching control with bad LQR tuning.	45
5.6	Simulink schematic with only LQR control.	46
5.7	Simulation with only LQR control.	46
5.8	Simulation of the first part of motion.	47
5.9	Kinematic relations among angles.	49
5.10	Simulink schematic including BLDC voltage control.	51
5.11	Simulation result including BLDC motor.	52
5.12	Operating points of BLDC motor.	53
5.13	BLDC motor efficiency.	54
A.1	Detailed simulink schematic.	57
A.2	BLDC block settings.	58
A.3	BLDC motor block connections.	59
A.4	BLDC motor - Gates.	59
A.5	BLDC motor - PWM on Voltage.	60
A.6	BLDC motor - Decoder.	60
A.7	Electromechanical conversion.	61
A.8	Torque losses.	61
A.9	System states evaluation.	61
A.10	Kinematics relations.	62
A.11	Voltage control.	62

B.1 EC20 Data-sheet	64
-------------------------------	----

Chapter 1

Project introduction

E-puck3 is small and cheap mobile robot, with many functionalities and performances appropriate for an educational tool. The main idea behind this project is to provide e-puck3 both of software and hardware flexibility. In Fig.1.1 you can see the first version of e-puck, which was followed by e-puck2 which took the same mechanics of the first but with a new electronic board.

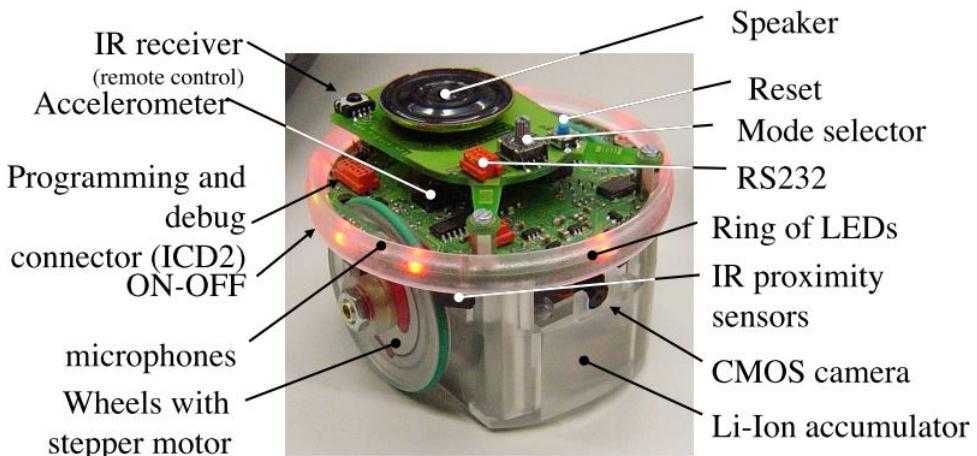


Fig. 1.1 e-puck1.^[2]

Now, with e-puck3, the goal is to have a main electronic board (shown in Fig.1.2) and then printable parts allowing the robot to easily switch among different configurations such as: an inverted pendulum (Fig.1.3), a caterpillar (Fig.1.4), an omnidirectional robot (Fig.1.5), but also a delta robot (Fig.1.6) or a drone (Fig.1.7); simply attaching the desired mechanical modular structure.

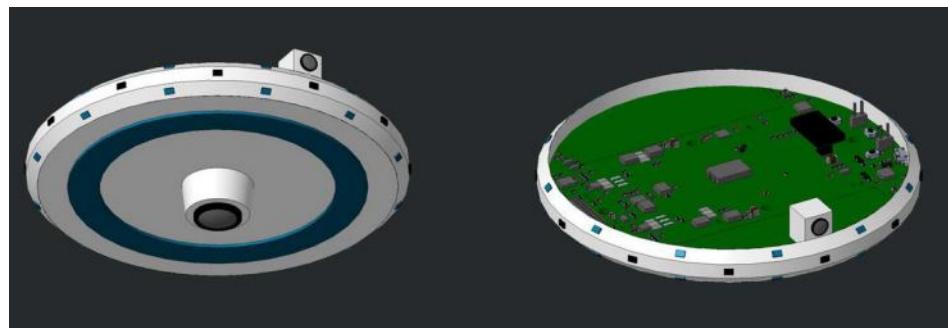


Fig. 1.2 Main electronic board for e-puck3^[2].

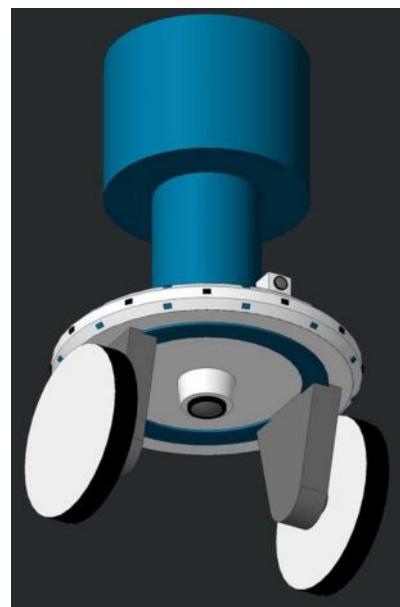


Fig. 1.3 Inverted pendulum^[2].



Fig. 1.4 Caterpillar^[2].



Fig. 1.5 Omnidirectional robot^[2].

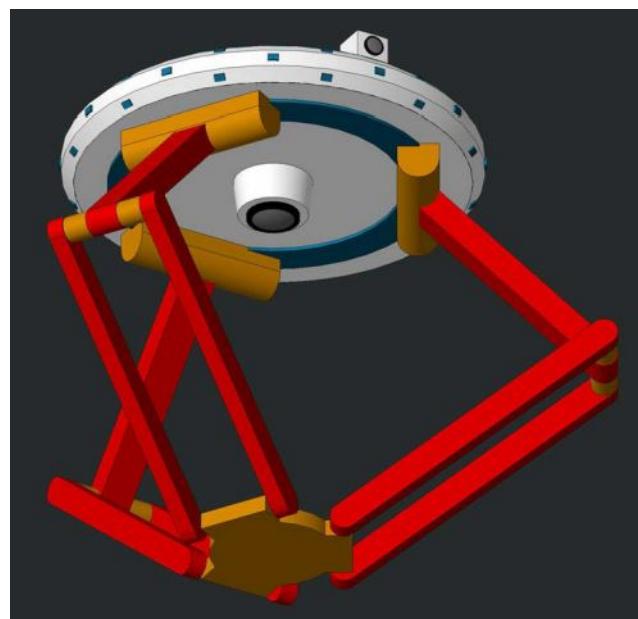


Fig. 1.6 Delta robot^[2].



Fig. 1.7 Drone^[2].

1.1 Thesis contribution

In the first part of this project, mainly mechanical aspects were treated (see chapter 2). Inverted pendulum and omnidirectional robot configurations exploit the same motor and transmission, thus at the very beginning, a shared design of the supporting frame for the sensor exploited on the motor was made. Then, different gearboxes were designed to perform a speed reduction from motors to wheels. Still in chapter 2, an important geometric analysis of the continuous alternate wheel exploited in omnidirectional robot was carried out.

The second part of the project addressed modeling and control techniques to perform a self-righting motion of the inverted pendulum configuration. Therefore, in chapter 3, the system has been modeled following two different approaches (Lagrange and Newton). In chapter 4 a significant parametrization of the main variables, describing the mechanical system, was made to prevent slipping conditions during motion. Finally, in chapter 5, the self-righting maneuver was simulated on Simulink with different control strategies having as further goal the validation of the motor choice.

Chapter 6 shows the conclusions of this work and which future improvements can implemented to overcome most of the problems faced during this project, the first on e-puck3.

1.2 Educational mobile robots: State of the art

So far, most of the educational applications of robotic technology have mainly focused on supporting the teaching of subjects closely related to the robotics field, such as robot programming, robot construction, or mechatronics. Moreover, especially in high school, most of the applications use the robot as a passive tool of the learning activity, meaning that building/programming the robot following a standard procedure becomes more important than exploring its functionalities.

On the contrary, e-puck3 modularity allows the user to play with the different features (both mechanical and electronic) depending on the selected robot configuration. Furthermore, although e-puck3 has not yet a first prototype and part of its requirements are still unknown, the final target is to have a mobile robot of mass around 0.5kg and maximum speed around 1m/s . A camera, LEDs, speakers, IR proximity sensors, IMU and DC brushless motors should be the main components students will play with.

Around the world lots of educational mobile robots are used, among them one can find:

- *Boe-Bot*: a detailed manual helps a beginner user to build step by step this differential drive robot. An USB port allows to download the programs created on PC (PBASIC language) to the robot. In terms of hardware, the main frame is made of aluminium parts, while all the three wheels (2 motorized + 1 for support) are made of plastic.
- *Sphero SPRK+*: it is a spherical robot which trains secondary school students to programming through the visual Sphero Edu App. Its inductive charging base supplies the energy necessary for navigate mazes, compete in a race, swim, dance, and communicate with his companions thanks to a Bluetooth connection within a distance of 30m . With regard to technical specifications, a polycarbonate shell protects wheels, motors, electronic board and battery. With a diameter less than 10cm it can reach up to 2m/s .
- *mBot v1.1*: this is a differential dirve robot thought for children and it's made up by several parts which have to be assembled by the user as part of the learning process. An ultrasonic range finder and a light sensor, in association with a IR remote control, are used for line-following races, football matches, obstacle courses. To program the mBot, one can either use the visual software designed by Makeblock or use the Arduino IDE. Different sensors and actuators can be used to change the robot's behaviour based on its surrounding environment.

- *Mobot-Explorer-A1*: a flat rectangular platform made of steel contains all robot components except wheels, fixed outside the structure. The latter are driven by four fixed-speed DC motors with high torque provided to each corresponding planetary gearbox. This heavy robot can't reach high speed (max $0.38m/s$), but its rubber tires adapted for difficult terrains and admissible payload of few kilos make it a strong and reliable mobile robot. It can be equipped with an appropriate choice of sensors and wireless module, enabling its control system based on the ATmega128 microcontroller to work autonomously.
- *TurtleBot3*: in both of his versions (Burger or Waffle) this robot offers lots of software/hardware evolution opportunities, making it accessible to all users and budgets. Projects including ROS, Machine Learning and SLAM are great chances to test its functionalities. It can be fully programmed using an Open Source control board with Arduino and built with 3D printed parts to reshape its external frame. This differential drive robot is driven by two servo-motors attached to plastic wheels, for a maximum speed of $0.22m/s$ and its small dimensions (Burger version $138 \times 178 \times 192mm$, $1kg$) allow a maximum payload of $15kg$.
- *Pioneer P3-DX*: it is a compact differential-drive mobile robot which has become the reference platform for robotics research. This versatile platform its ideal for autonomous navigation, manipulation and mapping or even remote-controlled applications. Software applications increase control and precision of movements, but also colour and speech recognition. Hardware options increase mobility and interactivity through robotised grippers and arms, video camera, ultrasonic motion sensor, joystick and gyroscope. Differently from previous mobile robots (masses up to $1kg$), its total mass is $9kg$, due in part to 3 batteries for an autonomy of around 9 hours.
- *Stingray*: this is a mid-size robot ($330 \times 277 \times 140mm$) used for a wide range of projects, especially where multiple tasks are requested at the same time. This capability is due to its multiprocessor control system, the Propeller Robot Control Board, which makes Stingray a good choice for advanced robotics. Being a differential drive robot, it has two DC spur gear motors attached to plastic wheels and two additional small omnidirectional wheels used for support. Multiple mounting locations for sensors, add-ons, etc. are spread around the main frame, making this robot more customizable.

Chapter 2

Mechanical design

2.1 Design of magnetic encoder support

The first task of my project was to design a support for the magnetic encoder exploited to control the motor. The operating principle of the sensor (Fig. 2.1) is very simple, indeed four small sensors, fixed inside the chip, continuously detect the magnetic field produced by the magnet. For this reason the sensor was fixed on an extension of the stator housing with the aim of decoupling the rotating movement of the magnet from the static Hall sensor.

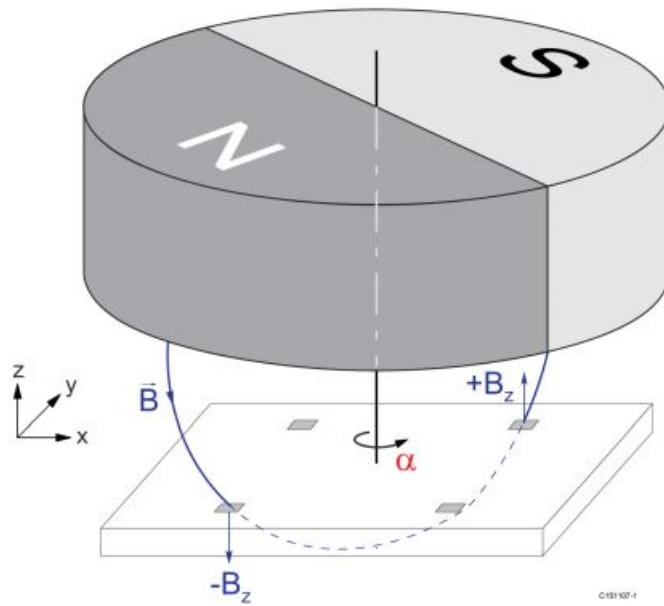


Fig. 2.1 Operating principle of a magnetic encoder^[3].

Starting by analysing the structure of the motor and consulting its data-sheet (provided in Appendix-B), I searched the Internet to find a magnet feasible for this application, so I found^[4] a disc magnet with a thickness of 3mm and a diametrical magnetization (required for the sensor principle).

At the moment we think just to glue the magnet on the rear part of the rotor (Fig.2.3) rather than build an apposite small support structure.

An evolution of the motor block is shown in Fig.2.2, Fig.2.3 and Fig.2.4

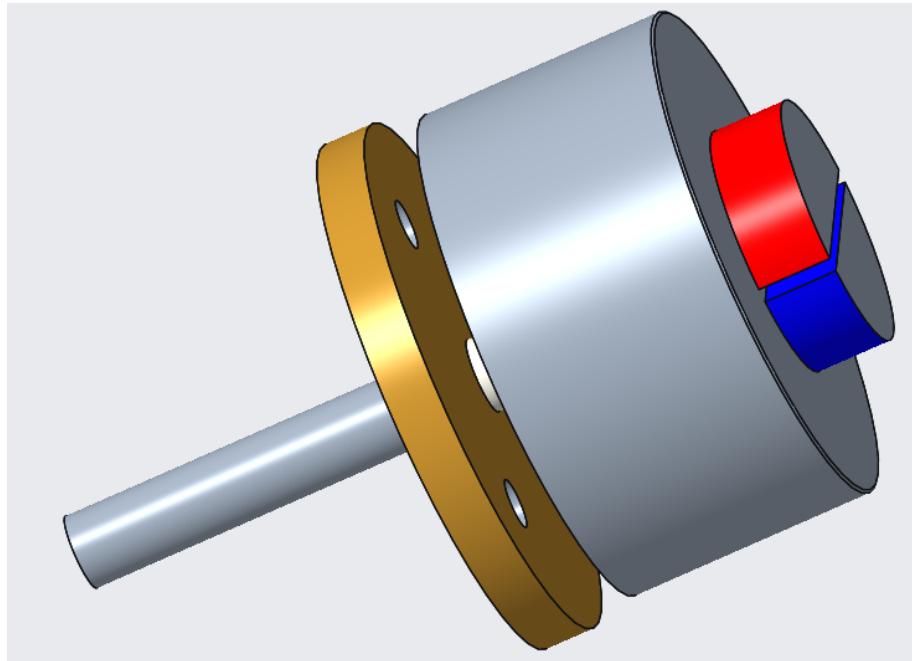


Fig. 2.2 Disc magnet glued on the rear part of the rotor.

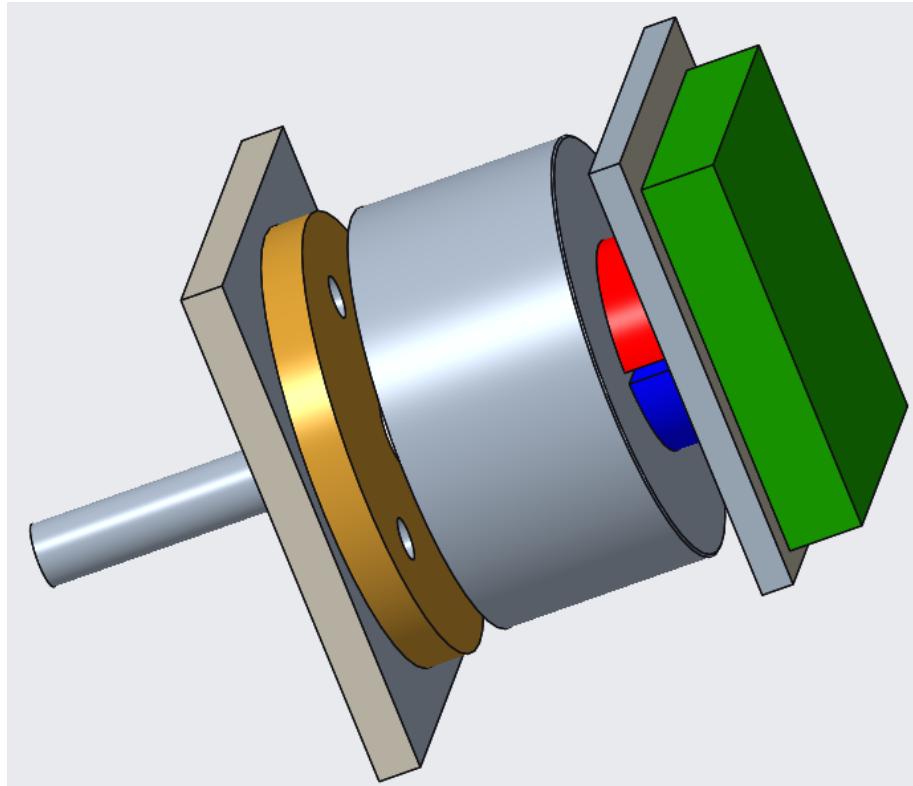


Fig. 2.3 Support plates on stator and sensor.

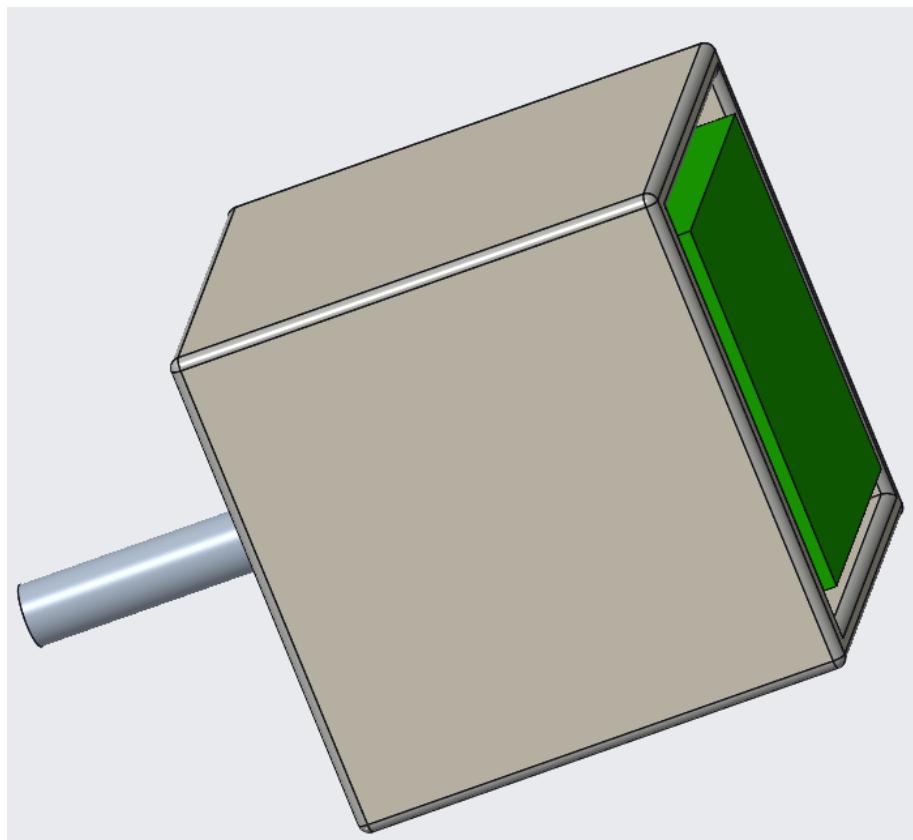


Fig. 2.4 Final housing of the motor.

2.2 First gearbox design: misaligned spur gears

Concerning the gearbox, I was told to design a transmission able to produce as output on the wheel an angular speed 35 times slower than the rotor one. A first idea was to use common spur gears, assembled in a manner such that the volume occupied by the gearbox plus motor was minimized as much as possible in order to let the camera (placed in the center of the main board) have a free cone for looking downward.

The final design, shown in Fig.2.5, presents three axis (in addition to the wheel one) on which six gears transmit the power from the motor to the wheel.

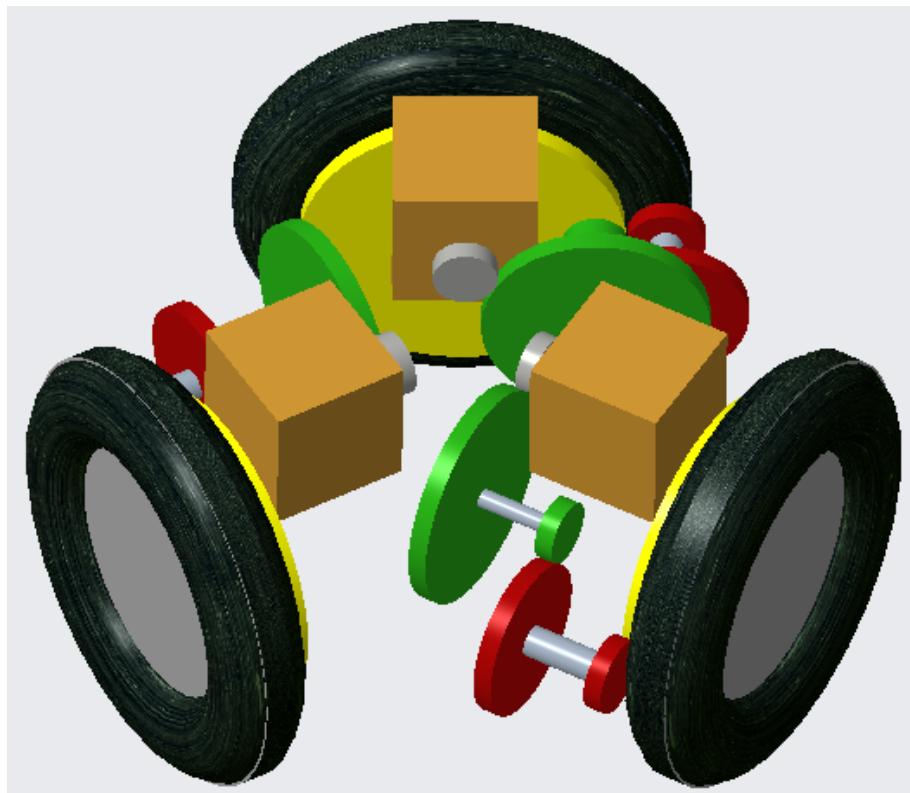


Fig. 2.5 First gearbox conception.

The main feature of this configuration is the misalignment between all axes (see Fig.2.7, which would lead on one hand to a pretty low motor position with enough space for the camera visual cone and on the other hand to a really difficult analysis of the stress distribution among both gears and shafts).

The axial width w of the gearbox in the wheel axial direction, comprising the motor (which in this configuration doesn't affect the measure), is 43,8mm, while the its height h is 35mm as shown in Fig.2.6 and Fig.2.7.

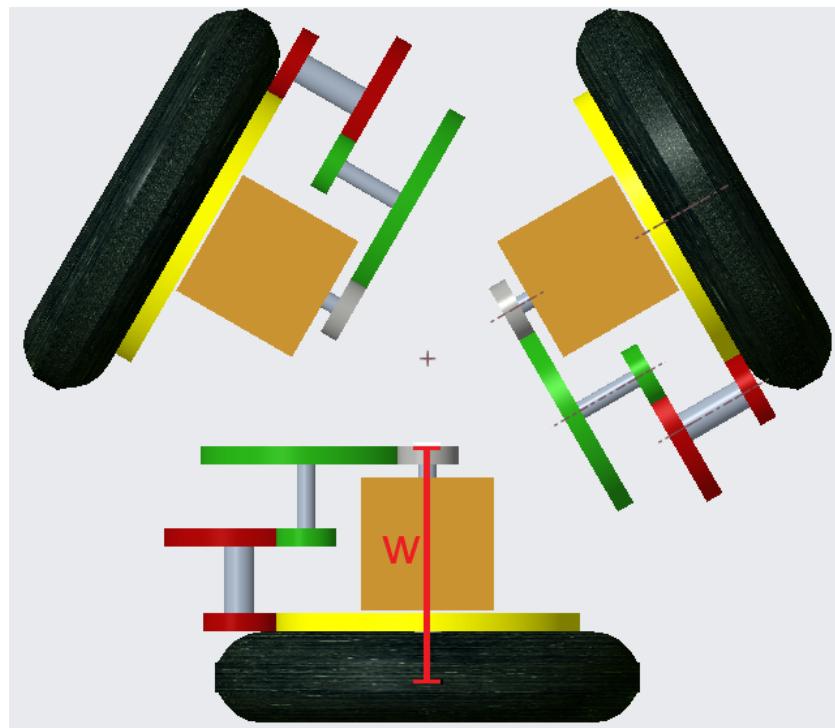


Fig. 2.6 Axial width of the motor block.

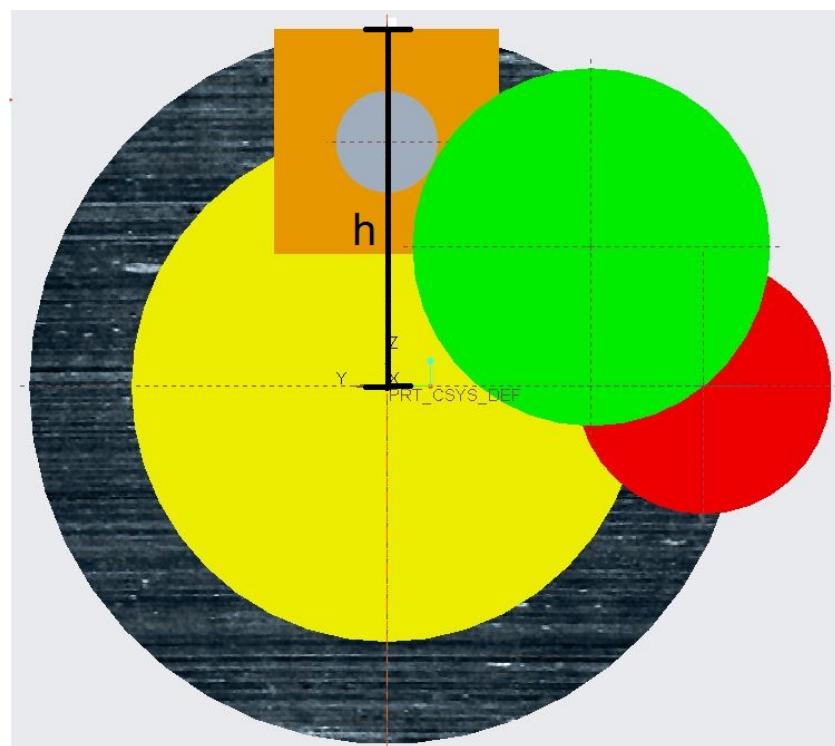


Fig. 2.7 Axes misalignment and motor height.

2.3 Second gearbox design: aligned spur gears

2.3.1 Gears with 0.8 module

To overcome the problems rose from the previous gears arrangement, a new configuration was developed still exploiting only spur gears, but placed on three vertically aligned shafts instead of four. As can be seen in Fig.2.8, the gear attached to the motor shaft transmits the power, then 3 couples of gears freely rotate around their shafts providing most of the reduction, while the last gear is again attached to the wheel shaft to transmit the final torque and speed. In addition, all components were arranged as similar as possible to the contour of the cone needed by the camera.

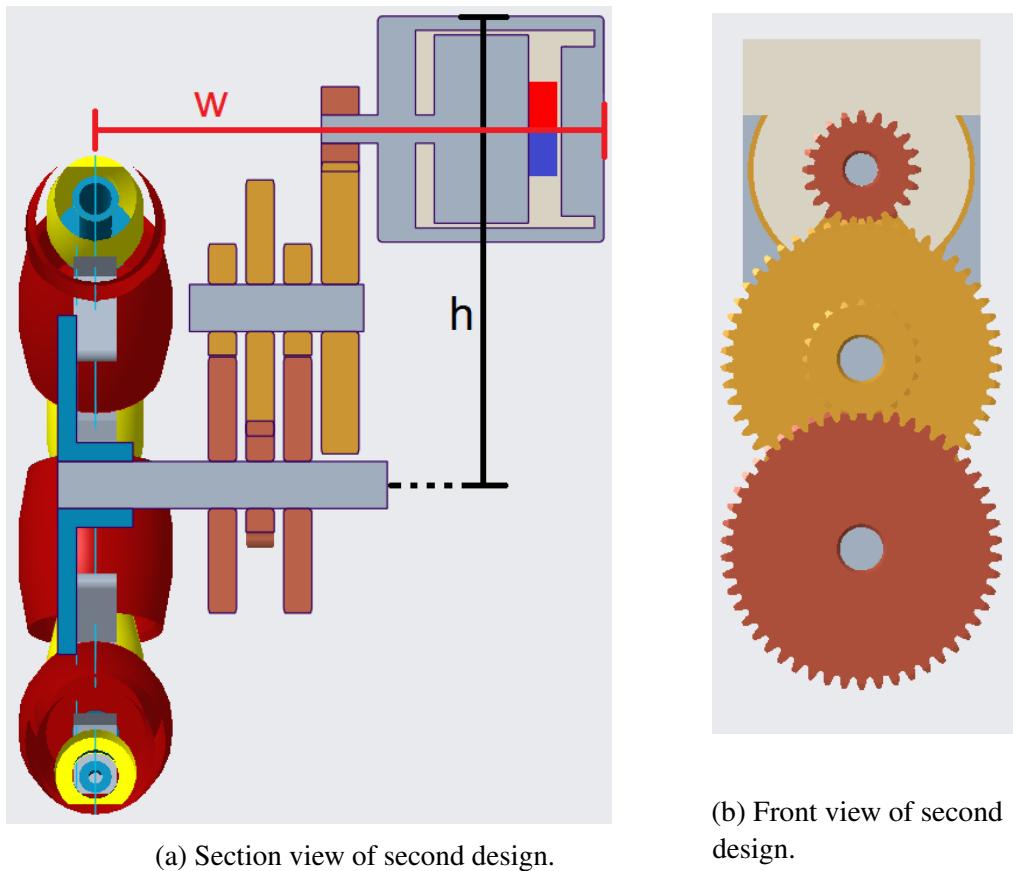


Fig. 2.8 Second gearbox design.

Since this solution is more reliable with respect to the one described earlier, a static stress analysis^[5] was performed in order to guarantee the safety of the transmission. The values in input (motor no-load speed and stall torque) are taken from the data-sheet (see Appendix-B) even if the motor used for e-puck3 is slightly different in some parameters.

These 2 input values cannot be present at the same time but it's useful to verify the gearbox at worst conditions. Therefore, by using steel^[6] gears, a maximum stress of $52,41MPa$ was found on the second last gear, thus much lower than the admissible stress (with safety factor of 2) of the material which was $177,5MPa$. However, it was expected to have maximum stress on the last gears since the reduction applied to the angular speed leads to an increasing of the torque for a constant (out of losses) power, in addition the second last gear has the smallest diameter so the torque is applied on a shorter radius than the last gear one, causing a bigger force on the teeth. A reduction ratio of $i = 36,41$ was obtained with an axial width $w = 53.5mm$ when a proper combination of gears (module 0.8) are taken from catalogues^[7]^[8]. The motor height is instead $h = 49.8mm$. Taking into account the 4 reduction steps and the usage of 5 bearings, the final efficiency reached 73,6%.

2.3.2 Gears with 0.5 module

Considering that the maximum stress on the teeth was less than 30% of the admissible one, the whole design was repeated in order to see whether gears with module 0.5 were feasible or not. Therefore, keeping the same material (steel C45E) and by only looking at Misumi catalogue^[8], a new and smaller gearbox was developed. The maximum stress was found again on the second last gear, with a value of $110,73MPa$, thus still in the admissible range. The reduction ratio is $i = 33,18$ in an axial width of $w = 47.5mm$ and a motor height of $h = 46mm$, while the clearance let by the second gear on the wheel axis (to have the wheel shaft rotating freely) increased from $3,8mm$ to $5mm$.

Another attempt had as target the design of a gearbox so as to have most of the gears below the motor, thus the length of the motor wouldn't have counted, differently from last design. Thus, given some initial parameters (the motor case height, the clearance between second gear and wheel shaft and the clearance between motor case and sixth gear) and assuming to use always the same smaller (red) and bigger (black) gears, it's shown in Fig.2.9b how gears diameters would change depending on the distance between the motor axis and the wheel one.

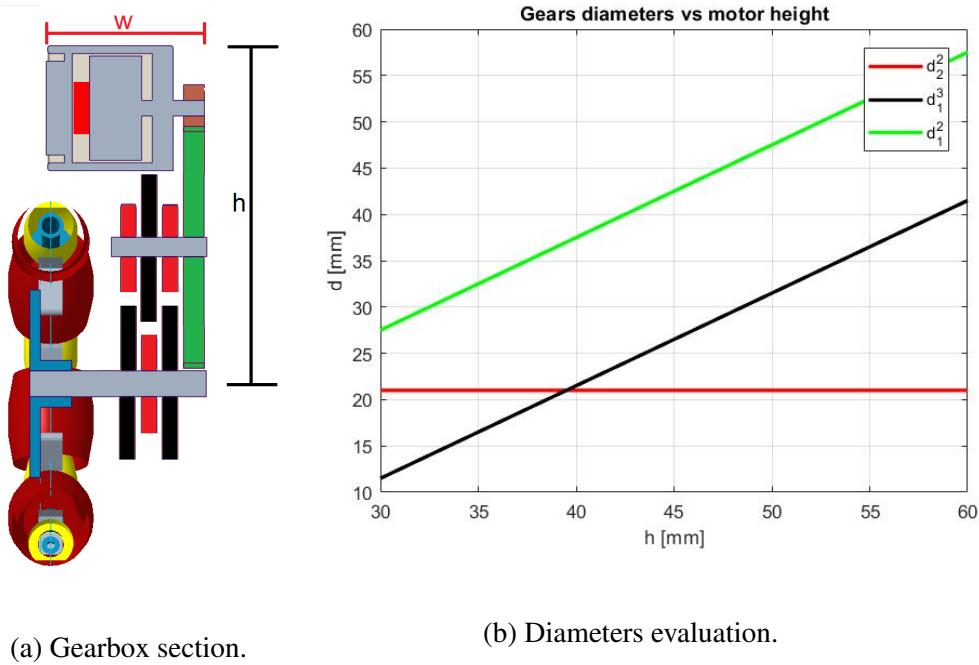


Fig. 2.9 Geometric constraints for gears below the motor.

As result, the speed of the motor starts to be reduced when black gears have bigger diameters than red ones, so i.e. placing the motor shaft at an height of 50mm above the wheel axis, to obtain a reduction of $i \approx 25$, four steps of reduction $\frac{d_1^3}{d_2^2} = 1,5$ are needed in addition to the main step provided by second and first gears (reduction $i = 5$). Therefore, this gearbox concept has been rejected first because by looking at catalogues, the best combination of gears ($d_2^2 = 18\text{mm}$, $d_1^3 = 30$, $d_1^2 = 50$) leads the upper part of the motor case to lie at $h = 66\text{mm}$ above the wheel axis (too much for our robot), second adding more bearings would decrease the efficiency.

2.3.3 Plastic gears

A final attempt was made by trying to exploit plastic gears, indeed the main purpose of this project is to provide modularity of the structure thus a 3D-printed part can be easily modified when needed. Of course printing gears using the commonly available plastics reduces surface finishing and durability of component compared to injection molded or machined plastic parts, but a brief research on the commonly used material for 3D printing shows how Polycarbonate or Nylon may also be a feasible choice for gears. Assuming for Polycarbonate^[9] a Yield tensile strength of 62MPa , a safety factor of 1,5 allows to have a gearbox of $w = 51.5\text{mm}$ axial width, $h = 46\text{mm}$ and a reduction ratio $i = 29,64$. This design

may become really interesting and lightweight alternative to the previous one if one found^[10] a plastic material with good mechanical properties (Yield point around $100MPa$).

2.3.4 Housing design for motor module

At the moment, since the robot has to be modular, an housing comprising the (steel) gearbox and the motor was designed and it's shown in Fig.2.10.

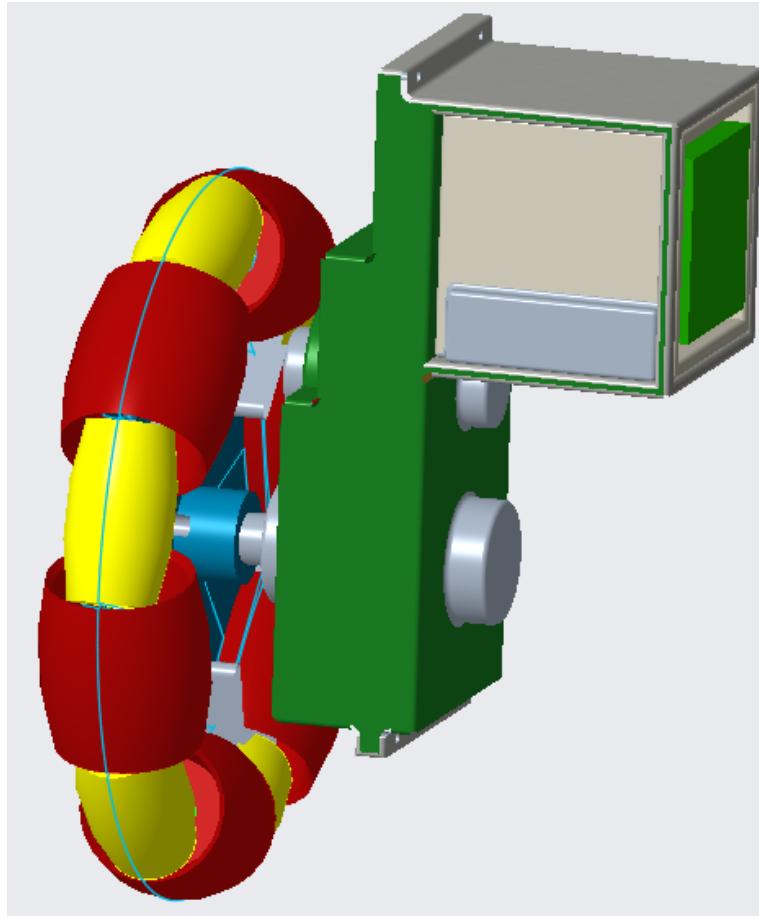


Fig. 2.10 Complete design exploiting a worm gear.

The idea is to have an housing which can be easily attached and detached to the wheel, so its frame was designed in a way that it follows as much as possible the shape of the gearbox to avoid touching the rotating wheel, but it has also enough stiffness to maintain the motor which can be easily removed. The housing is made up of two parts, joined by means of screws in the upper and lower part, plus other screws that lock the motor in the correct position. In order to reduce the weight, even if a stress analysis is still missing, the idea is to 3D-print the whole housing.

2.4 Third gearbox design: worm drive

Here the idea was to use worm gears because they can provide greater reduction ratios (1:100) than spur gears (1:5). To this aim, a single enveloping gear drive (where worm gear is made of a throated shape to wrap around the worm) was selected to have a better face contact between teeth (see Fig.2.11).

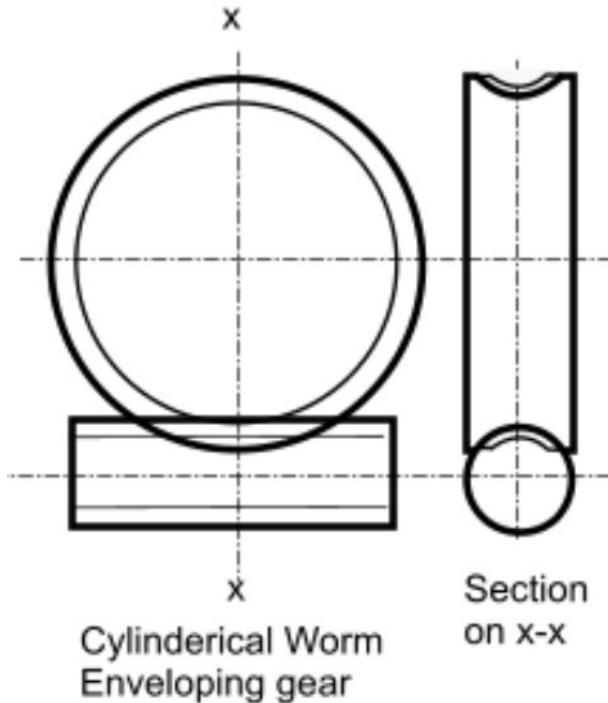


Fig. 2.11 Complete motor block attached to the wheel.

The worm was designed^[11] with double threads to have less stress on a single tooth, but (having a module of 0.5) the diameter of the gear had to be doubled to achieve a reduction ratio of $i = 30$.

As results of the worm gear sliding action, significant frictional losses are generated. A good combination of gear materials, to avoid heating and wear problems, could be an hardened alloy steel worm (ground finished) with a phosphor bronze gear. Eventually, a stress analysis was performed both on the worm^[12] and on the helical gear^[13], with the result that both components teeth can withstand the transmitted torque with a general transmission efficiency of 81,4%.

As one can see in Fig.2.12 was chosen to put the motor still above the wheel axis but in a way that its axis is perpendicular to it, as result the center of mass of the motor is closer to center of the wheel and the height of the motor is 42,3mm with an axial width of 33,5mm.

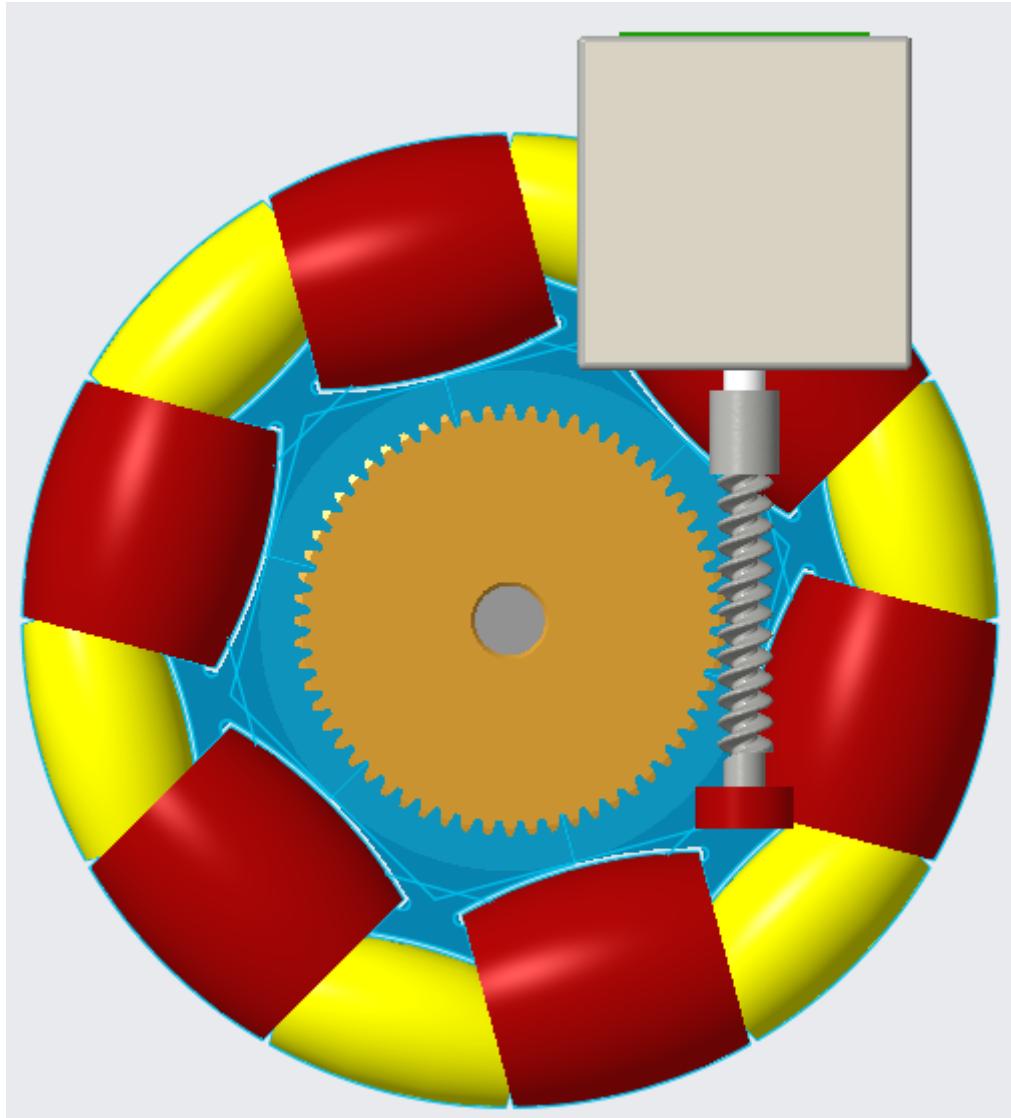


Fig. 2.12 Single enveloping gear drive.

Finally, as for the second design, an housing of the motor block was designed in order to provide facilities during the assembly phases. As shown by Fig.2.13, the housing (probably made of 3D-printed materials) has an upper plate on which the motor is fixed, then 2 lateral screws lock the housing in correspondence of 2 bearings placed at different heights and with angle offset of 90 with each other. Indeed, they support the worm and the gear which are perpendicular.

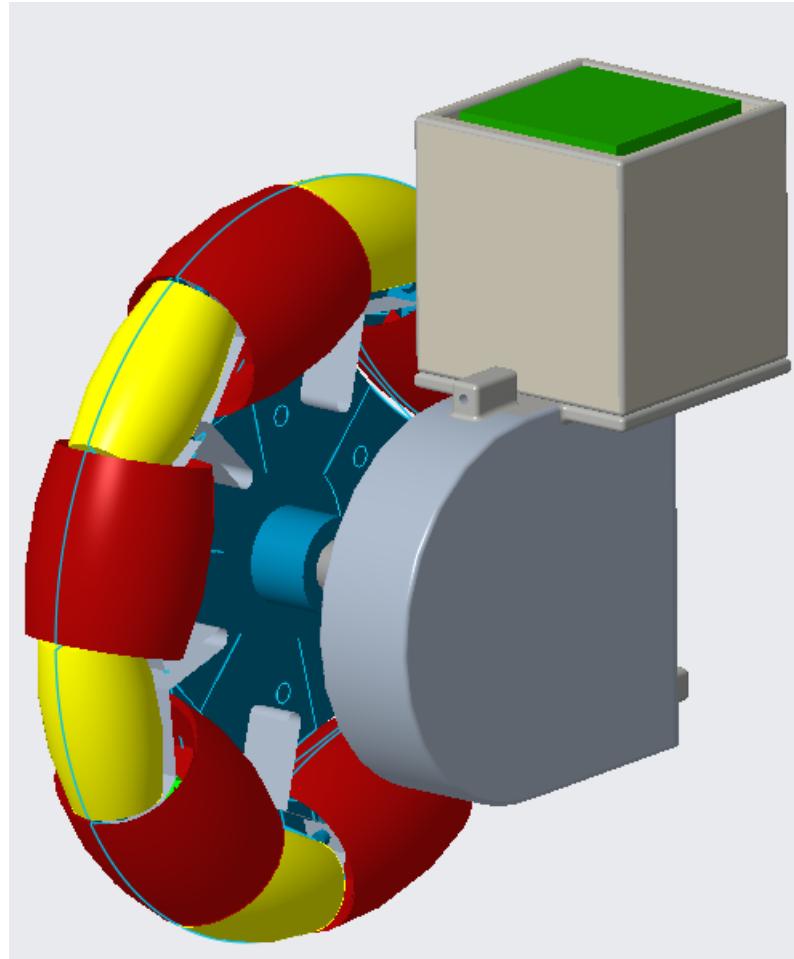


Fig. 2.13 Complete motor block of third design.

2.5 Omniwheel design

The most popular wheeled mobile robots are equipped with two independent driven wheels. Since these robots possess 2 degrees-of-motion (DoMs), they can rotate around a vertical axis, but cannot perform sideways motion. To overcome this limitation, omnidirectional mobile robots were proposed, indeed they are able to move in arbitrary directions without changing wheels orientation, because they can achieve 3 DoM on a two-dimensional plane. Fig.2.14 shows how the main types of omniwheels have gaps between passive rollers and since these gaps cause discontinuous contacts between the wheel and the ground, may occur unwanted vertical and/or horizontal vibrations.



Fig. 2.14 Main types of omniwheel^[14].

With regards to the omniwheel choice, the Continuous Alternate Wheel (CAW) was exploited. This type of wheel makes continuous contact with the ground and has alternating large and small passive rollers placed around the wheel circumference. It has recently been used from ETH of Zurich for their Rezero project^[15], based on a robot able to balance and drive on a single sphere by means of 3 CAWs.



Fig. 2.15 Rezero project by ETH^[15].

Each wheel was 10 cm diameter, but for epuck3 a smaller diameter is requested, possibly around 7 cm, hence a complete wheel design^[16] was made. All four radii (inner and outer) shown in Fig.2.16, have been characterized as function of the angle θ_i for given margins m_h , m_o and m_i , hence they are plotted in Fig.2.17 in which feasible angles start to be present (around 15°) when two conditions are simultaneously satisfied: the dashed green and red lines (minimum r_o and \hat{r}_o) go below the continuous green and red lines (maximum r_o and \hat{r}_o).

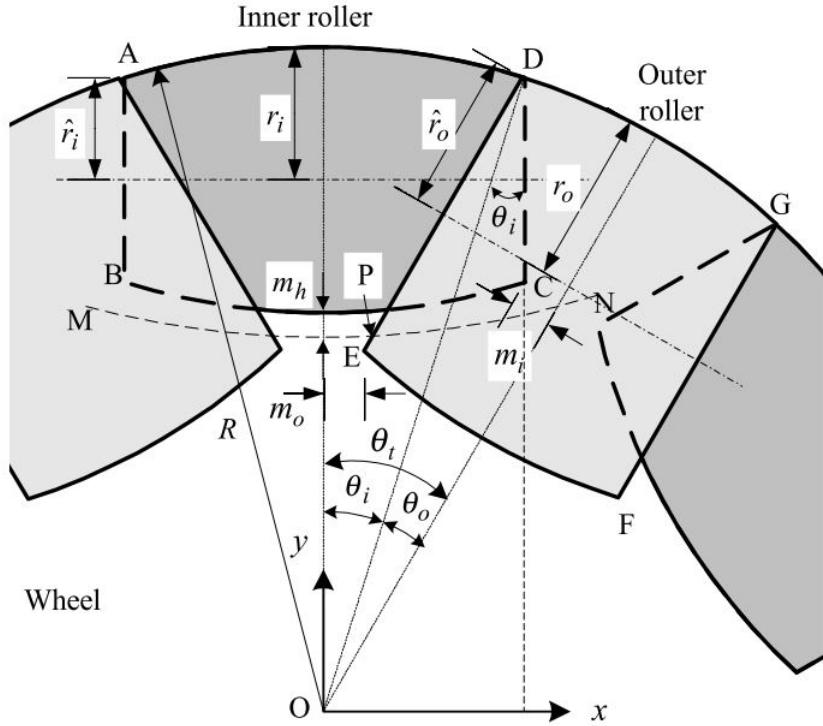


Fig. 2.16 CAW geometric features^[16].

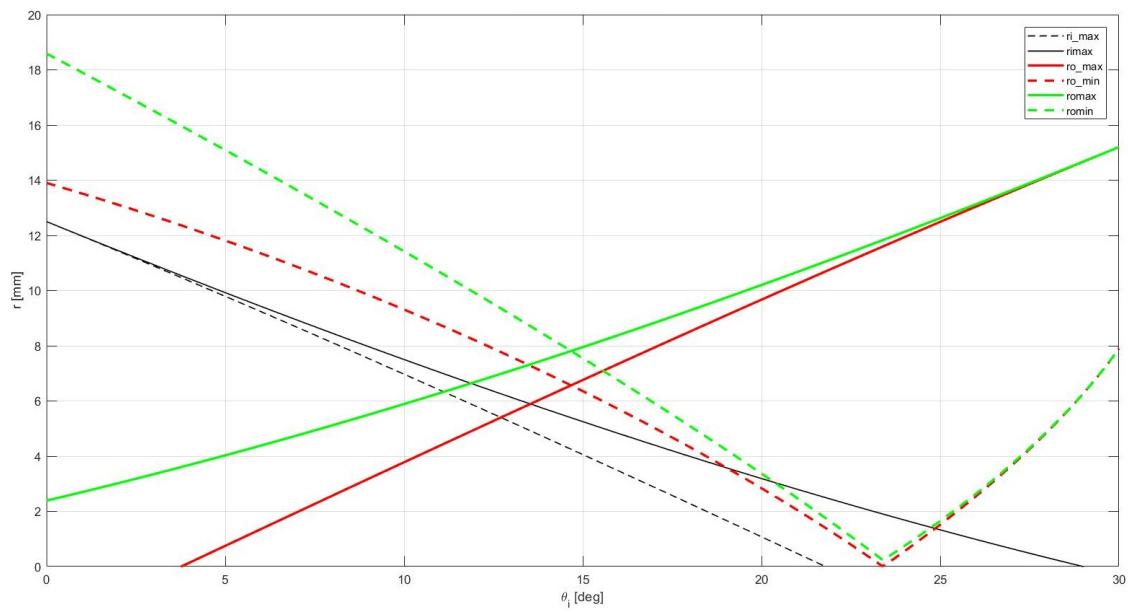


Fig. 2.17 Radii as function of θ_i , for 6 pairs of rollers.

It can be noted from Fig.2.18 that for $n < 6$ the end radius of an inner roller gets smaller while the other radii get larger, larger so bigger gaps appear. For $n > 6$ all the radii tend to decrease, which is not desirable from a manufacturing point of view. Thus, as suggested by the paper guidelines, the number of rollers was chosen as 6.

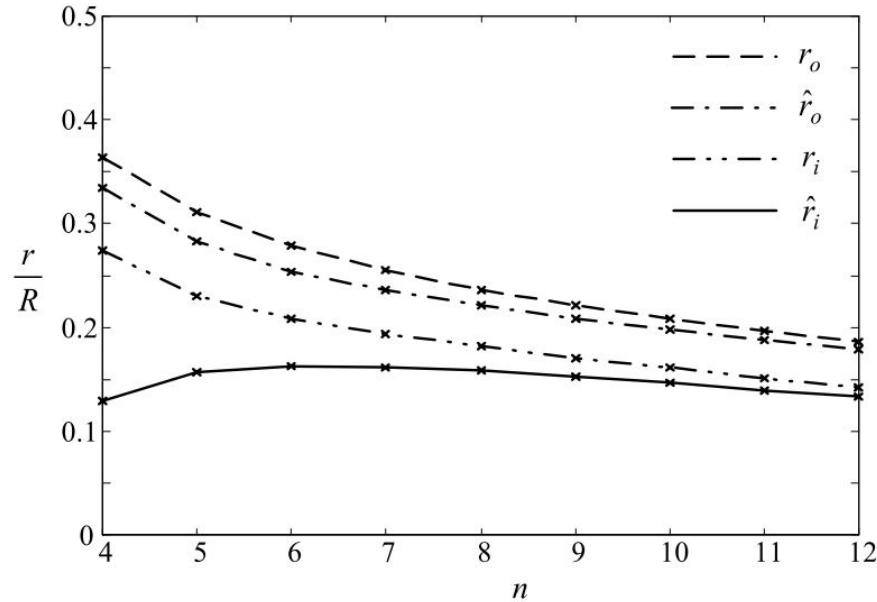


Fig. 2.18 Radii as function of number of rollers^[16].

The main problems of this design are due to manufacturing difficulties regarding the support frame of the inner wheels, indeed for e-puck3 (Fig.2.19) we have just 1.5mm space in the grey supporting frame where to put the shaft, so probably we have to decrease the number of rollers from 6 to 5 in order to increase this space.

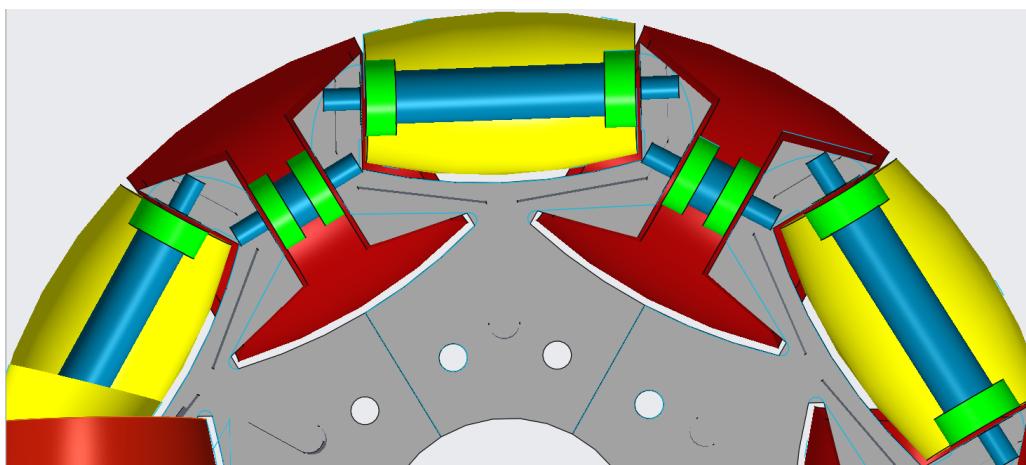


Fig. 2.19 Manufacturing problem for inner roller.

Chapter 3

Two-wheeled inverted pendulum configuration

Among the several configurations, e-puck3 could turn into a two-wheeled inverted pendulum; in this case two independently-actuated wheels are exploited not only for balancing while driving, but also to perform a self-righting maneuver from a horizontal orientation. This part of the project had as goal the validation of the motor choice to perform this self-righting motion from a typical rest position shown by Fig. 3.1.

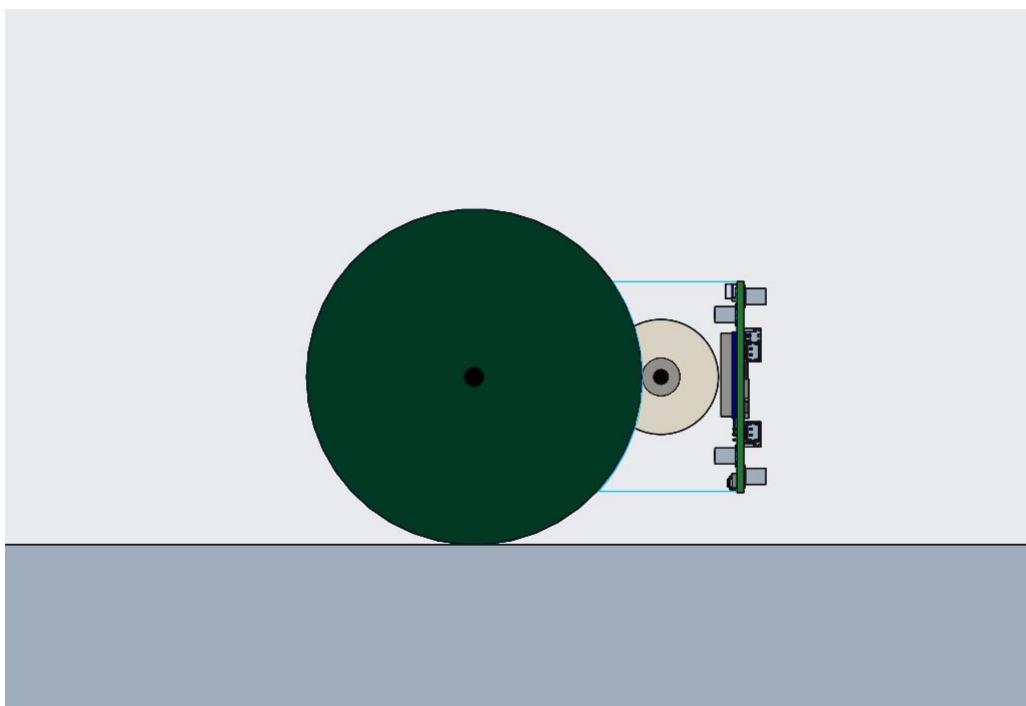


Fig. 3.1 e-puck3 horizontal configuration.

3.1 Modeling the 2WIP

The main issue regarded the maximum torque each motor can transmit to prevent wheels from slipping on the ground. To develop an analytic model of the system dynamic, e-puck3 was modeled as the 2D inverted pendulum shown in Fig. 3.2, where the body became a rod with centre of mass CoM concentrated in a point, distant from the wheel axis of:

$$d = \frac{d_{mb}m_{mb} + d_{ec}m_{ec}}{m_{mb} + m_{ec}}$$

where d_{mb} is the distance between motor block (motor+gearbox+housing) and wheel axes, d_{ec} the distance between electronic components (board + batteries) and wheel axes, while m_{mb} is the motor block mass, m_{ec} is the half of electronics mass because this is distributed on 2 wheels. From now on I will refer to $m_p = m_{mb} + m_{ec}$ as the pendulum mass sustained by each wheel, which corresponds, as said, to the half of the total pendulum weight.

Other important parameters are:

x : position of the wheel axis along the x-axis;

θ : clockwise angle between the positive z-axis and the pendulum CoM;

ϕ : clockwise rotation angle of the wheel from the positive z-axis.

R : wheel radius.

τ : torque applied on each wheel;

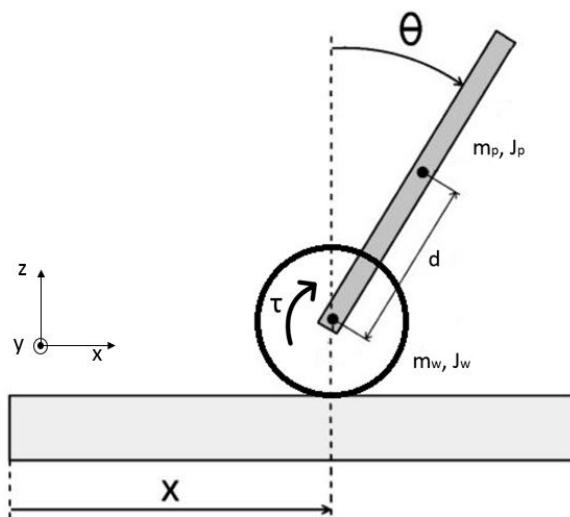


Fig. 3.2 Schematic of the 2D inverted pendulum model.

Two approaches were used to express the dynamic of this system: Lagrangian based and Newton-Euler based. Both are described here and lead, as expected, to the same result.

3.1.1 Lagrangian approach

By means of this approach, each system can be described by the following expression:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} + \frac{\partial D_i}{\partial \dot{q}_i} = F_i \quad (3.1)$$

The terms of this expression stand for:

L : Lagrange function which represents the difference between Kinetic and Potential energy;

q_i : i-th generalized quantity needed for describing the system;

D_i : dissipation function associated to q_i ;

F_i : generalized non-conservative force associated to q_i .

Since was chosen to describe the system by means of its x position and angle θ as generalized coordinates, with no dissipative elements, the previous terms become:

- $L = (K_p + K_w) - (P_p + P_w)$;

- $q_i = \begin{bmatrix} x \\ \theta \end{bmatrix}$

- $D_i = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

- $F_i = \begin{bmatrix} F_x \\ F_\theta \end{bmatrix}$

where the terms stand for:

K_p : pendulum kinetic energy;

K_w : wheel kinetic energy;

P_p : pendulum potential energy;

P_w : wheel potential energy;

Since the model is studied in a 2D space (assuming no steering wheels), the pose p_p of the pendulum and p_w of the wheel are:

$$p_p = \begin{bmatrix} x + d \sin \theta \\ d \cos \theta \end{bmatrix}$$

$$\mathbf{p}_w = \begin{bmatrix} x \\ 0 \end{bmatrix}$$

and their derivative with respect to time are:

$$\dot{\mathbf{p}}_p = \begin{bmatrix} \dot{x} + d\cos\theta \\ -d\sin\theta\dot{\theta} \end{bmatrix}$$

$$\dot{\mathbf{p}}_w = \begin{bmatrix} \dot{x} \\ 0 \end{bmatrix}$$

thus, by substituting Lagrangian terms, one has:

$$K_p = \frac{1}{2}m_p\dot{p}_p^2 + \frac{1}{2}J_p\dot{\theta}^2$$

$$K_w = \frac{1}{2}m_w\dot{p}_w^2 + \frac{1}{2}J_w\dot{\phi}^2$$

$$P_p = m_p g d \cos\theta$$

$$P_w = 0$$

replacing the angle ϕ as $\phi = \frac{x}{R}$, the full Lagrange function becomes:

$$L = \frac{1}{2}(m_p + m_w + \frac{J_w}{R^2})\ddot{x}^2 + m_p\dot{x}d\cos\theta\dot{\theta} + \frac{1}{2}(m_p d^2 + J_p)\dot{\theta}^2 - m_p g d \cos\theta \quad (3.2)$$

With this expression, substituted in 3.1, is possible to obtain the system dynamic equations of motion.

Thus, with $q_1 = x$ and $q_2 = \theta$, one has:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} = (m_p + m_w + \frac{J_w}{R^2})\ddot{x} + m_p d \cos\theta \ddot{\theta} - m_p d \sin\theta \dot{\theta}^2$$

$$-\frac{\partial L}{\partial x} = 0$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} = m_p d^2 \ddot{\theta} + m_p d \cos\theta \ddot{x} - m_p d \dot{x} \sin\theta \dot{\theta} + J_p \ddot{\theta}$$

$$-\frac{\partial L}{\partial \theta} = -(-m_p d \dot{x} \sin\theta \dot{\theta} + m_p d g \sin\theta)$$

$$\frac{\partial D_x}{\partial \dot{x}} = 0 ; \frac{\partial D_\theta}{\partial \dot{\theta}} = 0$$

$$F_x = \frac{\tau}{R}$$

$$F_\theta = -\frac{\tau}{R}(R + r)$$

The reason behind the definition of the two generalized forces will be made clearer in the Newton-Euler approach, for the moment is enough to say that the external force acting on the system is the one produced by the motor, which through the gearbox puts in rotation the wheel and in counter-rotation the pendulum (robot body). The different directions of rotation are a consequence of the Newton 3rd law of motion, indeed the exchanged force F_t on the gearbox teeth is sensed with different direction on the two matching gears, the one turning with the motor and the one turning with the wheel. Due to the presence of a gearbox, the point of application of the force acting on the pendulum is different from the one leading the wheel (motor axis instead of gear teeth).

Thus the total virtual work, acting on virtual displacements is:

$$\partial W = F_t \partial(s_w + s_p) = F_t \partial[x - (R + r)\theta] = \frac{\tau}{R} \partial[x - (R + r)\theta] = \frac{\tau}{R} \partial x - \frac{\tau}{R} (R + r) \partial \theta \quad (3.3)$$

where s_w and s_p are the virtual displacements of wheel and pendulum with respect to wheel axis and r is the motor gear radius (see section 3.1.2).

Hence, as result, the system dynamic (from eq.3.1) is described by the following two non-linear differential equations:

$$(m_p + m_w + \frac{J_w}{R^2})\ddot{x} + (m_p d \cos \theta) \ddot{\theta} - m_p d \sin \theta \dot{\theta}^2 = \frac{\tau}{R} \quad (3.4)$$

$$(m_p d \cos \theta) \ddot{x} + (m_p d^2 + J_p) \ddot{\theta} - m_p d g \sin \theta = -\frac{\tau}{R} (R + r) \quad (3.5)$$

3.1.2 Newton-Euler approach

By exploiting the equations coming from classic mechanics, in a generic instant, the system can be represented as in Fig. 3.3:

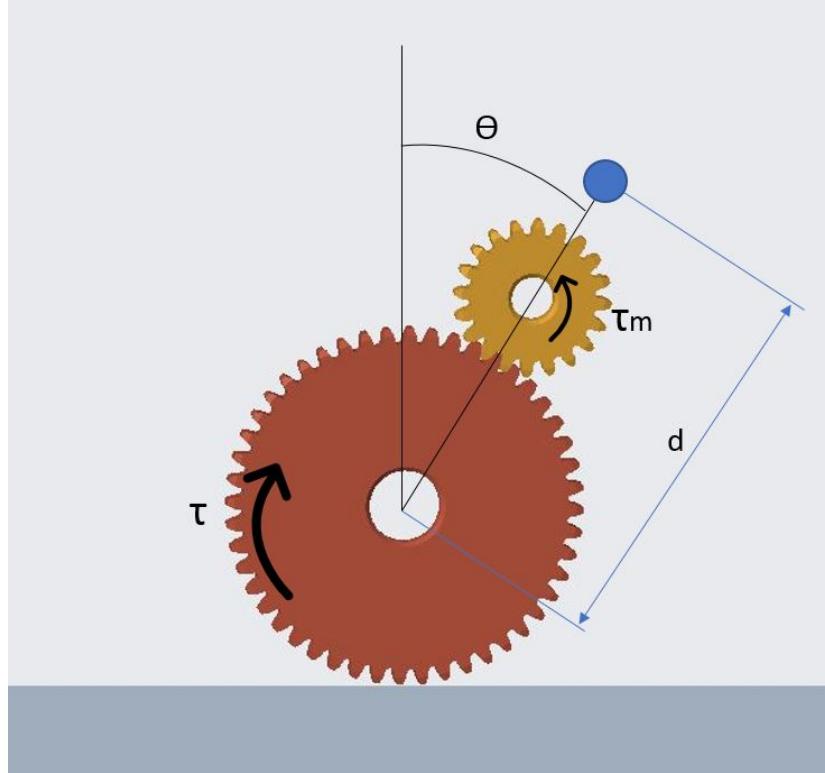


Fig. 3.3 Schematic of the system.

where the motor torque τ_m acts on the first gear of the gearbox (here for simplicity is assumed as a 1-stage gearbox) and, as a consequence, a torque τ (increased by gearbox reduction ratio) is provided to the last gear turning with the external robot wheel. The pendulum center of mass is at distance d from the wheel axis, as already introduced in section 3.1.

Regarding the forces which interact with this system, an important detail is shown in Fig.3.4, where the contact between teeth produces a tangential force F_{tm} on the motor gear equal and opposite to F_t acting on the wheel gear. Fig.3.4 also shows that by means of motor shaft, the force F_{tm} is transmitted to the motor housing in correspondence of its axis (not on teeth contact point) and consequentially to the pendulum body. Hence imagining the wheel gear blocked, the rotation of the pendulum is provided by a torque which can be expressed as product of F_{tm} and the distance between the wheel axis and the motor axis, thus $R+r$ with R the radius of the wheel gear (here equal to wheel radius) and r the radius of the motor gear.

Since F_{tm} is shared with the wheel gear, we can write:

$$\frac{\tau_m}{r} = F_{tm} = \frac{-\tau}{R} \Rightarrow F_{tm}(R + r) = \frac{-\tau}{R}(R + r) \quad (3.6)$$

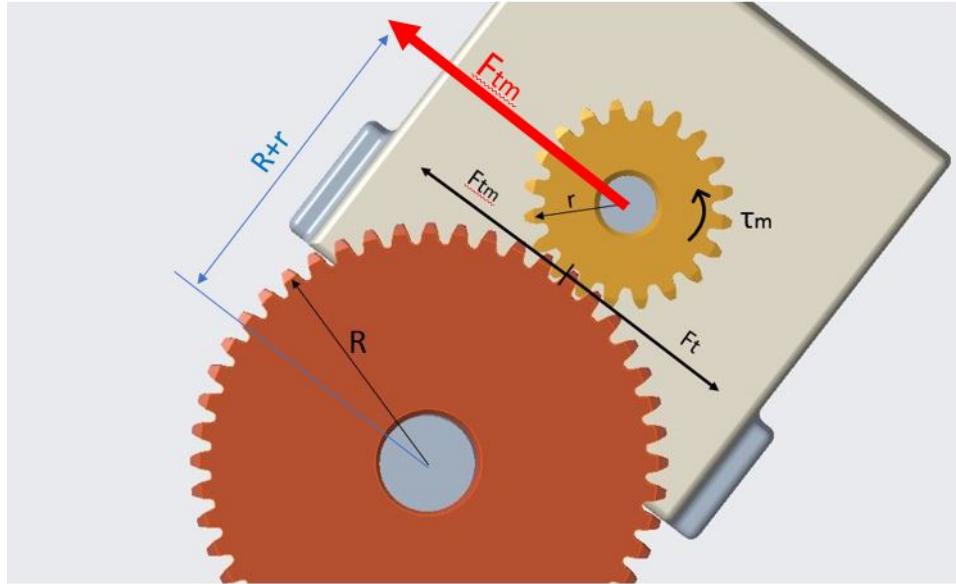


Fig. 3.4 Tangential forces on teeth contact point.

Said that, it's possible to define the equations of motion for both the two subsystems (pendulum in Fig.3.5 and wheel in Fig.3.6, in which the forces drawn in red are inertial forces).

In the following expressions the tangential force $F_{tm} = F_t$ acting on teeth will be decomposed in its vertical and horizontal components, respectively $T_z = F_t \sin \theta$ and $T_x = F_t \cos \theta$. The equilibrium of the pendulum is defined along the horizontal axis x, the vertical axis z and around the wheel rotation axis y as:

$$x\text{-axis:} \quad -T_x - m_p \ddot{x} - m_p d \cos \theta \ddot{\theta} + m_p d \sin \theta \dot{\theta}^2 = 0 \quad (3.7)$$

$$z\text{-axis} \quad T_z - m_p g + m_p d \sin \theta \ddot{\theta} + m_p d \dot{\theta}^2 \cos \theta = 0 \quad (3.8)$$

$$y\text{-axis} \quad F_{tm}(R + r) + J_p \ddot{\theta} + m_p d \cos \theta \ddot{x} + m_p d \dot{\theta}^2 - m_p g d \sin \theta = 0 \quad (3.9)$$

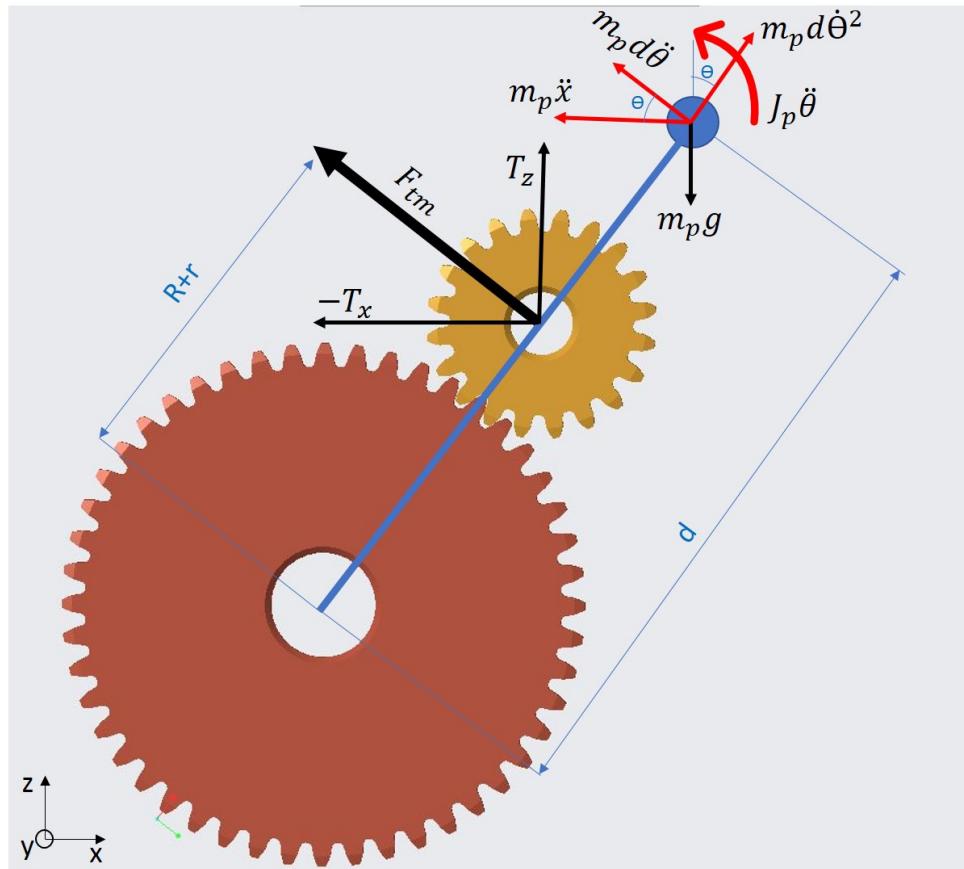
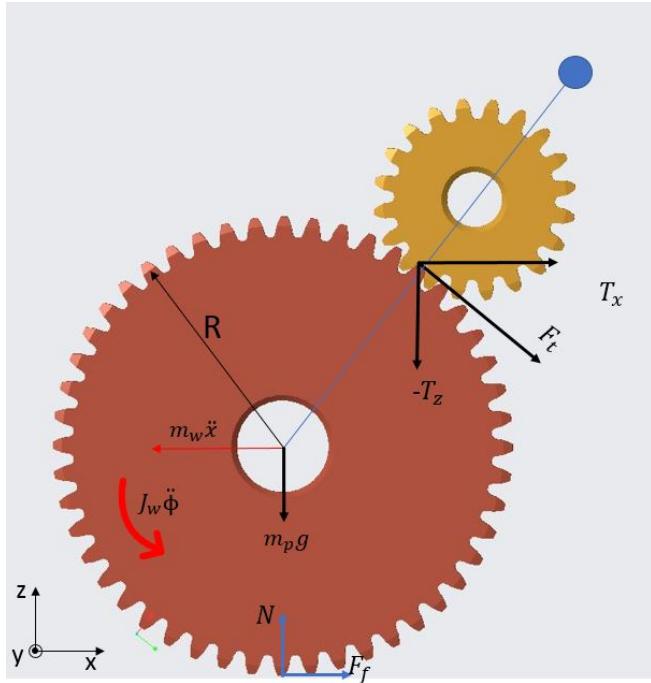


Fig. 3.5 Pendulum free body diagram.

With the same approach, but considering a reversed tangential force F_t and adding the ground reactions forces N and F_f , the equations for the wheel are:



$$x : \quad T_x - m_w\ddot{x} + F_f = 0 \quad (3.10)$$

$$z : \quad N - m_wg - T_z = 0 \quad (3.11)$$

$$y : \quad J_w\ddot{\phi} + F_fR - F_tR = 0 \quad (3.12)$$

Fig. 3.6 Wheel free body diagram.

Now, substituting the wheel angular acceleration $\ddot{\phi}$ with the term $\frac{\ddot{x}}{R}$, in order to obtain the first generalized coordinate, and know that $F_tR = \tau$, the equation 3.12 becomes:

$$J_w\frac{\ddot{x}}{R} + F_fR = \tau \quad (3.13)$$

where the term F_fR can be expressed using equation 3.10, which depends on equation 3.7. Thus, after few substitutions, the wheel final equation of motion is:

$$(m_p + m_w + \frac{J_w}{R^2})\ddot{x} + (m_p d \cos \theta)\ddot{\theta} - m_p d \sin \theta \dot{\theta}^2 = \frac{\tau}{R} \quad (3.14)$$

The other equation of motion, from pendulum side, can be derived from equation 3.9 (keeping in mind the important relation expressed in equation 3.6), thus:

$$(m_p d \cos \theta)\ddot{x} + (J_p + m_p d)\ddot{\theta} - m_p g d \sin \theta = -\tau \frac{(R+r)}{R} \quad (3.15)$$

Chapter 4

Friction analysis

4.1 System parameters correlation

The horizontal configuration shown in Fig.4.1 displays the main forces and torque acting on the system in static conditions. This analysis leads to the numerical definition of both the torque τ_{min} needed to maintain the pendulum body and the maximum torque τ_{max} before slipping on the ground.

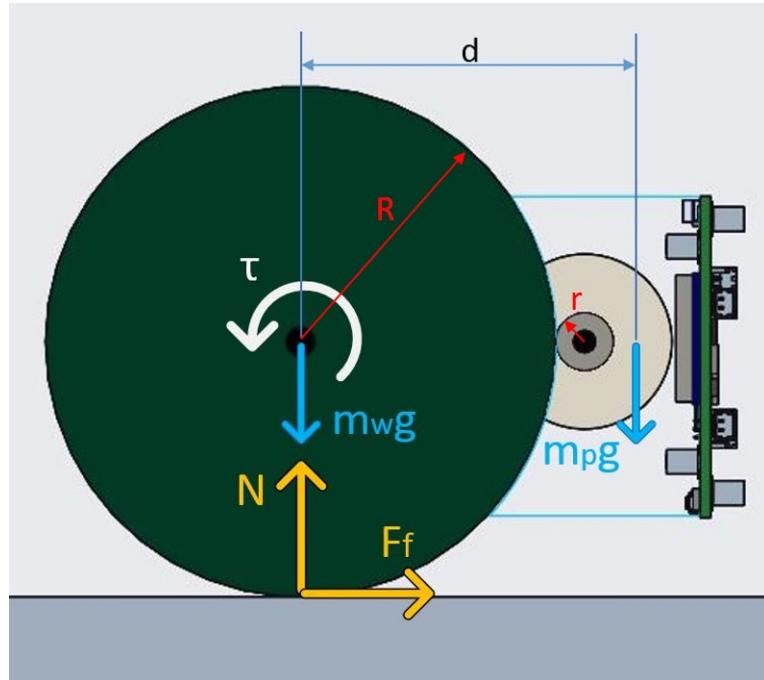


Fig. 4.1 Static forces on e-puck3.

Symbols in Fig.4.1 have already been defined in Chapter 3, and just for clarity, the torque τ^* here refers to the absolute value of the torque $F_{tm}(R+r)$ applied by the wheel on the pendulum body.

For the rotational equilibrium of the pendulum one can write:

$$\tau^* = m_p g d \quad (4.1)$$

This value is then referred to the wheel torque τ , remembering the relation found in eq.3.6:

$$\tau^* = F_{tm}(R+r) = \tau \frac{(R+r)}{R} = m_p g d \quad (4.2)$$

This equality gives the condition for the torque to apply on the wheel to sustain the pendulum, meaning that an up-righting motion is obtained if and only if:

$$\tau_{min} > \frac{R}{(R+r)} m_p g d \quad (4.3)$$

For the vertical and rotational equilibrium of the wheel:

$$N - m_w g - m_p g = 0 \quad (4.4)$$

$$- \tau + F_f R = 0 \quad (4.5)$$

The wheel will have grip on the ground until:

$$\tau \leq \mu_s N R = \mu_s (m_w + m_p) g R \quad (4.6)$$

where μ_s is the static friction coefficient between wheel and ground.

Substituting eq.4.3 in eq.4.6, one obtains:

$$\frac{m_w}{m_p} \geq \frac{d}{\mu_s (R+r)} - 1 \quad (4.7)$$

This condition express the mass ratio between wheel and pendulum to prevent slipping.

An interesting property of this relation is the following:

- for $0 < \frac{d}{\mu_s (R+r)} < 2$, wheel mass can be lower than pendulum one;
- for $\frac{d}{\mu_s (R+r)} \geq 2$, wheel mass has to be greater than pendulum one;

Conditions not considering simultaneously m_p and d , are derived from eq.4.3 and eq.4.6:

$$(m_w + m_p) \geq \frac{\tau}{g\mu_s R}$$

$$m_w \geq \frac{\tau}{gR} \left(\frac{1}{\mu_s} - \frac{R+r}{d} \right)$$

4.2 Gearbox variation

By exploiting e-puck2 parameters^[17], it turned out that the torque capable of uprightness the system is more than 4 times the maximum torque, this means e-puck2 couldn't have been able to upright autonomously because of slipping. Therefore, different parameters listed below are used to have a kind of dimensions reference for the final prototype.

- $m_p = 0.150\text{kg}$ [half of pendulum mass]
- $m_w = 0.070\text{kg}$
- $d = 35\text{mm}$
- $R = 40\text{mm}$
- $r = 5\text{mm}$
- $\mu_s = 0.7$
- $\tau_{min} = 45.8\text{mNm}$ [minimum torque to apply on pendulum]
- $\tau_{max} = 60.4\text{mNm}$ [maximum torque before slipping]

Having these reference value for e-puck3, and i.e. Maxon EC21 motor (nominal torque $T_{nom}^{EC21} = 9.5\text{mNm}$) providing $\tau = 58\text{mNm}$ on the wheel, the gearbox reduction ratio i becomes $i = 6.10$ neglecting mechanical losses. Since this motor has been customized for EPFL, is preferable to deal with a standard version (Maxon EC20 flat) for which one has a more detailed data-sheet; for this reason, assuming to use EC20 motor version, the gearbox design have to deal with a new nominal torque:

$$T_{nom} = 7.59\text{mNm}$$

With the hypothesis that the requested torque on wheel is still $\tau = 58\text{mNm}$, with a gearbox efficiency of $\eta = 95\%$, the reduction ratio becomes:

$$i = \frac{\tau}{T_{nom}\eta} = 8.04$$

This value it's clearly different from the ratio of 35 exploited in Chapter 2, and the reason behind that is shown by Fig.4.2. Here speed and torque of the motor used for e-puck2 (the same of MIP robot) are plotted with dashed lines, while continuous lines refer to e-puck3 motor (Maxon EC20). Even though both speeds almost have the same trend, the torque developed on e-puck3 increases more when increasing the reduction ratio. Therefore, to obtain with e-puck3 more or less the same performances of e-puck2, the reduction has to be decreased and the value of $i = 8.04$ handle not only this aspect but also slipping condition and weight difference between the two models.

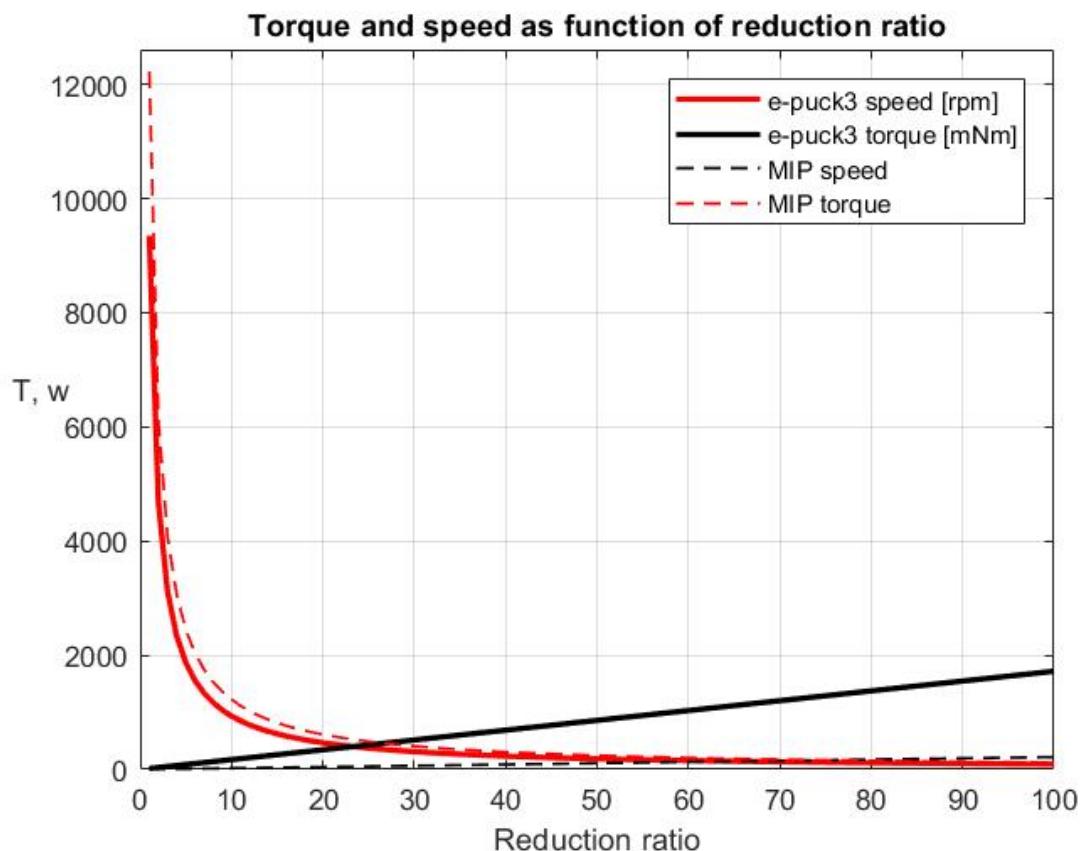


Fig. 4.2 Difference between e-puck2 and e-puck3 motor.

Knowing that, the following 3 gearboxes were designed:

- **Worm gear set:** single-envelope, 2 threads on worm,

- module $m = 0.5$
- worm diameter $d_1 = 5\text{mm}$
- gear diameter $d_1 = 8\text{mm}$
- reduction $i = 8$
- gearbox height $h = 32\text{mm}$

- **Spur gears, single-stage:**

- module $m = 0.5$
- $z_1 = 17$
- $z_2 = 135$
- reduction $i = 7.94$
- gearbox height $h = 48.5\text{mm}$

- **Spur gears, double-stage:**

- module $m = 0.5$
- $z_1 = 15$
- $z_2 = 44$
- module $m = 0.8$
- $z_3 = 15, z_4 = 42$
- reduction $i = 8.02$
- gearbox height $h = 33.4\text{mm}$

For this design bigger module on the last two gears allows customization through 3D-printing, which however requires (due to printability issues) modules $m > 0.5$.

Pinion minimum number of teeth z_{min} at each transmission stage, as function of reduction ratio i of the stage, and Φ the pressure angle, was determined by the following formula:

$$z_{min} = \frac{2i + 2\sqrt{i^2 + (1+2i)\sin^2\Phi}}{(1+2i)\sin^2\Phi}$$

4.3 Slipping evaluation

Furthermore, it's important to notice in Fig.4.3 the role played by the static friction coefficient μ_s . Indeed, fixed d , R and r , for lower values of μ_s the minimum mass ratio $\frac{m_w}{m_p}$ becomes higher to prevent slipping (see eq.4.7). A ratio between two positive quantities (masses) cannot be negative, thus an unfeasible region is present.

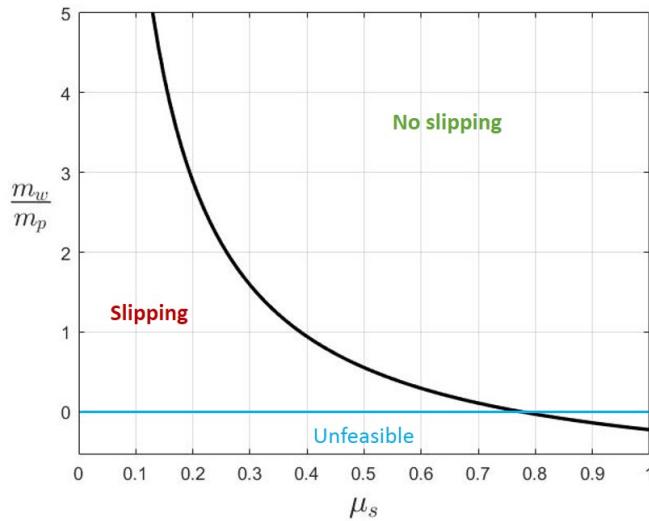


Fig. 4.3 Minimum mass ratio for different μ_s .

Fig.4.4 shows instead how, fixed μ_s , R and r , the minimum ratio $\frac{m_w}{m_p}$, as expected, increases with the distance d of the pendulum CoM.

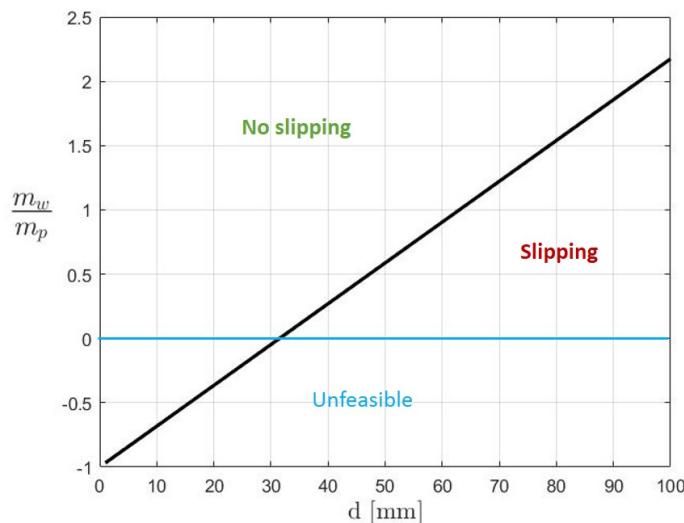


Fig. 4.4 Minimum mass ratio variation for different d .

This behavior is justified by eq.4.3 where, for a given mass m_p the torque τ increases with the distance and consequentially the horizontal ground reaction force F_f in eq.4.5 has to increase, but since its value is given by: $F_f \leq \mu_s(m_w + m_p)g$, the only way to avoid slipping is to increase the wheel mass m_w .

This latter in-equation provides an additional, and general, consideration on F_f which is proportional to the normal force N of the ground acting upon the robot wheel. This force N is the total robot mass distributed among the number of wheels, thus focusing on a single wheel the resulting normal force can be stated as the product of pressure times contact area: $N = pA$. For a wide wheel, the area is large but the pressure is small and the opposite happens for a narrow wheel. This means that the force of friction F_f is the same whether the wheel is wide or not. However, ground is not a uniform surface, thus having a wider wheel firstly increases the probability of having ground contact; secondly, the robot support base is increased making it harder to turn the robot over.

As for Fig.4.3, also here we have a possible negative minimum ratio, which leads to the same considerations discussed before.

Chapter 5

Controller design and simulation

Results found in Chapter 3, show that the system dynamic is nonlinear and, in addition, the system has a single input (τ) and multiple output ($x, \theta, \dot{x}, \dot{\theta}$) thus it's referred as a SIMO system.

In order to solve the non-linear equations governing the system, one may develop a non-linear control strategy such as Model Predictive Control (MPC)^[18], but the required computational power and memory storage would have a more significant impact on the electronics than other simpler techniques.

Since the goal of this study is to verify whether or not the motor is able to self-right the robot, a PID controller (controlling tilt angle θ) was used for the majority of the motion, indeed a Linear Quadratic Regulator (LQR)^[19] was then implemented just around the equilibrium point ($\theta = 0$, pendulum upright), in order to control simultaneously all the four states and to bring back the robot in case it had moved along x direction.

5.1 LQR design

The LQR method (in infinite-horizon) is based on the generic cost function shown below:

$$J = \int_0^{\infty} (x^T Q x + u^T R u + 2x^T N u) dt$$

State deviation from desired value represents a cost adjustable through the following factors:

- Q is a vector of weight factors corresponding to each state. If one wants to keep a state as fix as possible, the corresponding factor must be higher in regard to other factors.
- R is the input factor. If set low in regard to other factors, the system will use large input and will be more reactive. However, the real input may saturate or be destructive.

- N combines both comportment but is often set to 0, as for these simulations.

The feedback control that minimizes the cost function value is: $u = -Kx$

where K is the feedback gain that, in case of a continuous time system, can be determined in Matlab by the command "K=lqr(A,B,Q,R,N)".

All the simulations presented in this chapter are performed with a fixed-step size of 5e-05 for Simulink solver. This value corresponds to the one computed automatically in the last simulation (pendulum + real motor), thus for a better comparison also the other simulations have been carried out with the same solver setting.

5.2 Model linearization

LQR approach requires a linear model to compute the most efficient gain for controlling the system, thus the obtained equations were linearized through an approach called "Small signal linearization" which consist of approximating a function by a straight line which passes through a chosen operating point (or nominal state) \bar{x} , thus, by introducing Taylor series expansion at first order, a generic function $f(x)$ can be approximated as:

$$f(x) \approx f(\bar{x}) + \frac{d}{dx}f(\bar{x})\tilde{x}$$

where $\tilde{x} = x - \bar{x}$ is defined as the difference between the current state and the nominal one. Is noteworthy that this approach is valid only for small deviations \tilde{x} , therefore, it was decided to apply LQR control within a maximum deviation of 8° with respect to the equilibrium point, thus $-8^\circ < \theta < 8^\circ$. For a complete robot equilibrium, also the other states have to be 0, hence the full nominal state vector is:

$$X_{nom} = \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

5.2.1 State-space representation

To obtain a linear state-space representation of the system, the nonlinear states equations are developed from equations 3.14 and 3.15 so that only a second order derivative (\ddot{x} or $\ddot{\theta}$) is present at a time:

$$\ddot{x} = \frac{m_p d \sin \theta \dot{\theta}^2 - \frac{(m_p d)^2}{(m_p d^2 + J_p)} g \sin \theta \cos \theta + \tau \left[\frac{1}{R} + \frac{m_p d \cos \theta}{m_p d^2 + J_p} \frac{(R+r)}{r} \right]}{(m_p + m_w + \frac{J_w}{R^2}) - \frac{(m_p d \cos \theta)^2}{m_p d^2 + J_p}}$$

$$\ddot{\theta} = -\left(\frac{m_p d \cos \theta}{m_p d^2 + J_p}\right) \frac{m_p d \sin \theta \dot{\theta}^2 - \frac{(m_p d)^2}{(m_p d^2 + J_p)} g \sin \theta \cos \theta + \tau \left[\frac{1}{R} + \frac{m_p d \cos \theta}{m_p d^2 + J_p} \frac{(R+r)}{r} \right]}{(m_p + m_w + \frac{J_w}{R^2}) - \frac{(m_p d \cos \theta)^2}{m_p d^2 + J_p}} + \frac{m_p g d \sin \theta}{m_p d^2 + J_p} - \frac{\tau(R+r)}{(m_p d^2 + J_p)r}$$

Then, computing the Jacobians of these non-linear equations with respect to states and input, and evaluating them at the nominal state X_{nom} and nominal input $\tau_{nom} = 0$, is possible to define the state matrix A and input matrix B of the State-space model shown below:

$$\dot{X} = AX + Bu$$

$$y = CX + Du$$

where:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-R^2(gm_p^2d^2 - m_p^2d^3\dot{\theta}^2 - J_pdm_p\dot{\theta}^2)}{J_p(m_pR^2 + m_wR^2 + J_w) + m_pd^2(m_wR^2 + J_w)} & 0 & 0 \\ 0 & \frac{R^2(gm_p^2d - m_p^2d^2\dot{\theta}^2 + m_p m_w dg) + J_w m_p dg}{J_p(m_pR^2 + m_wR^2 + J_w) + m_pd^2(m_wR^2 + J_w)} & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{R(m_p d^2 + m_p d R + J_p + m_p d r)}{J_p(m_p R^2 + m_w R^2 + J_w) + m_p d^2(m_w R^2 + J_w)} \\ -\frac{(R^2 + Rr)(m_p + m_w) + m_p d R + J_w(\frac{r}{R} + 1)}{J_p(m_p R^2 + m_w R^2 + J_w) + m_p d^2(m_w R^2 + J_w)} \end{bmatrix}$$

The output matrix C is an identity matrix of dimension 4×4 because all the states are supposed to be available on the output, while since the input is not acting on the output, the transition matrix D is set to zero.

5.3 First simulation: PID and LQR control

As discussed at the beginning of this chapter, this control technique is chosen to be applied to a non-linear system when the pendulum tilt angle θ has an absolute value smaller than 0.14rad ($\approx 8^\circ$).

The main parameters exploited for the simulation are:

- $m_p = 0.150\text{kg}$
- $m_w = 0.070\text{kg}$
- $d = 25\text{mm}$
- $R = 40\text{mm}$
- $r = 5\text{mm}$
- $\mu_s = 0.7$
- $\tau_{min} = 32.3\text{mNm}$ [minimum torque to apply on pendulum]
- $\tau_{max} = 60.4\text{mNm}$ [maximum torque before slipping]

$$Q = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

$$R = 1000$$

$$X_0 = [0 \ \pi/2 \ 0 \ 0]^T \quad [\text{system initial conditions}]$$

A huge value was given to R to prevent the input (the torque) from exceeding the slippage limit, while the tuning of Q matrix was done to prevent an unstable behavior of the system (explained later).

With these parameters, the gain vector K2 produced by LQR is:

$$K2 = [-0.0100 - 0.1967 - 0.0390 - 0.1011]$$

PID controller gains were hand-tuned until a satisfying behaviour was reached, hence:

- Proportional $P = -0.5$;

- Integral $I = -0.001$;
- Derivative $D = -0.05$;
- Filter coefficient $N = 100$;

As one can see from the overall simulink schematic in Fig. 5.1, two saturation blocks were added on the output of the control gains, the one for PID was set at $\pm 0.95\tau_{max}$ because, as shown in Fig. 5.3, big torques (in yellow) would have been requested in the first 0.1 seconds, leading to slippage, thus with a coefficient of 0.95 a safety margin was also added on the limit condition. The saturation on LQR gain was instead set to $\pm 0.7\tau_{max}$ because when the robot is already moving, the friction coefficient is no more static, but dynamic (estimated as 70% of the static one).

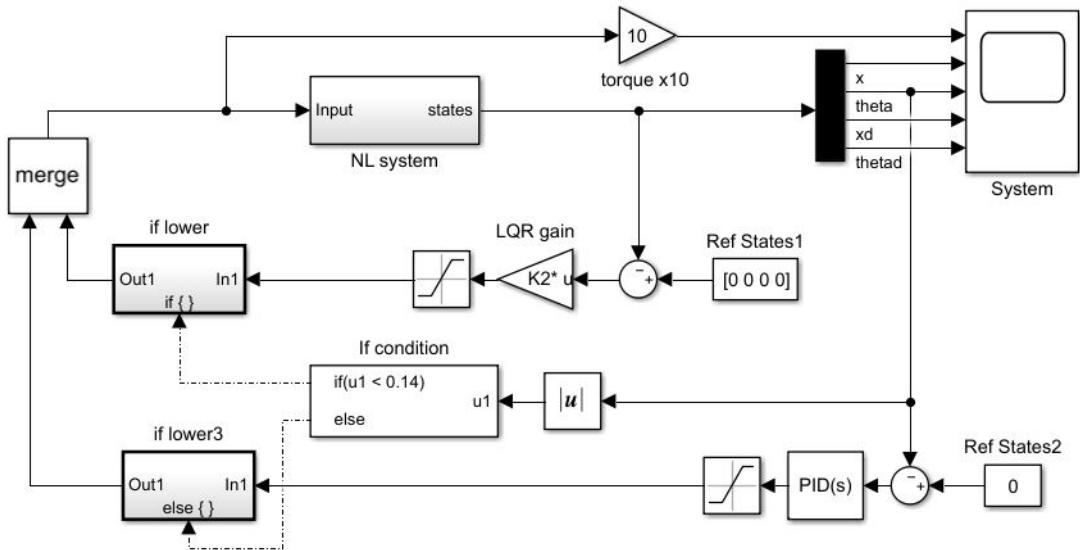


Fig. 5.1 Simulink schematic with PID and LQR control.

A 10 seconds simulation is shown in Fig. 5.2 below, demonstrating that with an appropriate tuning of the two controllers gains a stable self-righting manoeuvre is performed, in which e-puck3 reaches an upright position with a rising time (time required to a certain variable to cross the corresponding steady-state value) of $t_{r,\theta} \approx 0.70s$, travelling a distance $x_{max} \approx 1.16m$ on the floor with a maximum robot speed $\dot{x}_{max} \approx 0.6m/s$ and a maximum angular speed of the pendulum $\dot{\theta}_{max} \approx -12.4rad/s$.

The choice of firstly controlling the angle θ is rewarded with a small overshoot $\hat{s}_\theta \approx 8\%$ and good settling time (time required to a certain variable to stay within the $\pm\alpha\%$ of its steady-state value.) of $t_{s,\theta \pm 3\%} \approx 4.53s$.

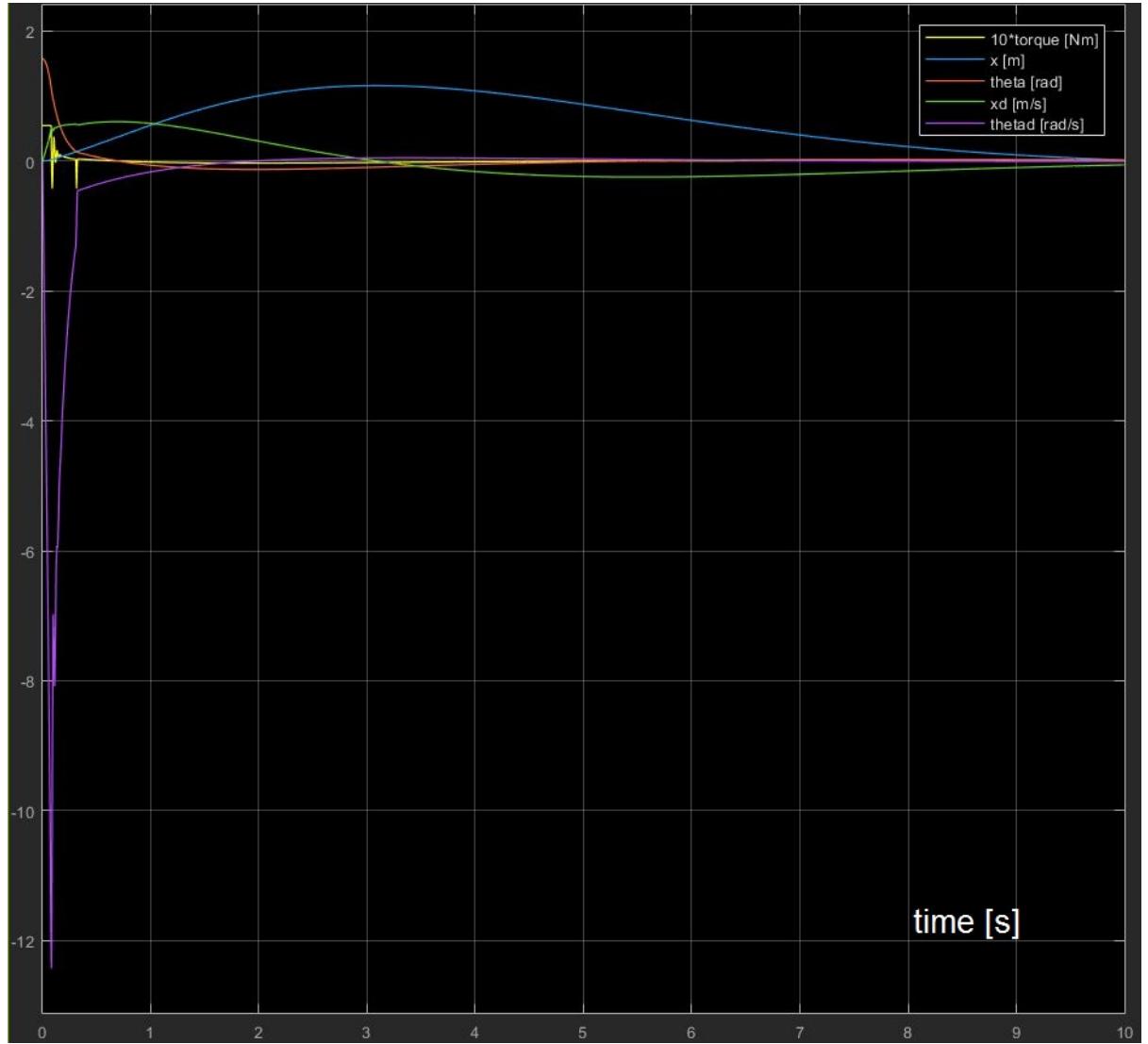


Fig. 5.2 Simulation with PID and LQR control.

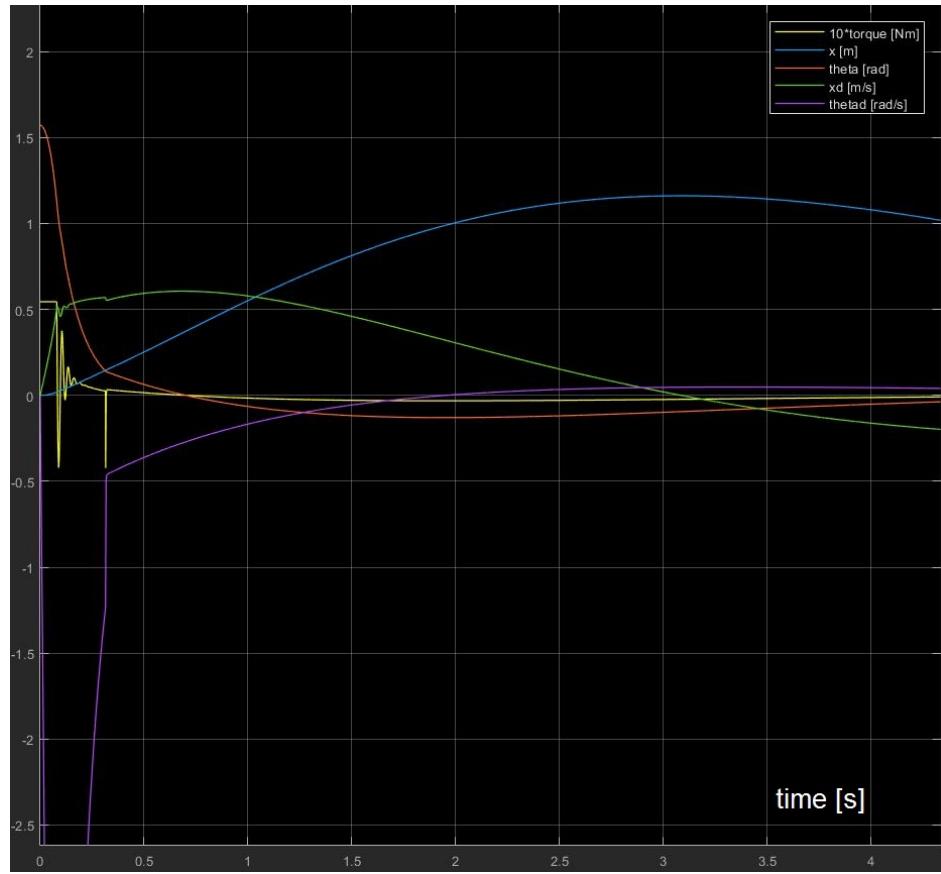


Fig. 5.3 Simulation of the first part of motion.

With this control approach, where two controllers are exploited, the tuning phase plays a very important role in terms of stability performances. An example of bad tuning is shown in Fig. 5.4 and 5.5, where the weights of Q matrix corresponding to θ and $\dot{\theta}$ were both put to 1.

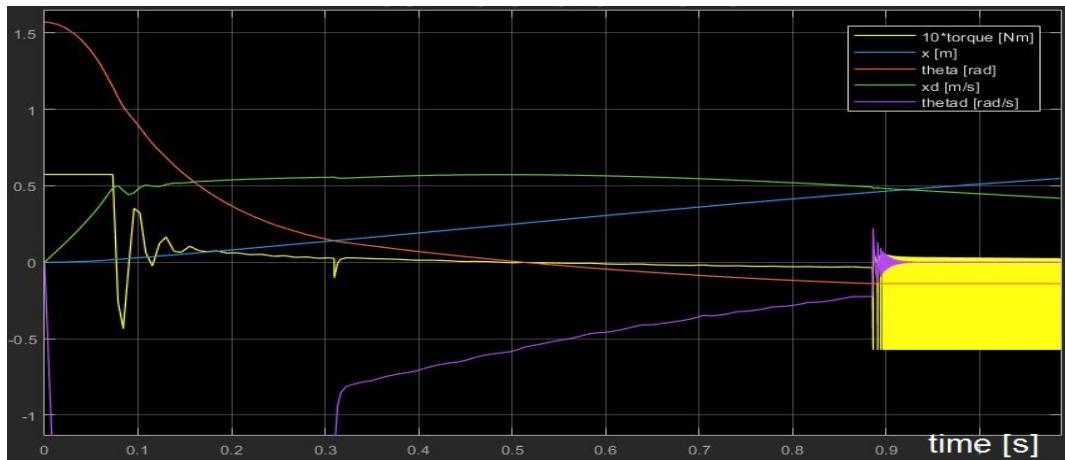


Fig. 5.4 Zoom on simulation with bad LQR tuning.

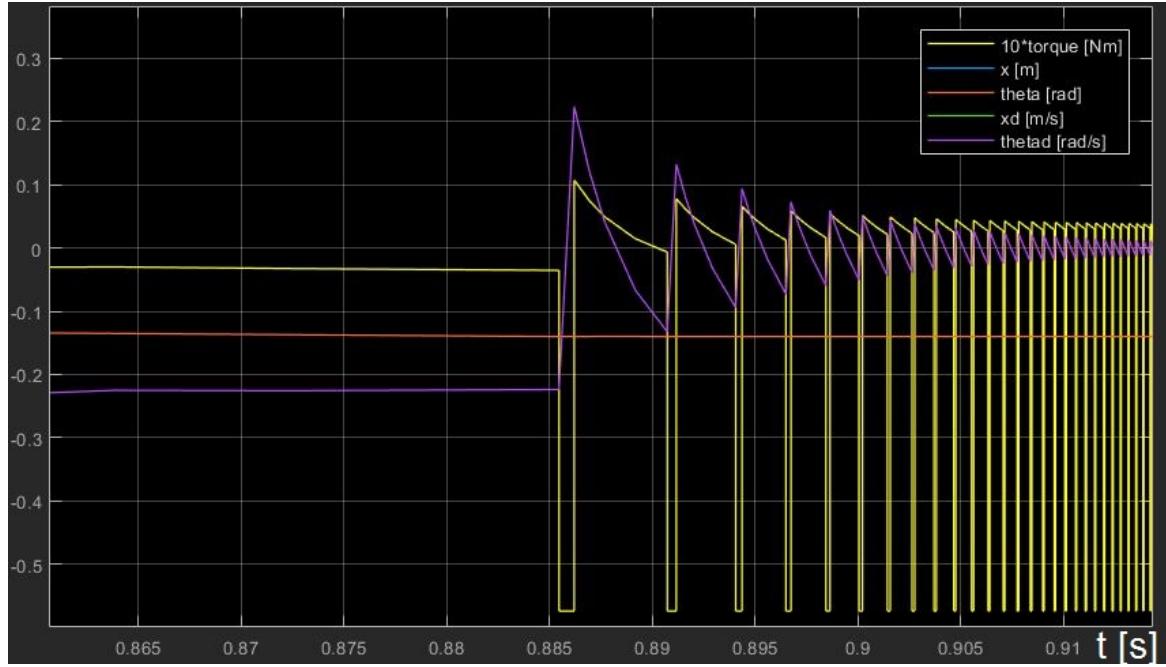


Fig. 5.5 Switching control with bad LQR tuning.

In this case the system gets stuck on the switching condition around the chosen negative threshold $\theta = -0.14\text{rad}$. The reason is simple: when the first (positive) threshold is reached, LQR is enabled and starts controlling all the 4 states, but since the control on θ is not strong enough, this state continues decreasing till reaching the negative threshold at which PID is re-enabled. At this point PID only governs θ so it tries to bring it back to zero, means increasing until $\theta = -0.14$, but here LQR is activated again and because of the improper weights in Q matrix, the control priority is not efficiently assigned, so the action on θ could be softened which means let it decrease below -0.14 , thus leading the system to continuously oscillates between these two phases.

5.4 Second simulation: LQR control

Even if the system is non-linear, applying only LQR control (Simulink diagram in Fig. 5.6) still provides good simulation results, as shown in Fig. 5.7 and 5.8.

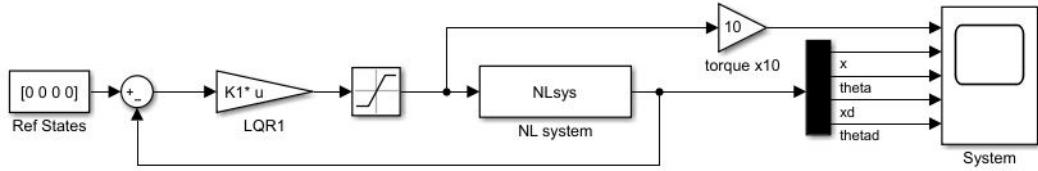


Fig. 5.6 Simulink schematic with only LQR control.

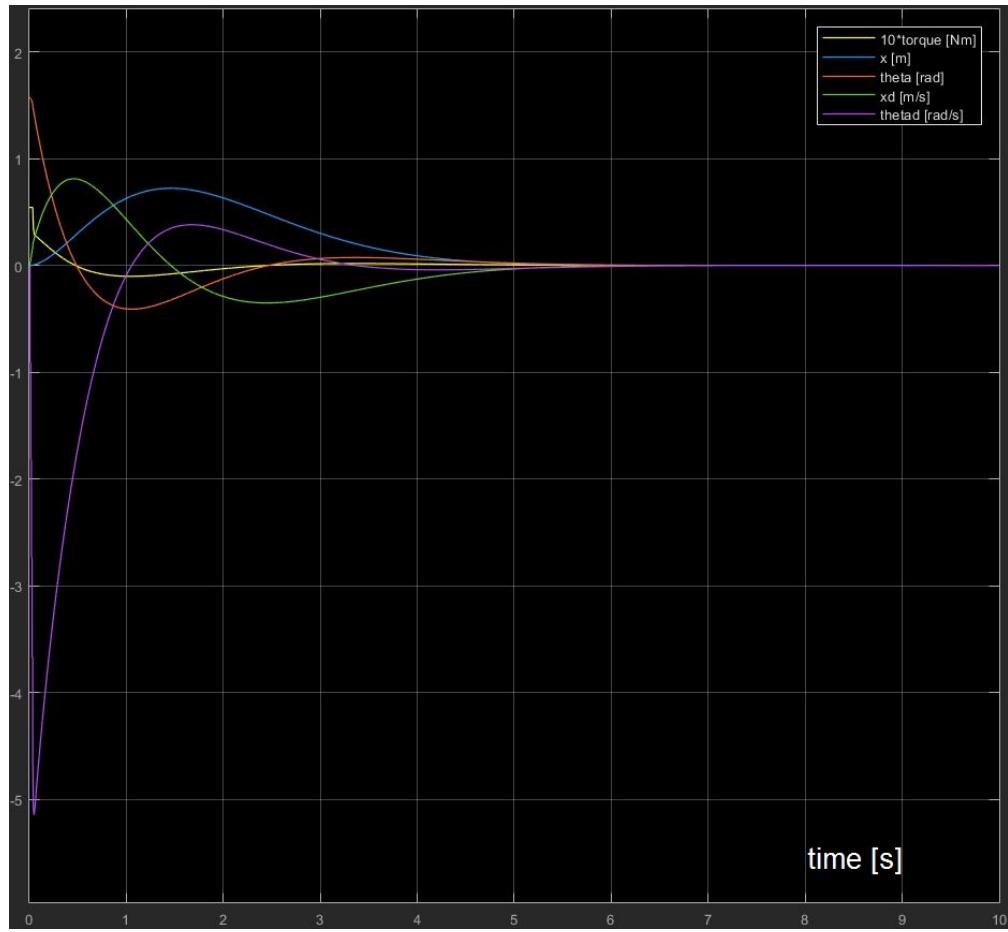


Fig. 5.7 Simulation with only LQR control.

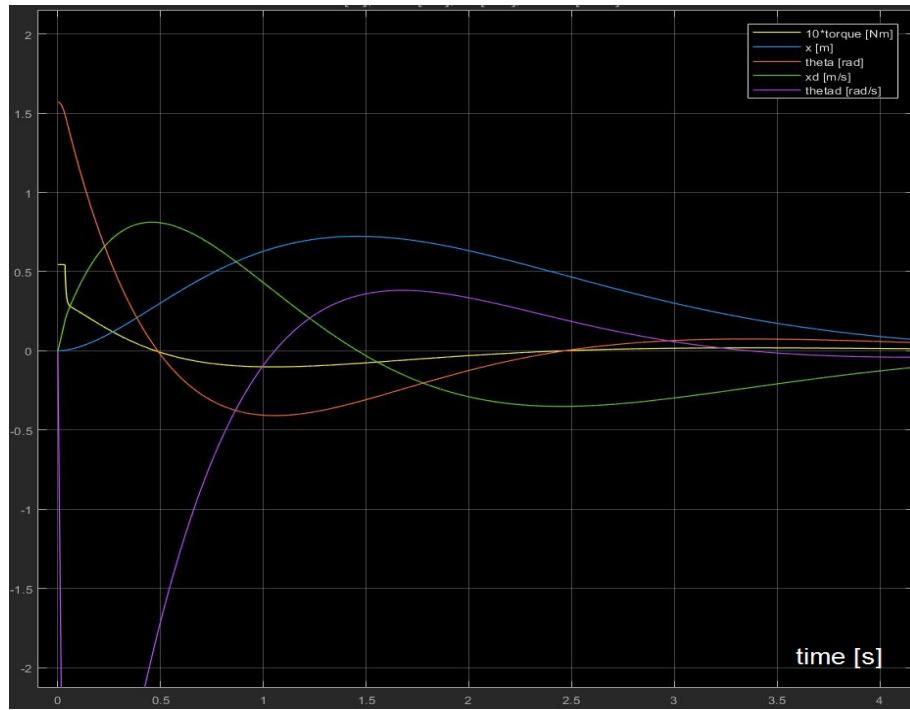


Fig. 5.8 Simulation of the first part of motion.

For this simulation, since the LQR control was enabled from the beginning of motion (not after PID), matrix Q has been modified from previous design and it corresponds to a simple identity matrix since no state has a critical effect on stability, thus:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$K1 = [-0.0316 - 0.1314 - 0.0616 - 0.0334]$$

This simulation shows better motion performances with respect to the previous one, in fact the travelled distance is $x_{max} \approx 0.72m$, with a maximum speed $\dot{x}_{max} \approx 0.81m/s$ and a maximum angular speed of the pendulum $\dot{\theta}_{max} \approx -5.1rad/s$ (less than half of the previous). In this case the overshoot on θ is $\hat{s}_\theta \approx 26\%$ while the rising time is $t_{r,\theta} = 0.48s$ and settling times is $t_{s,\theta \pm 3\%} = 4.75s$.

5.5 Third simulation: including BLDC motor

5.5.1 Introduction to simulation

As discussed at the end of paragraph 5.1, the main purpose of this project is to validate the motor choice for the self-righting manoeuvre, but non-linear equations developed in Chapter 3 describe a generic two-wheeled inverted pendulum without taking into account the physical limitations the exploited actuator. For this project, a brushless DC motor (BLDC), whose characteristics can be found on the catalog provided in Appendix-B, is exploited and hence a more accurate simulation has to be carried out. For the simulation, even if the motor to be validated is the EC21 flat from Maxon, as already mentioned in previous chapters, its data-sheet is not complete and for this reason EC20 flat is exploited because of similarity.

On Simulink, the BLDC motor block is deployed, receiving a voltage V as input and providing as output both an electromagnetic torque T_e and an angular speed ω_m . From theory of electro-mechanical conversion, the electromagnetic torque can be written as:

$$T_e = J\dot{\omega}_m + T_f + T_m \quad (5.1)$$

where J is the inertia of motor plus load, $T_f = t_{st} + \beta\omega_m$ is the torque loss due internal friction of the motor and can be represented as the sum of two effects (static+viscous), T_m is instead the mechanical load attached to the motor. As stated before, T_e and ω_m are provided by the motor block, T_f can be computed from motor data-sheet (product of no-load current and torque constant) and T_m is the torque requested by the load (the robot) which is provided by the controller gain. Since the inertia of the load is already taken into account in the non-linear equations of motion, the term J becomes only the motor inertia (on data-sheet).

Clarified this, the BLDC motor block on Simulink requires the exact t_{st} and β and not the total T_f , thus, knowing that in general the static component has bigger weight, I assumed $t_{st} = 90\%T_f$ and $\beta = \frac{10\%T_f}{\omega_m^*}$. Here, ω_m^* is the speed used to determine β , and corresponds to the speed of the motor when no load is applied out of the frictional one, that's to say the speed when the motor is absorbing no-load current; thus $\omega_m^* = 945\text{rad/s}$ is found by plotting the motor speed-torque characteristic and taking the speed corresponding to $T_f = k_t i_0 = 5.88 \frac{\text{mNm}}{\text{A}} \cdot 102\text{mA} = 5.99 \cdot 10^{-4}\text{Nm}$.

Problems rise up when simulating the motor in no-load conditions, applying the nominal voltage of 6V. Here, specifying t_{st} and β the motor reaches a no-load speed above 10000rmp when simulated with automatic fixed-step size and above 11000rpm with automatic variable-step size. This Simulink issue on BLDC motor model (whose internal parameters setting is

put in Appendix-A) affects any simulation results presented in the following paragraphs.

To include the motor physical limitations, the generation process of the states produced by the non-linear equations is modified. As opposed to the previous control approaches, now the idea is to compute the states θ and $\dot{\theta}$ from a kinematic relation among the rotor angle and the wheel angle, as shown in Fig.5.9.

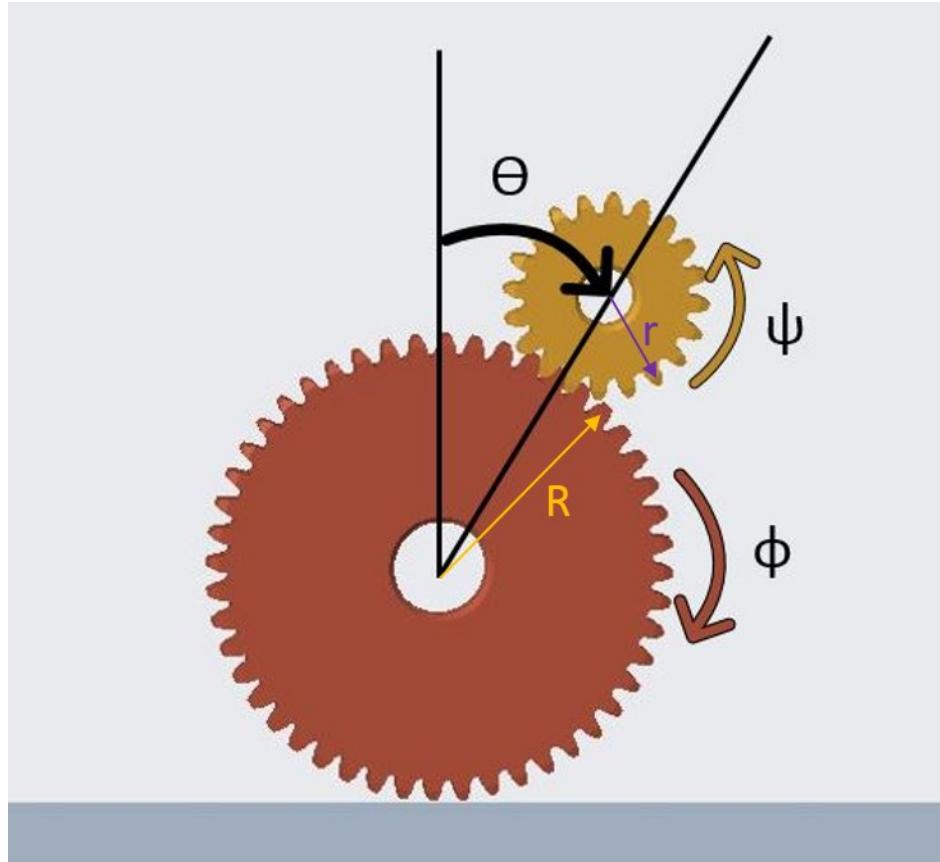


Fig. 5.9 Kinematic relations among angles.

As already discussed in previous chapters, the pendulum body is rotating around the center of the bigger gear, which is attached to the external wheel with radius R . Due to robot construction, the pendulum rotates at the same angular speed of the motor housing, thus θ can be also seen as the tilt angle of the smaller gear center with respect to the vertical position. The smaller gear (with radius r) turns counter-clockwise and the tangential speed $v_{pend} = \dot{\theta}(R + r)$ of its center, can be expressed as the sum of wheel and rotor speeds, thus:

$$v_{pend} = \dot{\phi}R + \dot{\Psi}r = \dot{\theta}(R + r) \quad (5.2)$$

Rearranging this equation, the angle speed $\dot{\theta}$ can be found as function of the system state \dot{x} and the motor speed ω_m , thus:

$$\dot{\theta} = \frac{1}{R+r}(\dot{x} + \omega_m r) \quad (5.3)$$

With this important relation, just by integration also the angle θ is defined so that together are exploited in the system of non-linear equations.

As before, given the current states $x, \theta, \dot{x}, \dot{\theta}$ and the input torque τ , the model provides the robot horizontal acceleration \ddot{x} and angular pendulum speed $\ddot{\theta}$, but with this new approach, the latter is not used to compute the next states since these will be computed through eq. 5.3 which will give a results depending on the next rotor speed, hence we can resume saying that the non-linear equations provide directly x, \dot{x} because of the integration on \ddot{x} and indirectly $\theta, \dot{\theta}$ because of eq.5.3.

The use of $\ddot{\theta}$ is anyway crucial for a good control of the system because, since BLDC motor is controlled by a voltage, the LQR control can only say what is the torque the motor has to generate, but not the corresponding rotor speed. The only way of making the system act in a controllable way, is to force the rotor speed to have a value which will make (at the next step) the pendulum angle θ to match the expected one by the integration of $\ddot{\theta}$ from second non-linear equation. All these concepts can be expressed through the following (simplified) blocks scheme, while the detailed one is put in Appendix-A.

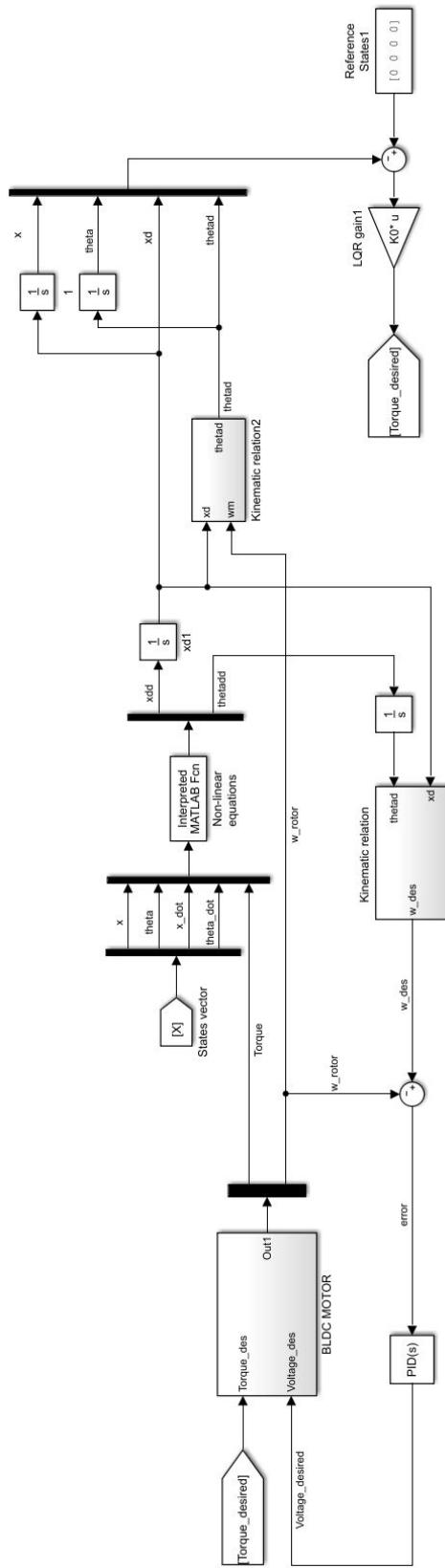


Fig. 5.10 Simulink schematic including BLDC voltage control.

5.5.2 Simulation including the motor

The simulation result is shown in the following Fig.5.11, where is visible that the BLDC motor is suitable for performing a self-righting motion.

Comparing this simulation with the other one which still exploits only LQR control, it's clear that performances are no more ideal since we added a real component (motor), thus we have: a travelled distance of $x_{max} \approx 0.74m$, with a maximum speed $\dot{x}_{max} \approx 0.87m/s$ and a maximum angular speed of the pendulum $\dot{\theta}_{max} \approx -6.5rad/s$. In this case the overshoot on θ is $\hat{s}_\theta \approx 27\%$ while the rising time is $t_{r,\theta} = 0.46s$ and settling times is $t_{s,\theta \pm 3\%} = 5s$.

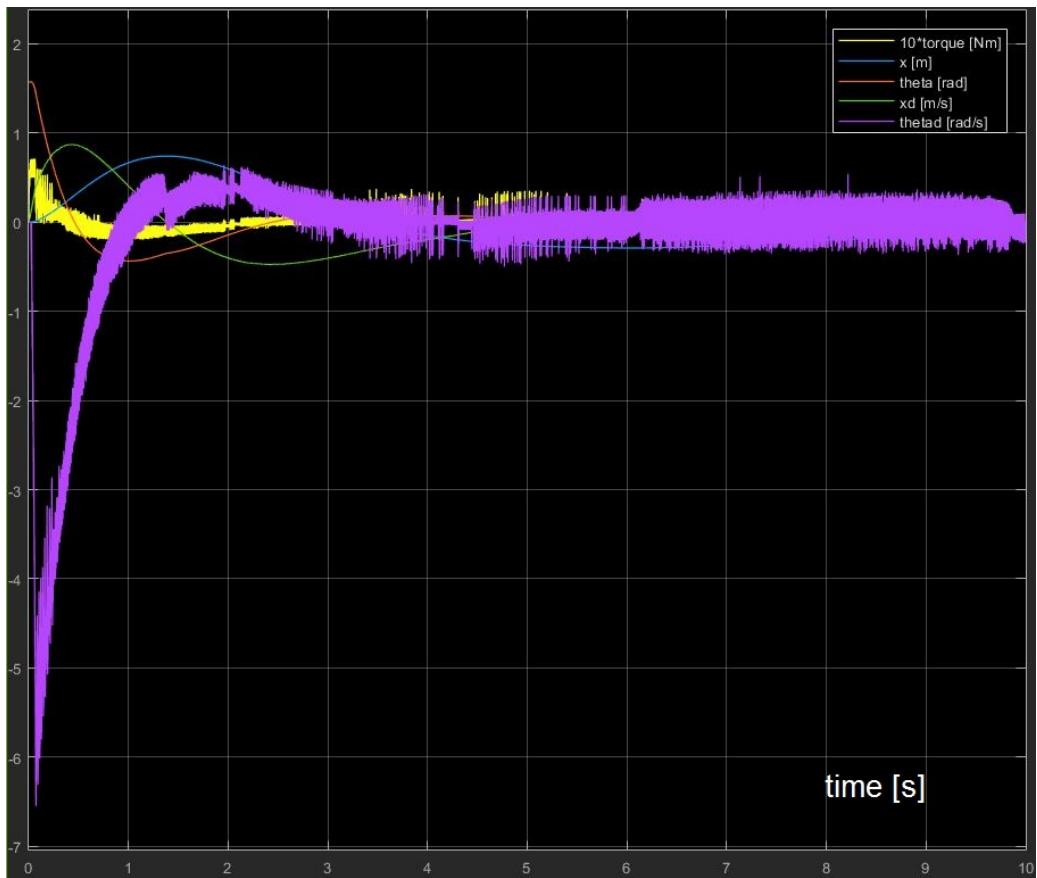


Fig. 5.11 Simulation result including BLDC motor.

The simulation is carried out with the same system parameters of previous simulation, with the same gain of the simulation exploiting only LQR control and with a PID on the speed error set as follows:

- Proportional $P = 14$; Integral $I = 0$; Derivative $D = 0.7$;
- Filter coefficient $N = 100$;

5.5.3 Motor operating conditions

Further analysis can be done on slipping condition, indeed in Fig.5.12 the motor speed-torque characteristic at nominal conditions is shown by a straight blue line, while the motor operating points during simulation are plotted in black. It's clear that all black points are below the nominal ones and in particular they lie in the "continuous operation range" where there are neither thermal issues nor wear problems for the motor.

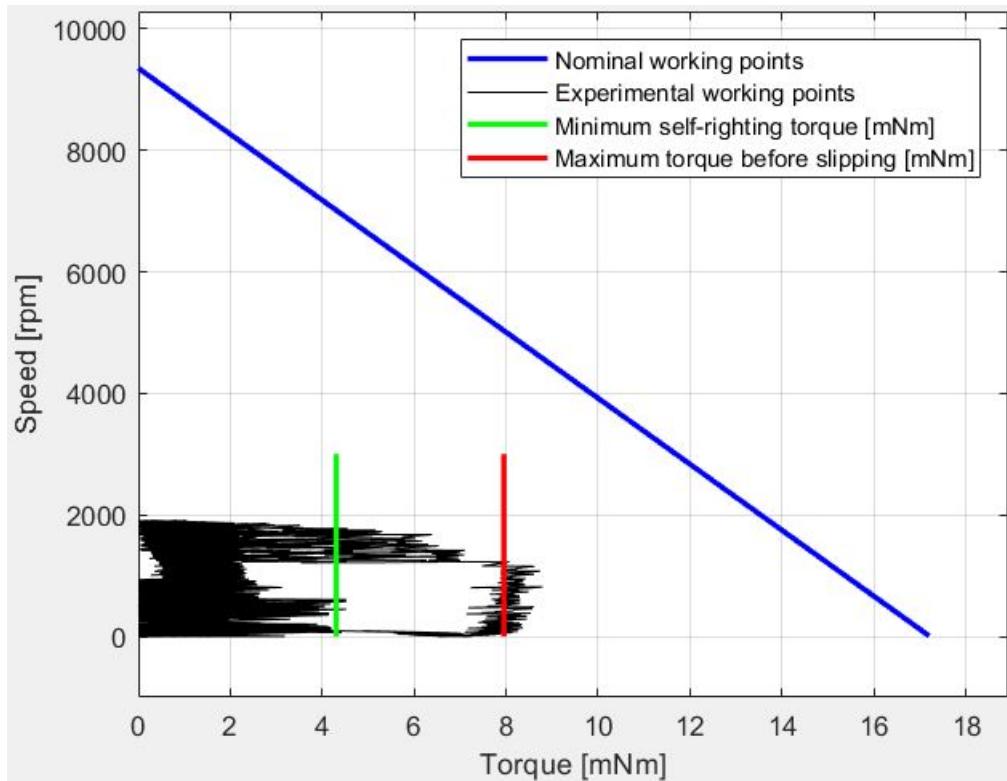


Fig. 5.12 Operating points of BLDC motor.

Even if there are few points exceeding the maximum admissible torque, the effective presence of these unwanted points is not guaranteed because it strongly depends on the electromagnetic torque which in simulation (Fig.5.11) varies with high frequency. Since the torque received by the wheel will probably be attenuated by mechanical components, a low-pass filter is added (see appendix A.7). Without a real system on which do some tests, parameters of this filter are chosen arbitrarily and for this reason points that overcome the red line may be not present if the attenuation is higher than what selected in simulation. Hence, by assuming a good torque attenuation, the real system will perform within the accepted range.

5.5.4 Motor efficiency

A final analysis was made on the motor efficiency η , expressed by the ratio between electrical and mechanical power, thus:

$$\eta = \frac{P_m}{P_e} = \frac{T_m \omega_m}{VI}$$

To avoid unfeasible spikes beyond an efficiency of 100%, the electrical power has been filtered with the same parameters exploited on the mechanical side discussed above.

Results are shown in Fig.5.13, where one can see how in general the efficiency is lower than what motor can offer. In fact, the efficiency at nominal condition (from motor data-sheet) is $\eta_{max} = 67\%$, while here the requested working points are less exorbitant than nominal ones, so it is comprehensible to have lower efficiencies.

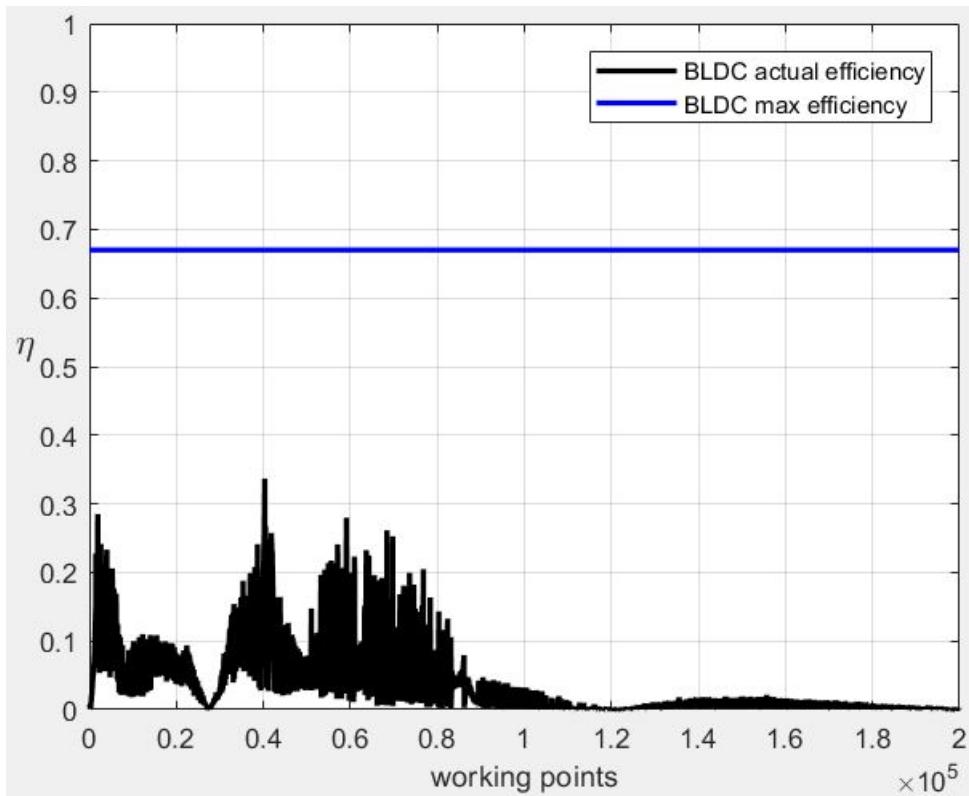


Fig. 5.13 BLDC motor efficiency.

Chapter 6

Conclusions and future improvements

A significant theoretical analysis was carried out on different engineering aspects regarding part of the configurations e-puck3 can assume. Since specific product requirements are not yet known, several designs of gearboxes and control approaches were studied and tested on the Two-Wheeled Inverted Pendulum (2WIP) configuration. As result, there's not a unique and best solution of design because everything depends on the desired robot performances, however this master project gives the possibility to understand lots of physical phenomena (and how they are related) which are often hidden inside a robot with small dimensions, as in the case of e-puck3. Several problems are linked indeed to manufacturing issues and high sensitivity to weights distribution on the robot itself; it's also important to keep in mind the economic value of the e-puck3, whom has been thought for educational purposes and thus it cannot rely on cutting-edge technology which would be too expensive.

By looking at the research work done for this project, I wish I'd have had less problems during modeling and simulations of the 2WIP system in order to develop a more challenging control approach (MPC) and a Simscape simulation to effectively show the robot animation. Regarding practical activities, lack of time didn't let me prototype the gearbox to perform practical tests with the motor but also to learn more about 3D-printing technology. An important aspect, for future studies, is for sure the development of a more reliable model of BLDC motor, without all the issues and uncertainties I faced. Finally, regarding the 2WIP modeling, friction wasn't taken into account both in Lagrangian and Newton approach, the two robot wheels have been assumed rotating at the same speed, thus the robot is not steering during motion and the gearbox was assumed to have only one stage (2 gears) with the gear attached to the wheel having a radius equal to the wheel itself, which is unlikely. A better implementation of this important aspects would allow people to have more flexibility in gearbox and control design phase.

References

- [1] <https://www.ucsdrobotics.org/edumip>
- [2] F. Mondada, "E-puck: a link among disciplines", École polytechnique fédérale de Lausanne.
- [3] F. Mondada, "Mobile Robots course: sensors (3)", École polytechnique fédérale de Lausanne.
- [4]
<https://supermagnetman.com/collections/neodymium/products/d1059d?variant=11410273731>
- [5]
[http://corsiadistanza.polito.it/corsi/pdf/04ASBP/ECM_1_calcolo_delle_ruote_dentate\(lewis\).pdf](http://corsiadistanza.polito.it/corsi/pdf/04ASBP/ECM_1_calcolo_delle_ruote_dentate(lewis).pdf)
- [6] <http://www.steel-bar.com/1-1191-c45e-steel/>
- [7] http://shop.sdp-si.com/catalog/product/?id=A_1B_1MY08016
- [8] <https://it.misumi-ec.com/vona2/detail/110300428950/>
- [9] <https://www.acplasticsinc.com/informationcenter/r/tensile-strength-of-polycarbonate>
- [10] <https://www.igus.eu/>
- [11] <https://www.sdp-si.com/resources/elements-of-metric-gear-technology/page5.php>
- [12]
https://www.khkgears.co.jp/KHK_GCSW/wormMain.do?method=trans&calc=hamen&lang=en
- [13] http://www.roymech.co.uk/Useful_Tables/Drive/Worm_Gears.html
- [14] R. Rojas, "A short history of omnidirectional wheels," See also URL
<http://robocup.mi.fu-berlin.de/buch/shortomni.pdf>, 2005.
- [15] <http://rezero.ethz.ch/>
- [16] Byun, K. S., & Song, J. B. (2003). Design and construction of continuous alternate wheels for an omnidirectional mobile robot. *Journal of Robotic Systems*, 20(9), 569-579.
- [17] H. Stoekli, "E-Puck II Self-balancing implementation" master's thesis, École polytechnique fédérale de Lausanne, pp.26, 2017.
- [18] Allgöwer, F & Findeisen, Rolf & Nagy, Z.K.. (2004). Nonlinear model predictive control: From theory to application. *J. Chin. Inst. Chem. Engrs.* 35. 299-315.
- [19] B. D. O. Anderson, J. B. Moore, *Optimal Control - Linear Quadratic Methods*, Prentice Hall, 1989.

Appendix A

Simulink Model blocks

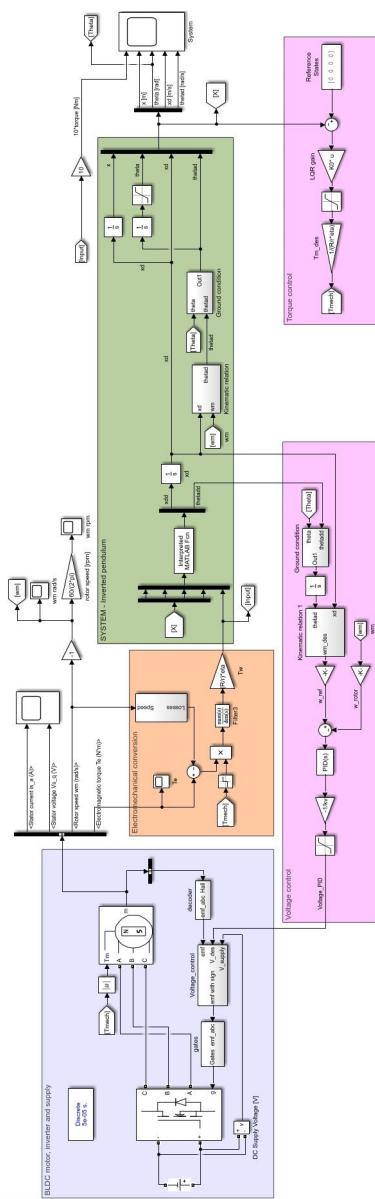


Fig. A.1 Detailed simulink schematic.

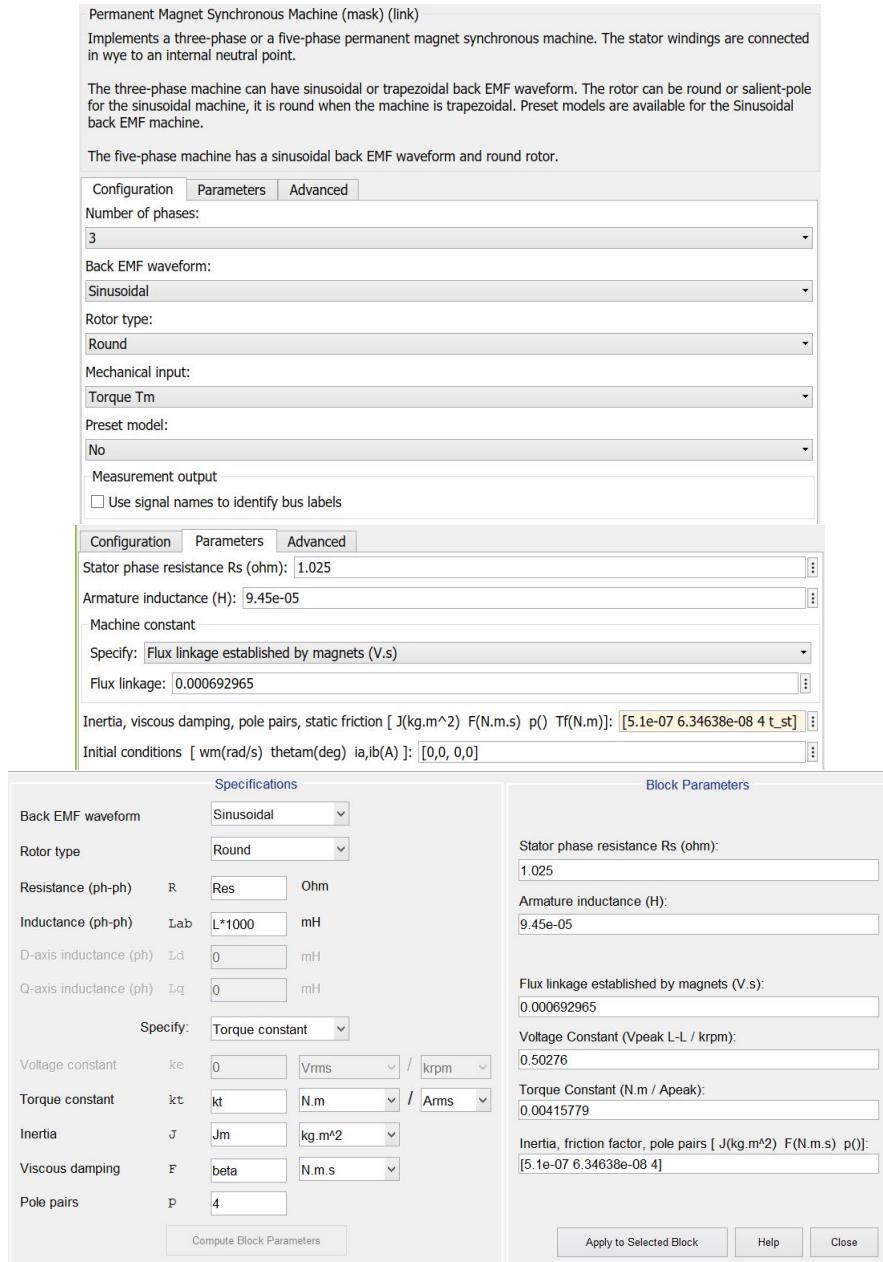


Fig. A.2 BLDC block settings.

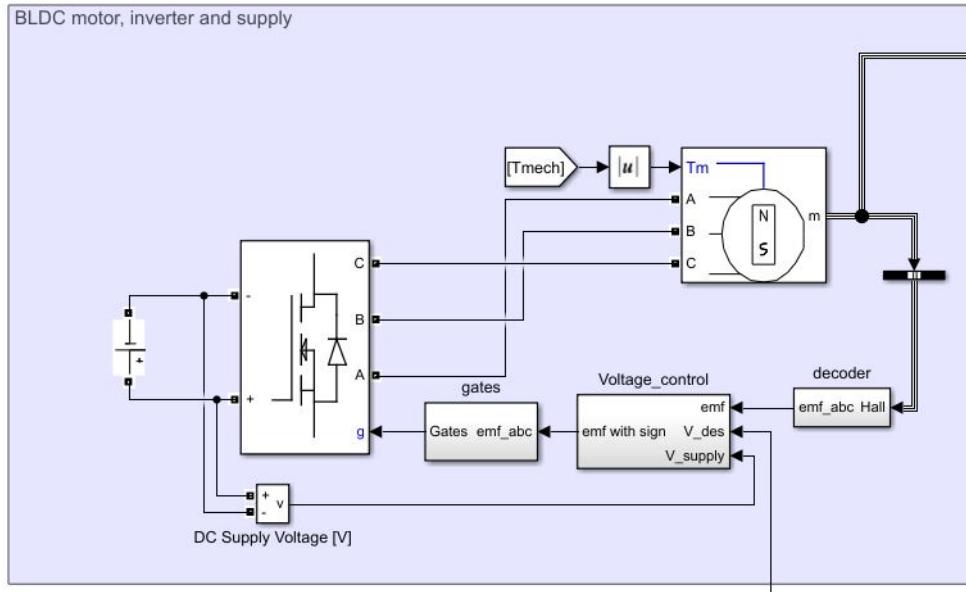


Fig. A.3 BLDC motor block connections.

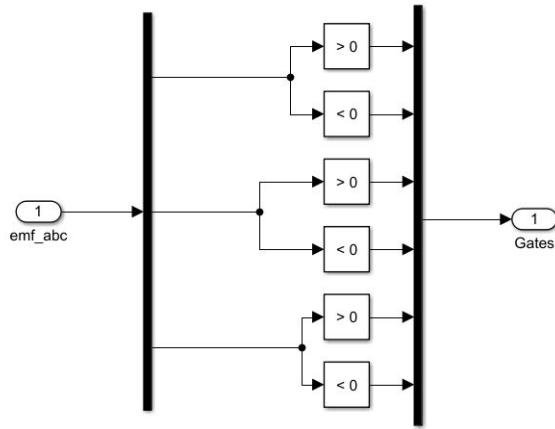


Fig. A.4 BLDC motor - Gates.

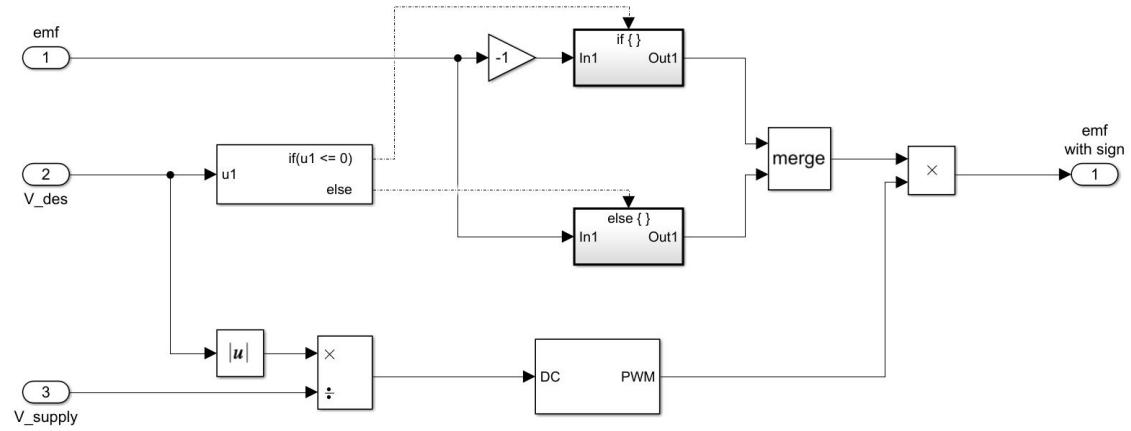


Fig. A.5 BLDC motor - PWM on Voltage.

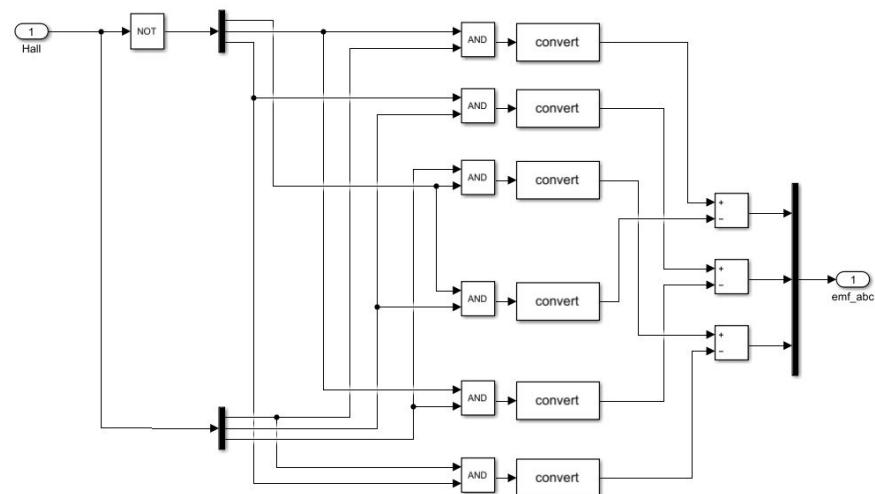


Fig. A.6 BLDC motor - Decoder.

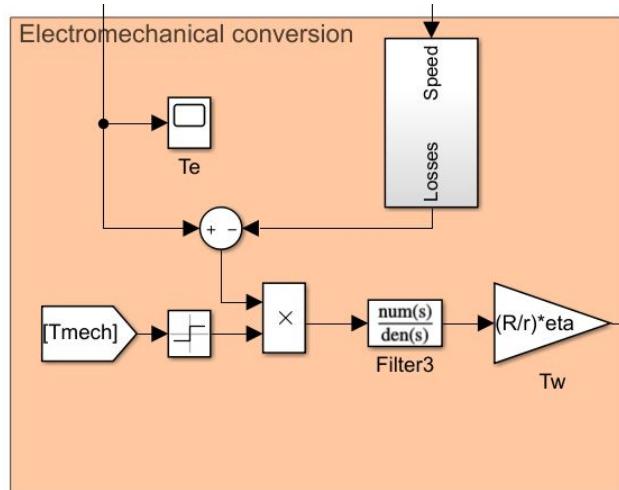


Fig. A.7 Electromechanical conversion.

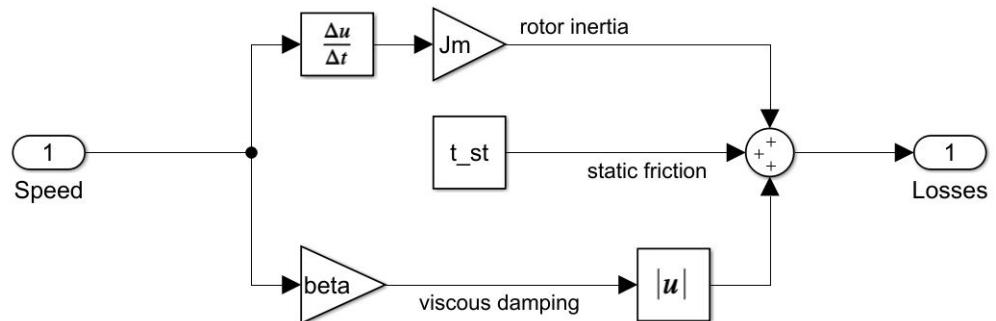


Fig. A.8 Torque losses.

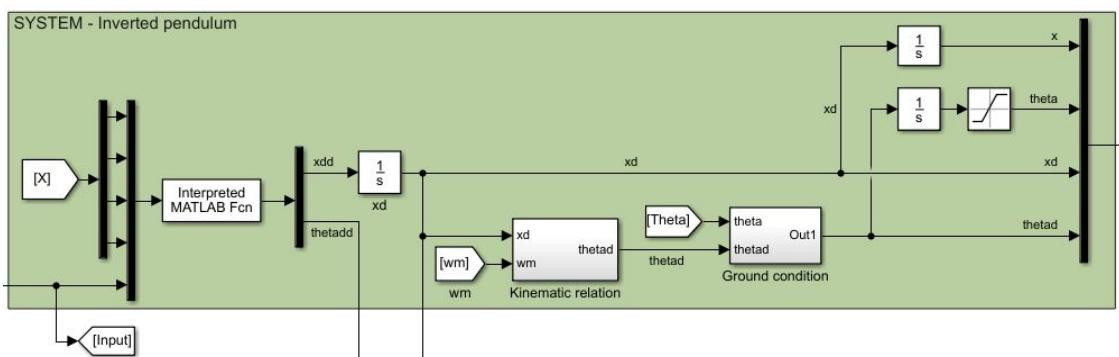


Fig. A.9 System states evaluation.

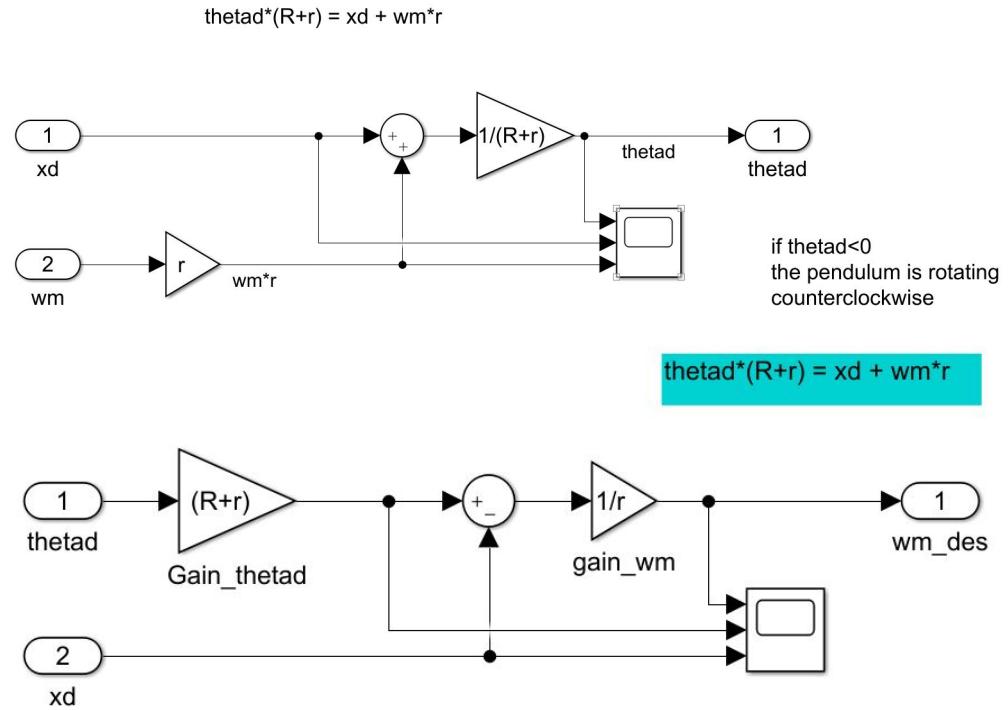


Fig. A.10 Kinematics relations.

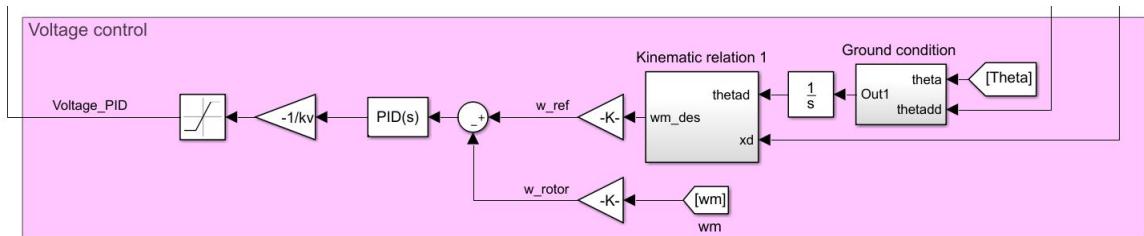
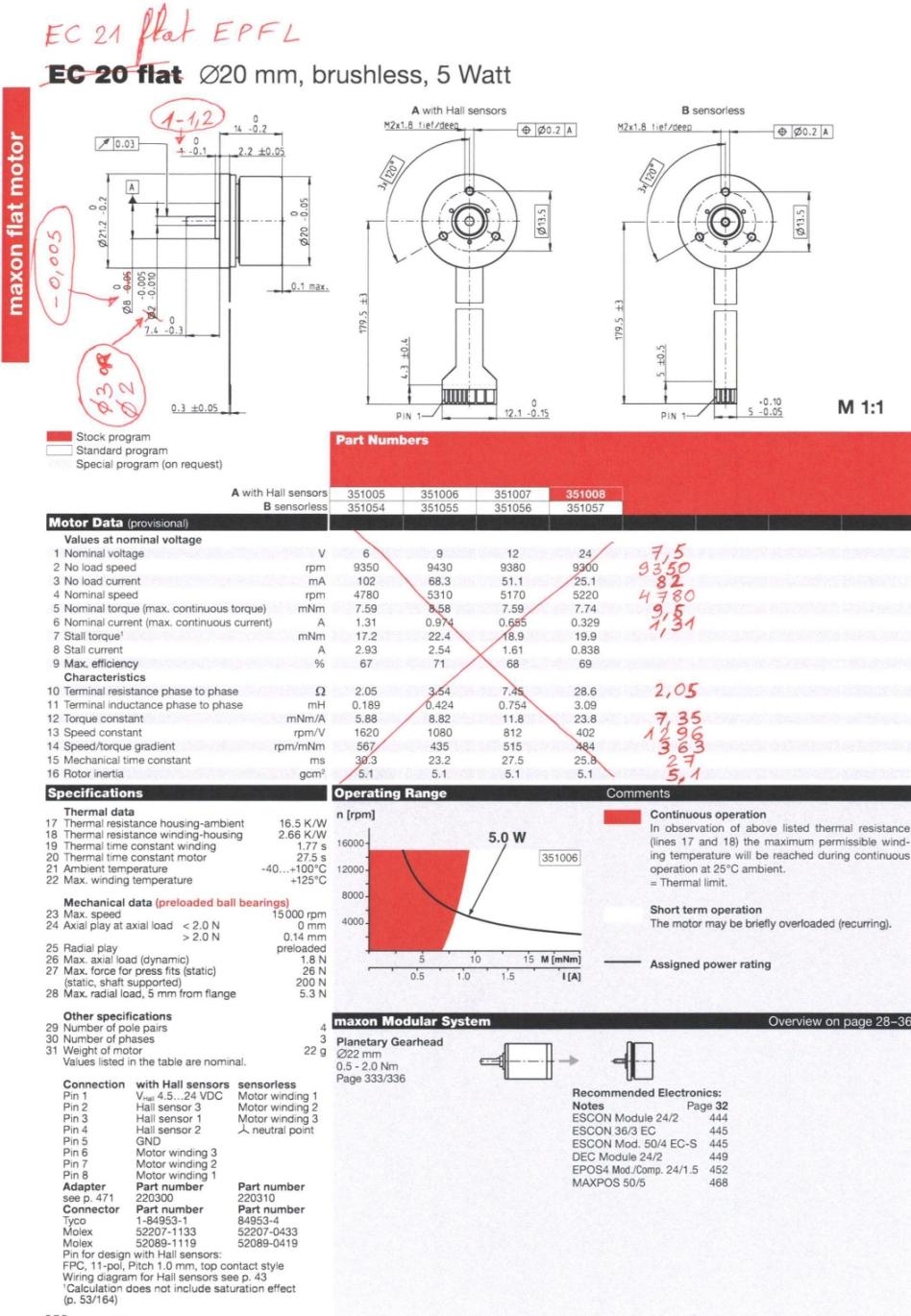


Fig. A.11 Voltage control.

Appendix B

Motor data-sheet



Modified by olch, 8.01.2019

Fig. B.1 EC20 Data-sheet