Universidade do Minho
Escola de Engenharia

Fernando João Ferreira Gonçalves

Design, Development and Control of a New Anthropomorphic Mobile Manipulator to Help and Interact with Humans in Daily Tasks

janeiro de 2026

Design, Development and Control of a New Anthropomorphic Mobile Manipulator to Help and Interact with Humans in Daily Tasks

Fernando João Gonçalves

UMinho | 2026


REPÚBLICA PORTUGUESA
EDUCAÇÃO, CIÊNCIA E INOVAÇÃO
fct
Fundação para a Ciência e a Tecnologia
NORTE2020
PORTUGAL 2030
PORTUGAL 2020
PESSOAS 2030
UNIÃO EUROPEIA
Fundo Social Europeu

Universidade do Minho
Escola de Engenharia

Fernando João Ferreira Gonçalves

Design, Development and Control of a New Anthropomorphic Mobile Manipulator to Help and Interact with Humans in Daily Tasks

Tese de Doutoramento
Programa Doutoral em Engenharia Mecânica

Trabalho realizado sob a orientação de
Professor Doutor João Paulo Flores Fernandes
Professor Doutor Agostinho Gil Teixeira Lopes

janeiro de 2026

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho



Atribuição-NãoComercial
CC BY-NC

https://creativecommons.org/licenses/by-nc/4.0/

ACKNOWLEDGMENTS

A PhD is a journey of both personal and professional growth. I've reached the finish line thanks to my own work and merit; however, I could not have made it alone. This section is dedicated to all those who helped me along the way.

First, I want to express my huge gratitude to my supervisor: Professor Paulo Flores. A true mentor and guide, his support on both a personal and professional level is what allowed this document to be finished. Thank you.

I would like to thank my co-supervisor Professor Gil Lopes. For always believing in me and my work, for sharing his enthusiasm with me, and for being there whenever I needed.

I am grateful to *Fundação para a Ciência e a Tecnologia* (FCT) for funding this work through the PhD grant SFRH/BD/145993/2019.

To my great friend and fellow researcher Joana Coelho. It was an honor to work alongside such an amazing, inspiring, and hardworking researcher.

I am grateful to the DEM (Department of Mechanical Engineering) of the University of Minho, and all my colleagues and professors. I want to express a deep thank you to Professor Filipe Marques, my colleague Mariana Silva, and to the other Filipe Marques.

To the LAR (Laboratory of Automation and Robotics) which housed the CHARMIE project. LAR's projects were what first enticed me into a PhD. I extend this gratitude to Professor Fernando Ribeiro, Tiago Ribeiro, and Inês Garcia, all great co-workers.

To my university colleagues Luciano Fernandes, Teresa Pedregal, César Rocha, Diogo Lopes, Elias Tinoco, Alexandre Viana, and João Paulo Magalhães who did the master's in mechanical engineering alongside me. It is great to be able to count on your constant support.

To Doctor Pedro Lourenço, an inspiring friend who guided me in the biggest hardships of this journey and taught me how to find my strength in the battle against the PhD.

To the people who inspire me, thank you for providing hours of rich entertainment, and leading by example, namely the writers Brandon Sanderson, Robert Jordan, and Tappei Nagatsuki, as well as the VTubers Almantas, Mori Calliope, Houshou Marine, and especially my great friend Janken Pomme. I also want to use this space to thank some of my online friends in the Age of Empires community, such as Dizzy, RBase, TeaTime, and Zelda.

I am blessed with great friends. Francisco Coelho, Rodrigo Mira, Orlando Carvalho, Thomas Lima, Afonso Castro, and Renato Afonso, I want to thank you for the constant laughs we shared, and for always cheering me up. I also want to thank my longtime friends Joel Morais, Luís Ventuzelos, Daniel Malhadas and Bruno Silva. No matter how much time passes, it strengthens me to always have you by my side.

Finally, the most important acknowledgment goes to my family. To my niece Sara, you were the happiest event during my PhD journey. It is wonderful to see you grow; I hope you keep your big smile. To my brother, João Gonçalves, a brilliant researcher, who taught me many things about life, who constantly supports me and helped me in this work, and with whom I share many of my greatest passions. To my sister-in-law, Ana Isabel, for her hospitality and for always providing great company for a talk. The grandest acknowledgment goes to my parents, for always supporting me and believing in me, for the excellent job they did in raising me, and for giving me everything I needed to follow my goals and ambitions.

Thanks to all of you, both those I mentioned and many others that I did not include.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

RESUMO

Projeto, desenvolvimento e controlo de um novo manipulador movel antropomórfico para auxiliar e interagir com humanos em tarefas quotidianas.

Os robôs são cada vez mais comuns no nosso quotidiano. Eles alcançaram posições chave nos sectores dos serviços e da indústria, mas a sua aplicação em contextos domésticos é recente, sendo o alvo de diversos estudos em curso. Com isto em conta, o presente trabalho visa criar um robô manipulador móvel antropomórfico para desempenhar tarefas domésticas de modo a assistir pessoas com mobilidade reduzida. Este projeto iniciou-se analisando o espaço de trabalho do robô e estudando a literatura de modo a estabelecer as especificações do projeto. Estas especificações foram a base para definir a morfologia do robô e estabelecer o seu desenho conceptual. Este conceito evoluiu para um projeto fazível e funcional através do uso de ferramentas CAD, nas quais se criaram modelos detalhados que consideram a manufatura e montagem do sistema. Antes de construir o protótipo, foi necessário validar e otimizar a escolha de atuadores e o projeto de mecanismo. Para este fim, programou-se um simulador computacional que soluciona a cinemática direta e a dinâmica inversa de todos os corpos do robô, modelando assim a dinâmica completa do sistema. O simulador foi empregue para determinar as forças e binários aplicados nas juntas, assim como para o estudo da otimização paramétrica de mecanismos de balanceamento estático para reduzir o consumo energético dos atuadores. O protótipo foi contruído e funcionou como intendido, validando a abordagem teórica proposta neste estudo. O robô operará em habitações domésticas, as quais são pouco estruturadas. Assim sendo, adaptabilidade no controlo das trajetórias é necessária para desempenhar tarefas de forma adequada. Isto é alcançado controlando os atuadores do robô com estratégias de aprendizagem por reforço. O simulador desenvolvido foi usado para treinar uma rede neuronal de otimização de políticas proximais para manipular objetos sem intervenção humana. O sistema de controlo foi capaz de autonomamente gerar trajetórias nas quais o robô transportou uma lata de metal de um balcão para outro. Robótica é um ramo de conhecimento marcado pela sua interdisciplinaridade: este trabalho criou um sistema robótico, contruiu um protótipo físico dele, importou-o para um novo simulador virtual, e implementou aprendizagem por reforço para gerar as trajetórias do sistema.

Palavras-chave: Aprendizagem por Reforço, Dinâmica de Sistemas Multicorpo, Projeto de Mecanismos, Robô Doméstico

ABSTRACT

Design, development and control of a new anthropomorphic mobile manipulator to help and interact with humans in daily tasks

Robots are becoming commonplace in our everyday lives. They have attained key positions in the services and industrial sectors, but their application in domestic environments is recent and still the target of ongoing research. Faced with this context, the present work created an anthropomorphic mobile manipulator robot to assist people with reduced mobility by performing domestic chores. The project was started by analyzing the intended workspace and reviewing the literature to establish the main required specifications. These specifications were the basis for defining the morphology of the robot and its conceptual design. This concept evolved into a feasible and functional system by employing CAD tools to create detailed models that considered both the manufacturing and assembly aspects of the project. Before constructing a prototype, the choice of actuators and the design of the mechanisms had to be validated and optimized. To this end, a computational simulator was programmed that solves the forward kinematics and inverse dynamics of all the bodies of the robot, completely modeling the dynamics of the full system. The simulator was employed to determine joint forces and torques and to perform the parametric optimization of energy-saving static balancing mechanisms. The prototype was then constructed, and it successfully functioned as intended, thus validating the theoretical approach proposed in this study. The robot's main work environment will be domestic households that are highly unstructured. Therefore, to adequately perform daily tasks, adaptability is required in trajectory control. This can be attained by utilizing reinforcement learning to control the robot's actuators. The developed simulator was used to train a proximal policy optimization neural network to manipulate objects without the need for any human input. The control system was able to generate trajectories where the robot autonomously transported a metal can from one counter to another. Robotics is a field marked by its interdisciplinarity: this work created a new robotic system, constructed a physical prototype of it, imported it into a newly created virtual simulator, and implemented a reinforcement learning control system to generate its trajectories.

Keywords: Domestic Robots, Mechanism Design, Multibody System Dynamics, Reinforcement Learning

TABLE OF CONTENTS

ACKNOWLEDGMENTS III

RESUMO V

ABSTRACT VI

TABLE OF CONTENTS VII

LIST OF FIGURES X

LIST OF TABLES XIII

LIST OF ABBREVIATIONS XVI

LIST OF SYMBOLS XVII

1. INTRODUCTION 1

1.1. Motivation 2

1.2. Scope and Objectives 4

1.3. State of the Art 5

1.4. Structure of the Thesis 13

1.5. Contributions of this Work 14

2. LITERATURE REVIEW 16

2.1. Mobile Manipulator Robot Design 16

2.2. Mechanism Synthesis 21

2.3. Multibody Dynamics in Robotics 25

2.4. Neural Networks 32

2.5. Summary and Discussion 34

3. DESIGN AND CONSTRUCTION OF THE PROPOSED CONCEPT 35

3.1. Robot Specifications 35

3.2. Design of the CHARMIE Robot 37

3.3. Mechanism Optimization 46

3.4. Prototype Construction 57

3.5. Summary and Discussion 61

4. MULTIBODY MODEL OF CHARMIE 62

4.1. Description of the Multibody System 63

4.2. Forward Kinematics Analysis 71

4.3. Inverse Dynamics Analysis 77

4.4. Robot-Environment Interaction 84

4.5. Summary and Discussion 92

5. REINFORCEMENT LEARNING TRAJECTORY CONTROL 93

5.1. Proximal Policy Optimization Neural Network 94

5.2. Environment Definition 96

5.3. Agent Training 101

5.4. Result Analysis and Performance Improvement 107

5.5. Summary and Discussion 112

6. CONCLUSIONS AND FUTURE DEVELOPMENTS 113

6.1. Concluding Remarks 113

6.2. Future Work 117

REFERENCES 118

LIST OF PUBLICATIONS 132

APPENDIX I - EQUATION FACTORS FOR THE BEVEL GEAR ANALYSIS 134

APPENDIX II - LIST OF PARTS OF THE ROBOT 141

APPENDIX III - TECHNICAL DRAWINGS 146

APPENDIX IV - ASSEMBLY GUIDE 242

APPENDIX V - GEOMETRIC PROPERTIES FOR COMPUTATIONAL MODEL 305

ANNEX I - DATABASE OF TENSION SPRINGS 315

LIST OF FIGURES

Figure 1.1 – Percentage of population aged 65 or over around the world for 2022, 2030, and 2050 according to the medium scenario predicted by (United Nations Department of Economic and Social Affairs - Population Division, 2022) 2

Figure 1.2 – Trends in the development of mobile manipulator robots considering design and applications. 11

Figure 2.1 – Comparison of the kinematics for a) 9-DOF, b) 7-DOF, and c) 6-DOF robotic manipulator arm. 20

Figure 2.2 – Kinematics of the a) Prismatic Joint, b) MacPherson strut, and c) Double Wishbone suspensions. 22

Figure 2.3 – a) Anglepoise lamp and b) Spring-And-Lever static balancing mechanisms 24

Figure 2.4 – Architecture of a Fully Connected Deep Feed Forward Neural Network 32

Figure 3.1 – CAD model of CHARMIE's preliminary design 38

Figure 3.2 – CAD model of the wheel-motor-bearing assembly 39

Figure 3.3 – CAD model of the suspension system assembly. One of the wheel-motor-bearing groupings is unassembled for better visibility. 40

Figure 3.4 – CAD model of the robot's hip with the squatting motion mechanism 42

Figure 3.5 – Schematic representation of the robot's squatting motion static balancing mechanism 43

Figure 3.6 – CAD model of the robot's trunk flexion motion mechanism 43

Figure 3.7 – CAD model of the subsystems of the manipulator arm with the output degrees of freedom highlighted: a) shoulder joint; b) mechanical claw; c) arm z-rotation; and d) arm x-rotation. 45

Figure 3.8 – CAD model of the robot's manipulator arms 45

Figure 3.9 – a) CAD model of the robot's head and b) detail highlighting the neck's DOF 46

Figure 3.10 – Parameters for the optimization of the static balancing mechanisms of the CHARMIE robot for the a) squatting mechanism and b) trunk flexion mechanism 47

Figure 3.11 – Prototype of CHARMIE's suspension and locomotion systems 58

Figure 3.12 – Prototype of CHARMIE's squatting and trunk flexion motion mechanisms: a) front view, b) side view, and c) three-quarter-view. 58

Figure 3.13 – Comparison of the power consumed by the actuators of the trunk flexion and squatting mechanism with or without static balancing springs 59

Figure 3.14 – Current functional prototype of the CHARMIE robot a) front view, b) rear view. 60

Figure 4.1 – Kinematic diagram showcasing the bodies and degrees of freedom of the CHARMIE robot 63

Figure 4.2 – Kinematic tree representing CHARMIE's multibody system 64

Figure 4.3 – Schematic representation of the notation used for the description of the body geometries. 65

Figure 4.4 – Identification of the key points required to describe the geometry of the robot's torso in a) the CAD model, and b) the computational simulation (see Table 4.1 for the associated point coordinates)..... 66

Figure 4.5 – Geometric parameters for the analysis of the indirectly actuated joints of CHARMIE, namely (a) the joint preceding body 2 and (b) the joint preceding body 3. 72

Figure 4.6 – Geometric parameters for the analysis of the indirectly actuated joints of CHARMIE, namely (a) the joint preceding body 2 and (b) the joint preceding body 3. 76

Figure 4.7 – Kinematic model of CHARMIE in the multibody simulation environment 77

Figure 4.8 – Free body diagram of CHARMIE's torso and hip. Green hues represent forces determined by the recursive algorithm, while orange hues represent calculated forces in the analysis of this subsection. 80

Figure 4.9 – Free body diagram of the static balancing mechanism bar for the squatting motion..... 82

Figure 4.10 – Forces and torques exerted by the floor on CHARMIE's base on the simulation of a 5-second determined using the method described in this thesis, as well as from a commercial software for comparison. 84

Figure 4.11 – Schematic representation of CHARMIE's independent suspension system. 85

Figure 4.12 – Response of CHARMIE's suspension system to the motion of the upper body. The suspension's spring constant was lower for visual effect. 87

Figure 4.13 – Forward (a) and diagonal (b) motion of CHARMIE's base simulating the dynamics of the locomotion system. The arrow represents the ideal intended motion, while the white circles represent the starting and finishing positions of the robot. 88

Figure 4.14 – (a) 3D CAD model and (b) virtual simulation of CHARMIE in its work environment..... 89

Figure 4.15 – Progression of the level of detail of CHARMIE's model for collision detection 90

Figure 4.16 – Increase in CHARMIE's level of detail in the computer simulation as it approaches an obstacle. Bodies colored in red are colliding with the obstacle. 91

Figure 4.17 – Finite state machine describing the state of the metal can in the simulation environment 91

Figure 5.1 – Actor-critic method, adapted from (Arulkumaran et al., 2017) 94

Figure 5.2 – Work environment for the robot for the neural network metal can manipulation task..... 96

Figure 5.3 – Average rewards for different neural network depths for a constant width of 128 neurons 103

Figure 5.4 – Average rewards for different neural network widths for a constant depth of six layers .. 103
Figure 5.5 – Difference in training results for different learning rates 104
Figure 5.6 – Rewards over time for the pre-training of the neural network 106
Figure 5.7 – Comparison of results of the training for the application scenario with and without pretraining. 107
Figure 5.8 – Training results for the pre-training of the neural network in a loose tolerance scenario. 108
Figure 5.9 – Training results for the neural network in scenarios with sequentially tighter tolerances. 109
Figure 5.10 – Maximum rewards for the neural network in scenarios with sequentially tighter tolerances. 109
Figure 5.11 – Evolution of the right claw position and orientation over time when controlled by the NN 110
Figure 5.12 – Motion over time of the robot controlled by the Neural Network in the simulation environment 111

LIST OF TABLES

Table 1.1 – Examples of robotic mobile manipulator systems discussed in the literature grouped by application and design. 10

Table 2.1 – Comparison of the different independent suspension types considering the intended application. 23

Table 3.1 – Requirements established for the design of the CHARMIE robot. 37

Table 3.2 – Properties of the compression springs used for CHARMIE's suspension system 41

Table 3.3 – Input variables for the parametric optimization of the static balancing mechanisms 48

Table 3.4 – Values of the constants β and A to estimate the minimum tensile strength of AISI 302 stainless steels, and the correction factors $n\mathrm{body}$, $n\mathrm{A}$, and $n\mathrm{B}$ to estimate the maximum allowable stresses for the safety factors calculations, adapted from (Budynas & Nisbett, 2011) 51

Table 3.5 – Best combinations of inputs for the parametric optimization of the squatting mechanism 52

Table 3.6 – Best combinations of inputs for the parametric optimization of the trunk flexion mechanism 53

Table 3.7 – Results for the best-performing solutions of the static balancing parametric optimization study 53

Table 3.8 – Maximum joint forces and torques generated in the arms of the CHARMIE robot 54

Table 3.9 – Properties of the motor-gearbox assemblies selected for the arms of the CHARMIE robot 55

Table 3.10 – Properties of the bevel gears used on CHARMIE and corresponding stresses and strengths 56

Table 3.11 – Shear Stresses and safety factors of the stainless-steel shafts of CHARMIE's arms 57

Table 4.1 – Coordinates of the points used to characterize the robot's torso represented in Figure 4.4. 66

Table 4.2 – Names and mass properties of CHARMIE's bodies. 67

Table 4.3 – Inertial properties the bodies from the multibody model of CHARMIE. 68

Table 4.4 – Properties defining the joints of CHARMIE. Bodies marked with * have variable values due to being bodies with variable lengths. 69

Table 4.5 – Joint actuator velocities, accelerations and position constraints for the CHARMIE robot 70
Table 4.6 – Mass properties of CHARMIE's base alongside the suspension and locomotion systems. 86
Table 4.7 – Inertial properties of CHARMIE's base alongside the suspension and locomotion systems. 86
Table 5.1 – Variables describing the status of the robot simulation environment 97
Table 5.2 – Logic table for the control of the acceleration of each joint using two neurons. 98
Table 5.3 – Variables describing the observation space of the neural network 98
Table 5.4 – Reward attribution for CHARMIE's motion based on the position of the manipulation claw 100
Table 5.5 – Start configuration of the CHARMIE robot for the neural-network agent training..... 101
Table 5.6 – Starting conditions for the ten scenarios used to train CHARMIE for generalized behaviors 105
Table 5.7 – Hyperparameters and architecture for the training of the PPO neural network..... 107
Table 5.8 – Tolerances for different scenarios for the training of CHARMIE's trajectory control. 108
Table A1.1 – Properties and design parameters of the bevel gears used in the arms of the CHARMIE robot..... 135
Table A1.2 – Bending strength geometry factors for straight-bevel gears with a 20o normal pressure angle and 90o shaft angle. 137
Table A1.3 – Pitting resistance geometry factors for straight-bevel gears with a 20o normal pressure angle and 90o shaft angle. 139
Table A1.4 – Equation factors for the stress and strength analysis of CHARMIE's bevel gears..... 139
Table A1.5 – Stresses, strengths, and safety factors of the bevel gear used in CHARMIE's arms..... 140
Table A2.1 – List of parts used for the CHARMIE robot 141
Table A5.1 – Local coordinates of body 1) Base 305
Table A5.2 – Local coordinates of bodies 1a\1d\1g\1j) Suspension system intermediary link..... 306
Table A5.3 – Local coordinates of bodies 1b\1e\1h\1k) DC motor assembly 306
Table A5.4 – Local coordinates of bodies 1c\1f\1i\1l) Swedish omni wheel 307
Table A5.5 – Local coordinates of body 2) Hip..... 308
Table A5.6 – Local coordinates of body 3) Torso 308
Table A5.7 – Local coordinates of body 4a) Left scapula..... 308
Table A5.8 – Local coordinates of body 5a) Left shoulder 308
Table A5.9 – Local coordinates of body 6a) Left proximal arm 309
Table A5.10 – Local coordinates of body 7a) Left distal arm..... 309

Table A5.11 – Local coordinates of body 8a) Left proximal forearm 309
Table A5.12 – Local coordinates of body 9a) Left distal forearm 309
Table A5.13 – Local coordinates of body 10a) Left hand 309
Table A5.14 – Local coordinates of body 11a) Left thumb claw 310
Table A5.15 – Local coordinates of body 11d) Left opposite claw 310
Table A5.16 – Local coordinates of body 4b) Right scapula 311
Table A5.17 – Local coordinates of body 5b) Right shoulder 311
Table A5.18 – Local coordinates of body 6b) Right proximal arm 311
Table A5.19 – Local coordinates of body 7b) Right distal arm 311
Table A5.20 – Local coordinates of body 8b) Right proximal forearm 311
Table A5.21 – Local coordinates of body 9b) Right distal forearm 311
Table A5.22 – Local coordinates of body 10b) Right hand 312
Table A5.23 – Local coordinates of body 11b) Right thumb claw 312
Table A5.24 – Local coordinates of body 11e) Right opposite claw 313
Table A5.25 – Local coordinates of body 4c) Lower cervical spine 313
Table A5.26 – Local coordinates of body 5c) Neck 314
Table A5.27 – Local coordinates of body 6c) Head 314
Table a1.1 – Properties of the compression springs used for CHARMIE's suspension system 315

List of Abbreviations

ANN	Artificial Neural Network
CAD	Computer-Aided Design
CHARMIE	Collaborative Healthcare/Home Assistant Robot by Minho Industrial Electronics
DOF	Degree of Freedom
DRL	Deep Reinforcement Learning
FDM	Fused Deposition Modeling
IDE	Integrated Development Environment
ML	Machine Learning
NN	Neural Network
PLA	Polylactic Acid
PPO	Proximal Policy Optimization
RL	Reinforcement Learning
TD	Temporal Difference

LIST OF SYMBOLS

Chapter 2

Latin Symbols

Symbol	Unit (SI)	Description
a, b, c, d	-	Real numbers for the definition of a quaternion
a_i	m	Length of link i
$c_1,\ c_2,\ c_3$	-	Cosine of the three angles for intrinsic ZXZ Euler rotations
d_i	m	Offset of joint i
$\mathbf{E}$	N.m	Vector of the moments resulting from external forces and torques applied to a body
$\mathbf{F}$	N	Total forces applied to a body
$\mathbf{F}_k$	N	Nonconservative forces applied to a body
$\mathbf{G}$	N.m	Vector of the moments resulting from gravitational forces
H	J	The total energy of a system
$\mathbf{H}_j^i$	-	Homogenous transformation matrix from frame i to frame j
$\mathbf{i}, \mathbf{j}, \mathbf{k}$	-	Basis vectors for the definition of a quaternion
$\mathbf{L}$	kg.m²/s	Angular momentum of a body
L	J	Lagrange function
m	kg	Mass of a body
$\mathbf{M}$	kg, kg.m²	Global mass matrix of a body
$\mathbf{M}_T$	N.m	Total moments applied to a body
$\mathbf{p}_k$	-	Generalized momenta of a system
q	-	Quaternion
$\mathbf{q}_k$	-	Generalized coordinates of a system
$\dot{\mathbf{q}}_k$	-	Generalized velocities of a system

Symbol	Unit (SI)	Description
$\ddot{\mathbf{Q}}$	m/s², rad/s²	Acceleration matrix of a system
$\mathbf{R}^{i}_{j}, \mathbf{R}^{i}_{k}, \mathbf{R}^{j}_{k}$	-	Rotation matrix from the coordinate frame in the superscript to the one in the subscript
$s_1,\ s_2,\ s_3$	-	Sine of the three angles for intrinsic ZXZ Euler rotations
t	s	Time
$\mathbf{T}$	N.m	Vector of applied torques
T	J	The kinetic energy of a body
U	J	The potential energy of a body
$\mathbf{v}$	m/s	Linear velocities of a body
$\mathbf{V}$	N.m	Vector of the moments resulting from centrifugal forces
$\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{z}}_i$	-	Basis vectors of the coordinate frame i
$\hat{\mathbf{x}}_j, \hat{\mathbf{y}}_j, \hat{\mathbf{z}}_j$	-	Basis vectors of the coordinate frame j
Z_1, X_2, Z_3	rad	Angle of the three rotations for intrinsic ZXZ Euler rotations

Greek Symbols

Symbol	Unit (SI)	Description
α_i	rad	Twist angle of link i
θ_i	rad	Joint angle of joint i

Chapter 3

Latin Symbols

Symbol	Unit (SI)	Description
a_i	m	Dimensional parameter defining the squatting static balancing mechanism
A	Pa.m$^{\beta}$	Parameter to estimate the minimum tensile strength of spring materials
$A_s x$, $A_s y$, $A_s z$	m	Cartesian coordinates of the spring endpoint A_s
b		Bevel great teeth face width
b_{body}	-	Correction factor to estimate the maximum allowable stress in the spring body
b_A	-	Correction factor to estimate the maximum bending and axial stresses in the spring hook
b_B	-	Correction factor to estimate the maximum shear stress in the spring hook
b_i	m	Parameter defining spring choice for the squatting static balancing mechanism
$B_s x$, $B_s y$, $B_s z$	m	Cartesian coordinates of the spring end point B_s
c_j , d_j	m	Dimensional parameters defining the trunk flexion static balancing mechanism
C_{body}	-	Variable confirming the structural integrity of the spring body
C_A	-	Variable confirming the structural integrity of the spring hook to bending and axial stresses
C_B	-	Variable confirming the structural integrity of the spring hook to shear stresses
C_{ys}	-	Variable confirming the tension spring will not be compressed
d	m	Spring wire diameter
d_p	m	Bevel gear pitch diameter
d_S	m	Shaft diameter
D	m	Spring mean coil diameter
D_E	m	Spring coil outer diameter
D_I	m	Spring coil inner diameter
E	Pa	Young's modulus
e_j	m	Parameter defining spring choice for the trunk flexion static balancing mechanism

Symbol	Unit (SI)	Description
f	-	Objective function
F_{A}	N	Axial force exerted on a shaft
F_{min}	N	Minimum possible force exerted by the suspension arm
F_{max}	N	Maximum possible force exerted by the suspension arm
F_{Avg}	N	Average force exerted by the linear actuator
F_{Max}	N	Maximum force exerted by the linear actuator
F_{spring}	N	Force exerted by the spring
F_{x} , F_{y} , F_{z}	N	Maximum forces exerted in the x, y and z axes of a joint
$\|F\|$	N	Maximum force magnitude exerted in a joint
g	m/s^2	Acceleration of gravity
G	Pa	Shear modulus
k	N/m	Spring constant
K_{A}	-	Overload factor for bevel gear analysis
$K_{\mathrm{H\beta}}$	-	Load-distribution factor for bevel gear analysis
K_{υ}	-	Dynamic factor for bevel gear analysis
K_{θ}	-	Temperature factor for bevel gear analysis
L	m	Spring coiled length
L_0	m	Spring free length
L_{S}	m	Spring solid length
m_{et}	m	Outer transverse module of a bevel gear
m_{total}	kg	Total mass of the robot
M_{F}	N.m	Bending moment exerted on a shaft
n_{body}	-	Safety factor calculated regarding the stress in the spring body
n_{A}	-	Safety factor calculated regarding the bending and axial stresses in the spring hook
n_{B}	-	Safety factor calculated regarding the shear stress in the spring hook
n_{spring}	-	Safety factor used for the mechanical design of the springs
N_{A}	-	Number of active spring coils
Pnt	-	Punishment function
Rwd	-	Reward function

Symbol	Unit (SI)	Description
S_F	-	Bending safety factor for bevel gear analysis
S_H	-	Contact safety factor for bevel gear analysis
Sc_{Max}	-	Score function evaluating the reduction of maximum actuator force
Sc_{Avg}	-	Score function evaluating the reduction of average actuator force
s_k	-	Sign variable defining the behavior of joint k
t	s	Time
T	N.m	Torsional moment exerted on a shaft
T_x, T_y, T_z	N.m	Maximum torques exerted in the x, y, and z axes of a joint
$\|T\|$	N.m	Maximum torque magnitude exerted in a joint
V	N	Transversal cutting force exerted on a shaft
W^t	N	Force applied on the teeth of a bevel gear
y_{spring}	m	Spring deformation
y_{min}	m	Minimum possible length of the suspension system's spring
y_{max}	m	Maximum possible length of the suspension system's spring
Y_J	-	Bending strength geometry factor for analysis of bevel gears
Y_{NT}	-	Stress-cycle factor for bending strength analysis of bevel gears
Y_x	-	Size factor for bending analysis of bevel gears
Y_z	-	First reliability factor for bevel gear analysis
Y_β	-	Lengthwise curvature factor for bending strength analysis of bevel gears
Z_E	-	Elastic coefficient for pitting resistance analysis of bevel gears
Z_I	-	Pitting resistance geometry factor for the analysis of bevel gears
Z_{NT}	-	Stress-cycle factor for pitting resistance analysis of bevel gears
Z_W	-	Hardness-ratio factor for bevel gear analysis
Z_x	-	Size factor for pitting resistance analysis of bevel gears
Z_{xc}	-	Crowning factor for pitting analysis of bevel gears
Z_Z	-	Second reliability factor for bevel gear analysis

Greek Symbols

Symbol	Unit (SI)	Description
β	-	Parameter to estimate the minimum tensile strength of spring materials
Δy_0	m	Spring deformation under normal balance conditions
Δt	s	Simulation timestep
μ_f	-	Average objective function value
ϑ_k	m, rad	Configuration of joint k
$\vartheta\mathrm{min}_\mathrm{k}$	m, rad	Minimum configuration of joint k
$\vartheta\mathrm{max}_\mathrm{k}$	m, rad	Maximum configuration of joint k
$\dot{\vartheta}_\mathrm{k}$	m/s, rad/s	Velocity of joint k
$\dot{\vartheta}\mathrm{max}_\mathrm{k}$	m/s, rad/s	Maximum velocity of joint k
$\ddot{\vartheta}_\mathrm{k}$	m/s², rad/s²	Acceleration of joint k
$\ddot{\vartheta}\mathrm{act}_\mathrm{k}$	m/s², rad/s²	Reference acceleration associated with joint k
σ_yield	Pa	Yield stress
σ_F	Pa	Bending stress exerted on the teeth of a gear
σ_FP	Pa	Strength of the teeth of a gear towards bending stress
σ_Flim	Pa	Allowable bending stress on the teeth of a gear
σ_H	Pa	Contact stress exerted on the teeth of a gear
σ_HP	Pa	Strength of the teeth of a gear towards contact stress
σ_Hlim	Pa	Allowable contact stress on the teeth of a gear
τ_{yield}	Pa	Yield shear stress
τ_max	Pa	Maximum shear stress for a round profile

Chapter 4

Latin Symbols

Symbol	Unit (SI)	Description
$0_{[3\times3]}$	-	Null matrix with three rows and three columns
$0_{[3\times1]}$	-	Null matrix with three rows and one column
$a_1,\ a_2,\ a_3,\ a_4$	N, N.m	Variables referring to the left-hand side of the equation to study the distribution of forces on the robot's torso
$a_s,\ b_s$	m	Lengths of the sides of a triangle to determine the amplitude of an angle
$a_{2e},\ b_{2e}, c_{2g}$	m	Lengths used to characterize the kinematics of the static balancing mechanism of the squatting motion
$\mathbf{A}_i^{'}$	m	Local coordinates of the connection point between link i and its preceding link
$\mathbf{A}_i^{''}$	m	Coordinates of the connection point between link i and its preceding link after the rotation operation around its origin
$\mathbf{B}_{i-1}$	m	Global coordinates of the connection point between link i and its preceding link
$c_1,\ c_2,\ c_3$	-	Cosine of the three angles for intrinsic ZXZ Euler rotations
c_a	N.s/rad	Suspension system angular dampening coefficient
c_b	N.s/m	Suspension system linear dampening coefficient
C_l	-	Binary variable identifying robot collisions with a counter
C_{can}	-	Binary variable identifying robot collisions with a metal can
$C_c x, C_c y, C_c z$	m	Global cartesian coordinates of the center of a metal can
d_i	m	Distance between the origin of link i and its connection point to the preceding link in the kinematic chain
d_{wheel}	m	Diameter of the robot's wheels
$\dot{d}_i$	m/s	Linear velocity of the prismatic joint preceding link i
$\ddot{d}_i$	m/s^2	Linear acceleration of the prismatic joint preceding link i
$e_2,\ l_2,\ h_2$	m	Auxiliary parameters for the analysis of the intersection between two circumferences
$\mathbf{f}$	N	Sum of forces exerted in a system
$\mathbf{f}_j^i$	N	Force exerted by link j preceding link i on link i expressed in the local coordinates of link i
$\mathbf{f}_k^k$	N	Force exerted by link k succeeding link i on link i expressed in the local coordinates of link k

Symbol	Unit (SI)	Description
$\mathbf{f}_{w}^{w}$	N	Force generated by the wheel-floor interaction expressed in the wheel's local reference
$F_{0Spring}$	N	Force resulting from the pre-tension of a tension spring
F_{Spring}	N	Force exerted by a tension spring
F_{Susp}	N	Force exerted by a suspension compression spring
g	m/s²	Acceleration of gravity
$\mathbf{I}_3$	-	3x3 identity matrix
$\overline{\mathbf{I}}$	kg.m²	Inertia matrix
$\overline{\mathbf{I}}_{i}^{i}$	kg.m²	Inertia matrix of link i
I_{xx}, I_{yy}, I_{zz}	kg.m²	Principal moments of inertia
I_{xy}, I_{xz}, I_{yz}	kg.m²	Product of inertia terms of the inertia matrix
j	-	Index of the links preceding link i
J	-	Number of links preceding link i
JT_i	-	Main joint type preceding link i
JO_i	-	Orientation of the main joint preceding link i
k	-	Index of the links succeeding link i
k_{Spring}	N/m	Tension spring constant
k_a	N/rad	Suspension torsion spring constant
k_b	N/m	Suspension compression spring constant
K	-	Number of links succeeding link i
l	m	Compression spring current length
l_0	m	Compression spring free-length
l_s	m	Calculated length of an auxiliary body
$\dot{l}$	m/s	Compression spring deformation velocity
L_0	m	Tension spring free-length
L_{Spring}	m	Tension spring current length
m	kg	Mass of the system
m_i	kg	Mass of link i
m_{total}	kg	Total mass of the robot
$\mathbf{O}_i^{'}$	m	Local coordinates of the origin point of link i

Symbol	Unit (SI)	Description
$\ddot{\mathbf{p}}$	m/s²	Linear acceleration of the system
$\ddot{p}_{\mathrm{x}}$	m/s²	Lateral acceleration of the robot
$\ddot{p}_{\mathrm{y}}$	m/s²	Forward acceleration of the robot
$\ddot{\mathbf{p}}_{\mathrm{Bi}}^{\mathrm{i}}$	m/s²	Linear acceleration of connection point between link i and its succeeding link in the local coordinates of link i
$\ddot{\mathbf{p}}_{\mathrm{Bi-1}}^{\mathrm{i-1}}$	m/s²	Linear acceleration of connection point between link i and its preceding link in the local coordinates of the link preceding i
$\ddot{\mathbf{p}}_{\mathrm{Ci}}^{\mathrm{i}}$	m/s²	Linear acceleration of the center of mass of link i in its local coordinates
$\mathbf{P}_{\mathrm{i}}$	m	Global coordinates of a generic point P of link i
$\mathbf{P}_{\mathrm{i}}^{'}$	m	Local coordinates of a generic point P of link i
$\mathbf{P}_{\mathrm{i}}^{'\mathrm{j}}$	m	Coordinates of a generic point P of link I expressed in the local coordinates of body j
$\mathbf{P}_{\mathrm{i}}^{''}$	m	Coordinates of a generic point P of link i after the rotation operation around its origin
$\mathrm{P_i}x\,,\mathrm{P_i}y\,,\mathrm{P_i}z$	m	Global cartesian coordinates of a generic point P of link i
$\mathrm{P}_{\mathrm{i}}^{'}x\,,\mathrm{P}_{\mathrm{i}}^{'}y\,,\mathrm{P}_{\mathrm{i}}^{'}z$	m	Local cartesian coordinates of a generic point P of link i
$\mathrm{P}_{\mathrm{i}}^{'\mathrm{j}}x\,,\mathrm{P}_{\mathrm{i}}^{'\mathrm{j}}y\,,\mathrm{P}_{\mathrm{i}}^{'\mathrm{j}}z$	m	Cartesian coordinates of a generic point P of link I expressed in the local coordinates of body j
r_{c}	m	Radius of a metal can
$\mathbf{r}_{\mathrm{Ai,Bi}}$	m	Vector defining the relative position between two consecutive joints in the local coordinates of link i
$\mathbf{r}_{\mathrm{Ai,Bk}}$	m	Vector defining the relative position between the main joint preceding link i and its succeeding joint k in the local coordinates of link i
$\mathbf{r}_{\mathrm{Ai,Ci}}$	m	Vector defining the position of the center of mass of link i concerning its main preceding joint in the local coordinates of link i
$\mathbf{r}_{\mathrm{Aj,Ci}}$	m	Vector defining the position of the center of mass of link i concerning its preceding joint j in the local coordinates of link i
$\mathbf{r}_{\mathrm{Oi,Bi}}$	m	Vector defining the position of a succeeding joint concerning the origin of link I in the local coordinates of link i
$\mathbf{r}_{\mathrm{Oi,Ci}}$	m	Vector defining the position of the center of mass of link i concerning its origin in the local coordinates of link i
$\mathbf{R}_{\mathrm{j}}^{\mathrm{i}}$	-	Generic rotation matrix from the coordinate frame in the superscript to the one in the subscript
$\mathbf{R}_{\mathrm{i}}^{\mathrm{i-1}}$	-	Rotation matrix converting the orientation of the link preceding i into the orientation of link i.
$\mathbf{R}_{\mathrm{i-1}}^{0}$	-	Rotation matrix describing the orientation of the link preceding i
$\mathbf{R}_{\mathrm{i}}^{0}$	-	Rotation matrix describing the orientation of link i

Symbol	Unit (SI)	Description
$\mathbf{R}_{k}^{i}$	-	Rotation matrix describing the rotation between link i and its succeeding link k
s_1 , s_2 , s_3	-	Sine of the three angles for intrinsic ZXZ Euler rotations
$S_i x$, $S_i y$, $S_i z$	m	Global coordinates of the first of the ends of a segment whose length will be determined
$S_j x$, $S_j y$, $S_j z$	m	Global coordinates of the second of the ends of a segment whose length will be determined
$\mathbf{t}_i$	-	Unit vector defining the orientation of the joint preceding link i
T_l	m	Threshold range to detect collisions of the robot with obstacles
w	-	Index of the robot's wheel assemblies
x_{min} , x_{max}	m	Abscissa maximum and minimum boundaries for a paralepidid obstacle
$\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, $\hat{\mathbf{z}}$	-	Basis vectors referring to the three cartesian coordinates
y_{min} , y_{max}	m	Ordinate maximum and minimum boundaries for a paralepidid obstacle
$\mathbf{z}_i$	-	Unit vector defining the orientation of the main joint preceding link i for the dynamic algorithm
z_{max}	m	Applicate maximum and boundary for a paralepidid obstacle
$z_{max\,C}$	m	Applicate coordinate maximum value for a metal can
Z_1 , X_2 , Z_3	rad	Angle of the three rotations for intrinsic ZXZ Euler rotations
ZXZEuler	-	Function to generate rotation matrices from the value of Euler angles

Greek Symbols

Symbol	Unit (SI)	Description
α_2	rad	Auxiliary angle to determine the orientation of the robot's torso
α_s	rad	Angle to be calculated to characterize the robot's geometry
β_3	rad	Angle defining the orientation of the displacement between the two main preceding joints of the torso of the robot
ϕ_{2a}	rad	Angle defining the rotation of the lower linear actuation in relation to the hip of the robot
ϕ_{3a}	rad	Angle defining the rotation of the upper linear actuation in relation to the torso of the robot
μ_{m1} , μ_{m2} , μ_{m3} , μ_{m4}	N.m	Torques exerted by the four locomotion motors
μ_m	N.m	Torque exerted by one of the robot's locomotion motors

Symbol	Unit (SI)	Description
μ_{susp}	N.m	Torque exerted by the suspension system on the robot's base
$\boldsymbol{\mu}$	N.m	Sum of torques exerted in a system
$\boldsymbol{\mu}_j^i$	N.m	Torque exerted by link j preceding link i on link i expressed in the local coordinates of link i.
$\boldsymbol{\mu}_k^k$	N.m	Torque exerted by link k succeeding link i on link i expressed in the local coordinates of link k.
θ	rad	Current angle of the torsion spring of the suspension system
θ_0	rad	Free position of the torsion spring of the suspension system
θ_3	rad	Rotation angle between the hip and torso of the robot
θ_z	rad	Yaw angle of the robot
$\dot{\theta}$	rad/s	Angular velocity of the torsion spring of the suspension system
$\dot{\theta}_i$	rad/s	Angular velocity of the revolute joint preceding link i
$\ddot{\theta}_i$	rad/s²	Angular acceleration of the revolute joint preceding link i
$\vartheta\min_k$	m, rad	Minimum configuration of joint k
$\vartheta\max_k$	m, rad	Maximum configuration of joint k
$\dot{\vartheta}\max_k$	m/s, rad/s	Maximum velocity of joint k
$\ddot{\vartheta}\mathrm{act}_k$	m/s², rad/s²	Reference acceleration associated with joint k
$\boldsymbol{\omega}$	rad/s	Angular velocities of the system
$\boldsymbol{\omega}_i^i$	rad/s	Angular velocities of link i in its local coordinates
$\boldsymbol{\omega}_{i-1}^{i-1}$	rad/s	Angular velocities of the link preceding i in its local coordinates
$\dot{\boldsymbol{\omega}}$	rad/s²	Angular acceleration of the system
$\dot{\boldsymbol{\omega}}_i^i$	rad/s²	Angular acceleration of link i in its local coordinates
$\dot{\boldsymbol{\omega}}_{i-1}^{i-1}$	rad/s²	Angular acceleration of the link preceding i in its local coordinates

Chapter 5

Latin Symbols

Symbol	Unit (SI)	Description
a_{n}	-	Action space of a system
f	-	Function
Rwd_2	-	Reward for the completion of a task
s_{n}	-	Current state space of a system
$s_{\mathrm{n+1}}$	-	Next state space of a system
t	s	Time
x	-	Current value of an observation space variable
x'	-	Normalized value of an observation space variable
$\min(x)$	-	Minimum possible value of an observation space variable
$\max(x)$	-	Maximum possible value of an observation space variable

"*No accomplishment has substance nearly as great as the road used to achieve it.*"
Brandon Sanderson

1. INTRODUCTION

Caregivers of persons with impaired mobility suffer intense strains, leading to the deterioration of their physical and mental health (Bom et al., 2018). Domestic robots can be employed to increase the autonomy of people with mobility limitations, enhancing the quality of life of both patients and caregivers alike. Robotic developments have already caused a massive positive shift in diverse fields, from revolutionizing industrial environments (Oztemel & Gursev, 2020) to providing vital tools for performing surgeries (Leal Ghezzi & Campos Corleta, 2016). For domestic tasks, most versatile manipulator robots are still in the prototyping stage due to the non-structured nature of the work environment, the need for adaptability to perform a variety of tasks, and the concerns regarding safety in human-robot interaction.

The technologies regarding actuation and sensors for developing domestic human-inspired robots are already commonplace, however, some developments are necessary in this field. The research gap is twofold, more finished products are required to test the products in applicable scenarios (this is often limited due to the cost-benefit relation of this kind of robots), and more robust and complete control solutions must be implemented. The construction of such a complex robotic system is a multidisciplinary task. Thus, the present work contributes to the development of a novel anthropomorphic mobile manipulator by focusing on its mechanical design, structure, and motion control by using a neural network model.

1.1. Motivation

The rapid aging of the worldwide population is one of the key concerns in modern societies (Rudnicka et al., 2020). Data from the (United Nations Department of Economic and Social Affairs - Population Division, 2022) shows the severity of this issue (Figure 1.1), especially in the regions of Europe, North America, Eastern Asia, South-Eastern Asia, Australia, and New Zealand. Current trends predict that this situation will worsen in the following years, with the most critical regions reaching a 25% population of older adults (aged 65 or over) by 2050. This gives rise to several pressing matters, such as ensuring the quality of life of older adults (van Leeuwen et al., 2019), mitigating the negative physical and mental impacts caused on the health of informal caregivers (unpaid care provided by the family or friends) (Bom et al., 2018), and managing resources to guarantee appropriate care (Fulmer et al., 2021).

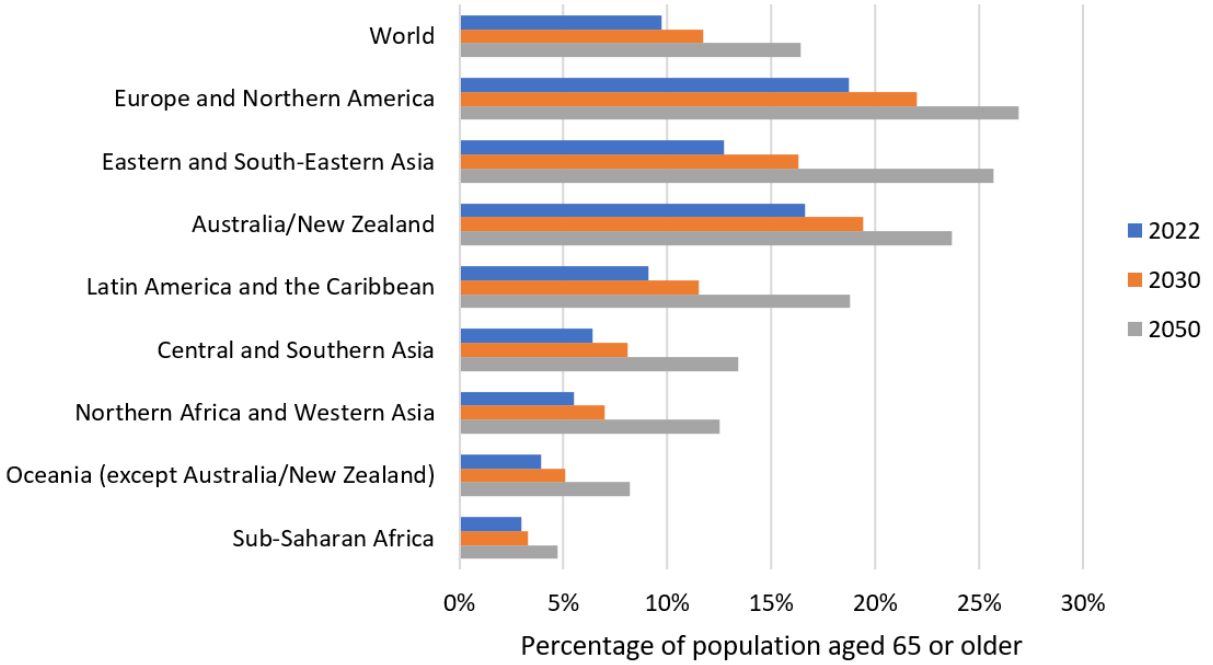


Figure 1.1 – Percentage of population aged 65 or over around the world for 2022, 2030, and 2050 according to the medium scenario predicted by (United Nations Department of Economic and Social Affairs - Population Division, 2022)

Older adults greatly favor adopting independent lifestyles for as long as it is possible (Peek et al., 2016), a process called "aging in place". Technology is an important ally in sustaining this practice, but its adoption and acceptance by older adults is a focal concern. The social and personal context of the users, as well as the outcomes of their previous interactions with modern devices, are major factors that affect the overall perception of technologies (Peek et al., 2016). The study by (Mitzner et al., 2010) reports older adults are receptive to newer technologies, using them and viewing them with mostly positive

attitudes. The main contributing factors for negative opinions were the technologies themselves creating inconveniences, providing unhelpful features, or raising concerns regarding safety and reliability. This highlights that the design and implementation process of novel devices can greatly affect their acceptance.

The role of robots in assisting older adults has been increasing with the most recent technological developments (Shishehgar et al., 2018). The perception of these robots is greatly affected by their intended applications. A survey by (Smarr et al., 2014) evaluated which tasks older adults prefer to be done by robots rather than human caretakers. Humans were considered preferable for personal care or health-related tasks (except reminding to take medications), as well as some social tasks, namely providing entertainment or calling family members. Robots were perceived as preferable for manipulating objects, managing information, and performing domestic chores (except meal-related tasks, such as cooking food, going grocery shopping, or setting up the table, all of which can have a heavy social component). Older adults also specifically reported they wanted robotic assistance to organize and tidy up shelves, books, groceries, drawers, etc.

In this PhD thesis, a novel anthropomorphic robot is developed to assist older adults. This new mobile manipulator, named Collaborative Healthcare/Home Assistant Robot by Minho Industrial Electronics (CHARMIE), considers the previously listed issues and preferences with the goal of better interacting with humans and performing chores in domestic environments. Ease of use is one of the main priorities, so intuitive commands can be given to the robot by using voice and gestures. Its range of motion allows manipulation of objects both on the floor and high to reach places, and it navigates using a wheeled omnidirectional base to enhance safety, stability, efficiency, and maneuverability in domestic indoor environments (Bruzzone & Quaglia, 2012). An anthropomorphic design was chosen to improve acceptance, communication, and the users' understanding of the robot's functionalities and capabilities (Duffy, 2003). A human-like shape also improves task performance in environments designed for human use.

1.2. Scope and Objectives

The focal point of this work is the development of a new anthropomorphic mobile manipulator robot. Developments in the field of robotics are multifaceted endeavors, so defining the proper scope and objectives was key for managing this work. The main goals of this project are twofold: obtaining a physical prototype of a novel anthropomorphic mobile manipulator robot, and creating a set of computational tools to analyze, optimize, and allow further research within this robot. The two goals are interconnected, the computational tools assist in the adequate design of the robot, while the physical prototype then provides experimental data to validate the theoretical approaches used in the computational models.

The prototype design was centered around the mechanical and structural design of the robot. The main objective was to first create conceptual designs, and then evolve these designs into a functional physical prototype. The fulfillment of this goal implies the selection of adequate normalized components, the design and manufacturing project of novel parts, and the study of how these parts interact and are assembled. This required the development and use of auxiliary tools, such as Computer-Aided Design (CAD) models to design the geometry of components and test their assembly, and the multibody simulation model to determine the loads and torques applied on each component, as well as to optimize the design of mechanisms. The total fulfillment of this task requires the completion of the following goals:

- to create a CAD model detailing the geometry and mass properties of the components of the robot, as well as the connections between them;
- to document an assembly guide to allow replicability of the work;
- to formulate the mathematical description of the motion of the designed mechanisms;
- to delineate the technical drawings for the production of non-standard components;
- to produce a physical prototype that must fulfill the design specification established in the conceptual stage, including locomotion and object interaction capabilities.

The analysis of multibody systems plays a key role in the development of robotic technologies. These simulators provide virtual landscapes to test control solutions and allow researchers to analyze the forces and torques that originated within the system; they are an initial step often integrated into the development of more advanced and specialized software solutions. The versatility and adaptability of the multibody model used in CHARMIE were maximized by implementing recursive algorithms to tackle the

fundamental kinematics and dynamics of the robot. This simulator is a key contribution developed within this PhD, and its development had to fulfill the following requirements:

- the simulator must be modular, and allow disabling its blocks for higher computational efficiency;
- the developed code must be hardware-independent, so it can also be used on the robot's embedded computer;
- the kinematics and dynamics of the multibody model must be studied in the simulator;
- the multibody model of the robot must be able to interact with its surroundings;
- the physical properties of any component (mass, geometry, inertia, ...) can be updated directly by accessing the input variables of the software;
- the timesteps of the simulation can be altered dynamically;
- the computational efficiency of the used algorithms must be prioritized;
- the system must be able to adequately integrate the testing and training of different neural network control solutions.

This work aims to contribute to the field of robotics in four main ways: developing and prototyping a new robot (CHARMIE); tackling key issues in the development of robotic systems in an interconnected and "start-to-finish" structure; adapting and implementing algorithms to analyze the dynamics of the robot to study and optimize its key mechanisms; and presenting the application and training of a reinforcement learning (RL) neural network (NN) for the trajectory control of a manipulator robot.

1.3. State of the Art

The present work tackles three key areas within the multidisciplinary field of robotics, namely: i) multibody computer simulation; ii) the development of new mobile manipulation robotic systems; iii) and trajectory control using neural networks.

Multibody computer simulation (i) plays a key role in the development of robotic systems, providing researchers with methods to gather data and evaluate the performance of mechatronic systems without requiring a physical prototype. The analysis of robotic systems using multibody simulations usually falls into one of two main methods: a) A commercial or open-source multibody simulation software

is used to model the robotic system and evaluate its performance, or b) The equations of motion that describe the system's motion are implemented into an integrated development environment, which then is used to run the code to study how the motion changes over time. Method a) is advantageous for its greater simplicity of use, the quality of graphic interfaces provided by the software, and the availability of computational programs that are already greatly optimized for specific problems. On the other hand, method b) allows further optimization of the computational model for the problem at hand (rather than using generalized and slower methods), it gives greater control and understanding of the model for the researchers, and the resulting code has greater compatibility with other software and higher portability for different hardware. Overall, no method is clearly advantageous over the other, the correct choice depends on the requirements of the project at hand.

The complexity of humanoid, and human-inspired robots, is a strong incentive for a majority of researchers to adopt commercial and open-source software solutions (Ivaldi et al., 2014). The state-of-the-art multibody simulators are constantly updated, and the appeal of simulators varies as more modern alternatives appear. One of the most predominant tools for simulation in engineering is MATLAB, a programming language, and numeric computing environment, which can integrate a set of add-ons and libraries (such as Simulink) for more specific problems (Žlajpah, 2008). Several robotic studies are done resorting to MATLAB simulations, such as studying obstacle avoidance algorithms of mobile robots (Rostami et al., 2019), planning the footsteps of legged humanoid robots on uneven terrain (Deits & Tedrake, 2014), and analyzing control methods for the motion of an industrial simulator (Urrea & Pascal, 2018). For more complex recent studies that integrate several aspects of robot simulation simultaneously, more modern alternatives have been growing in usage. V-REP is a resource intensive option (Pitonakova et al., 2018), but comprises a set of useful features such as multiple physics libraries and the ability for the user to interact with the world during simulation (Pitonakova et al., 2018). Some studies using V-REP for robotics include using the simulation environment to train deep reinforcement neural network either for vision-based object manipulation (Y.-L. Chen et al., 2023) or for trajectory control with obstacle avoidance (Sangiovanni et al., 2021), and to test the design of the navigation control of mobile robots using neuro-fuzzy logic system (Mishra et al., 2022). Another of the modern alternatives is ARGoS, which reduces the complexity of the environment and physics engine for increased computational performance (Pitonakova et al., 2018). The computational efficiency of this simulator makes it optimal for tackling intricate problems; most studies in the literature tackle the specific problem of swarm robotics (Khaldi et al., 2022; Pinciroli et al., 2018; Stolfi & Danoy, 2022).

The third popular modern software is Gazebo, which despite being close to V-REP in terms of features while having improvements in computational efficiency, is reportedly hard to use with several limitations (Pitonakova et al., 2018). Uses of Gazebo include evaluating the performance of a construction task teaching solution through demonstration (Liang et al., 2020), testing a person-following algorithm for wheeled autonomous mobile robots (Chebotareva et al., 2020), and the simulation of a robotic grasping control solution which employs deep reinforcement learning (Joshi et al., 2020).

The major alternative to using commercial or open-source simulators is to implement in-house software specialized for tackling the problems at hand. This is achieved by implementing the equations of motion that describe the physical behaviors of the system into an Integrated Development Environment (IDE) and using it to evaluate the system's evolution over time. The implementation of the equations of motion, and other required formulations, can be facilitated with the use of computational libraries. Developing in-house software implies a more laborious setup process, but it is a growing practice in engineering research where a great emphasis is placed on code performance, maintainability, portability, scalability, and reusability (Arvanitou et al., 2021). This process produces optimized solutions, that can be hardware-independent and have greater compatibility with other computational tools. The developed software is often characterized by the employed mathematical formulations. The equations of motions can be deduced by employing the Newton-Euler formulation, (Ghaedrahmati et al., 2019), the Lagrange formulation (Emam, 2020), or the Hamilton formulation (Turno & Malczyk, 2023). Then, more specific problems, such as contact mechanics (Rodrigues da Silva et al., 2022), the correct modeling of wheeled locomotion (Z. Jia et al., 2013), or solving the dynamics of closed kinematic loops (Marques et al., 2021), can be addressed by adding mathematical formulations compatible with the used notation. The process of obtaining multibody models using a specific notation can be systematized into a methodology, such as recursive Newton-Euler algorithms (Featherstone, 2008; Müller, 2018) or divide-and-conquer algorithms based on either Langrage's formulation (Malczyk & Frączek, 2012; Poursina & Anderson, 2013) or Hamilton's equations (K. Chadaj et al., 2017). The literature provides examples of research that uses in-house code to study dynamics. The study by (Huynh et al., 2020) analyses the varying dynamics of a KUKA KR90 robotic arm for machining operations. The research by (Z. Wang et al., 2016) uses the absolute nodal coordinate formulation alongside Chebyshev sampling methods to propose a method for efficiently solving the dynamics of rigid-flexible multibody systems with uncertain physical parameters. Additionally, the work by (Corral et al., 2020) presents a general approach for modeling the locomotion of biped robots, which is then used to study the influence of different integrators and normal contact force

models. The paper by (Muhayyuddin et al., 2018) proposes a physics-based motion control strategy for robot-object interaction in uncertain environments. Furthermore, the study (Kobilarov, 2014) uses multibody dynamics to develop stabilizing control laws for the trajectory control of aerial manipulators.

Mobile manipulation is one of the most complex and intensely researched topics in robotics. It's research is a key factor towards advanced automation, allowing increases in versatility when compared to fixed manipulation. Knowledge generated from mobile manipulation tackles the integration, interaction, and optimization of a range of relevant problems, such as computer vision, kinematic control, autonomous locomotion, sensor and actuator integration, and real-time decision-making.

A mobile manipulator is a redundant manipulator with associated locomotive capabilities. In (Long Jin et al., 2018) redundant manipulators are defined as manipulators which have more Degrees of Freedom (DOF) than those required to perform tasks, allowing for optimal motions. By adding mobility to a redundant manipulator, its workspace is expanded limitlessly.

Before starting a robotics project, it is of utmost importance to identify the design trends in mobile manipulation. This was achieved by studying 174 mobile manipulators, developed by 69 different scientific or commercial entities. The obtained robots were grouped into nine possible main applications: Domestic; Customer Service; Industry; Agriculture; Military; Rescue and Emergency Response; Space; Warehouse Management; and Research, as well as into 4 possible structural configurations:

- Redundant manipulator mounter on a mobile base (shortened in this work to Mounted Manipulator), the simplest design choice consisting of one or more manipulator arms mounted in a base with associated locomotive capabilities;
- Animal-inspired robots, which are shaped to mimic different animals such as mammals, insects, or lizards; they often feature legged locomotion and are optimized for outdoor applications requiring navigation through unstructured environments;
- Human-inspired robots present distinctive features from human morphology, such as a head/torso/arms upper body structure, but also have particular characteristics that differentiate them from humanoid robots, most commonly a wheeled locomotive system;
- Humanoid Robots, which intend to fully mimic human morphology and motion with as much accuracy as it is functionally possible, often adopt bipedal locomotion, employing two manipulator arms, and presenting a clear torso and head structure.

The studied robots were sorted into two categories (application and design) to find common trends. Two limitations were identified during this classification: i) several robots are part of ongoing projects, and their applications can shift over time (an example is ATLAS from Boston Dynamics, a research robot adapted by Team IHMC to perform as a Rescue and Emergency Response robot in the Darpa Robotics Challange (Johnson et al., 2015)); and ii) by definition, all robots studied are mounted manipulators, and in limited specific cases, the boundary between categorizing a robot as a Mounted Manipulator or a Human-Inspired Robot can be tenuous. Despite these shortcomings, the described limitations were present in a few cases, and the number of studied robots prevented them from compromising the validity of the studied sample.

The classification of 37 mobile manipulator robots from the sample of 174 is shown in Table 1.1 accompanied by supporting references to the relevant associated literature.

One of the most noticeable trends is that for applications where human interaction is paramount, such as customer service or work in domestic environments, the use of human-like designs is heavily favored. The literature justifies this approach, as stated by (Fink, 2012) "One approach to enhance people's acceptance of robots is the attempt to increase a robot's familiarity by using anthropomorphic (humanlike) design and "human social" characteristics. (...) the role of anthropomorphism in robotics is not to build an artificial human but rather to take advantage of it as a mechanism through which social interaction can be facilitated." (p. 199). However, these empathy-generating mechanisms can backfire and reduce robot acceptance. One of the known possible limitations of this approach is the uncanny valley effect, whereby taking the wrong steps in the design stage, the human-like features can cause repulsion rather than empathy (Mori et al., 2012).

The analyzed robots were also sorted according to their current development stage. Figure 1.2 illustrates the state of robots inserted into specific combinations of application and design. Outliers exist for each category; this image represents the state of most robots.

Table 1.1 – Examples of robotic mobile manipulator systems discussed in the literature grouped by application and design.

Robot	Citation	Application	Design
TIAGo	(Grama & Rusu, 2018)	Domestic	Mounted Manipulator
HSR	(Yamamoto et al., 2019)	Domestic	Mounted Manipulator
Justina	(Negrete et al., 2018)	Domestic	Human-Inspired
AMIGO	(Paletta et al., 2018)	Domestic	Human-Inspired
MoRo	(Bogue, 2017)	Domestic	Human-Inspired
ARMAR-III	(Asfour et al., 2006)	Domestic	Human-Inspired
Cosero	(Stückler et al., 2016)	Domestic	Human-Inspired
Pepper	(Pandey & Gelin, 2018)	Customer Service	Human-Inspired
Orihime-D	(Takeuchi et al., 2020)	Customer Service	Human-Inspired
Xiaoyi	(Pee et al., 2019)	Customer Service	Human-Inspired
REEM-C	(Diaz-Boladeras et al., 2015)	Customer Service	Human-Inspired
Fetch	(Zhang et al., 2022)	Industry	Mounted Manipulator
Little Helper	(Madsen et al., 2015)	Industry	Mounted Manipulator
RB-KAIROS	(Indri et al., 2019)	Industry	Mounted Manipulator
Grape	(Roure et al., 2018)	Agriculture	Mounted Manipulator
Harvey	(Lehnert et al., 2017)	Agriculture	Mounted Manipulator
Packbot	(Yamauchi, 2004)	Military	Mounted Manipulator
ANDROS Wolverine	(Murphy et al., 2008)	Emergency and Rescue	Mounted Manipulator
TALON	(Kawatsuma et al., 2012)	Emergency and Rescue	Mounted Manipulator
Warrior	(Kawatsuma et al., 2012)	Emergency and Rescue	Mounted Manipulator
DRC-HUBO+	(Oh et al., 2017)	Emergency and Rescue	Humanoid
ATLAS	(Johnson et al., 2015)	Emergency and Rescue	Humanoid
Chimp	(Stentz et al., 2015)	Emergency and Rescue	Animal-Inspired
Centauro	(Klamt et al., 2019)	Emergency and Rescue	Animal-Inspired
Spirit MER	(Grotzinger et al., 2012)	Space	Mounted Manipulator
Curiosity	(Grotzinger et al., 2012)	Space	Mounted Manipulator
Valkyrie	(Radford et al., 2015)	Space	Humanoid
Kiva Robot	(Wurman et al., 2008)	Warehouse	Mounted Manipulator
Kengoro	(Asano et al., 2016)	Research	Humanoid
BHR-5	(Yu et al., 2014)	Research	Humanoid
AILA	(Lemburg et al., 2011)	Research	Humanoid
KHR-3 HUBO	(Park et al., 2005)	Research	Humanoid
Toro	(Englsberger et al., 2015)	Research	Humanoid
HRP-5	(Kaneko et al., 2019)	Research	Humanoid
Rollin' Justin	(Fuchs et al., 2009)	Research	Human-Inspired
ANYmal	(Hutter et al., 2017)	Research	Animal-Inspired
Lauron V	(Roennau et al., 2014)	Research	Animal-Inspired

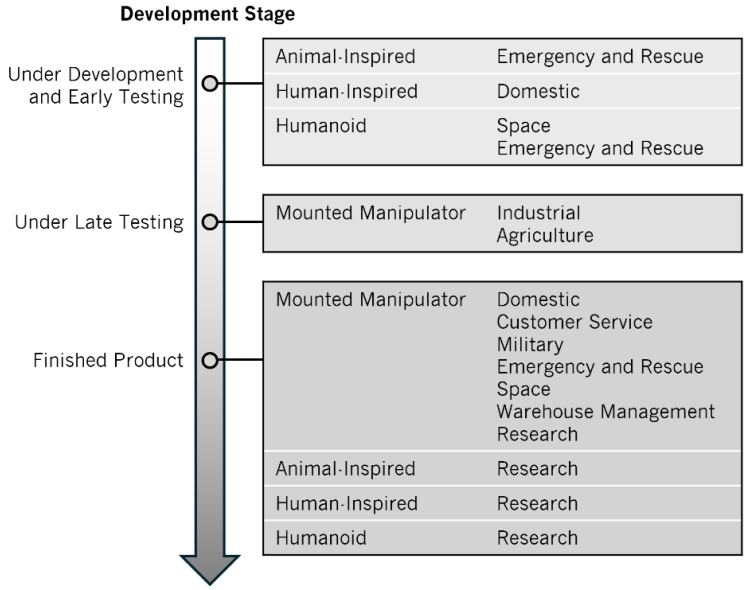


Figure 1.2 – Trends in the development of mobile manipulator robots considering design and applications.

The trends portrayed in Figure 1.2 are easily justified. Most finished products are mounted manipulator robots due to their higher simplicity of design. The only mounted manipulators that require additional testing are for industrial applications, due to the need to comply with safety standards to avoid the possibility of accidents in dangerous work environments alongside human workers, and for agricultural applications due to both the navigation environment and the target of manipulation being highly variable and unstructured. The requirements for control of the robot are decreased when the robot is remotely controlled, such as for military or emergency and rescue missions, or the work environment is highly structured, for example, customer service. The animal-inspired and humanoid emergency and response robots are mostly limited by the need for robust structure and control solutions to navigate through rough unstructured terrains. The main limiting factor to the development of domestic human-inspired robots is the need to: i) guarantee safe human-robot interaction; and ii) achieve autonomous and robust control solutions for highly unstructured indoor environments.

This study identified the relationship between the design types and intended application for mobile robots, the current challenges being tackled for each configuration, and the current state of the art of the industry. This information was pivotal in defining the key characteristics of the CHARMIE robot, such as the choice of locomotion type and the robot morphology.

Neural networks are, as described by (Y.-Y. Chen et al., 2019), "...powerful connectionist systems vaguely inspired by biological neural networks." (p. 3, adapted). Neural networks are gathering growing interest in the control of mobile manipulators for several reasons. One of the main advantages of using

neural networks is their ability to be trained online which allows the control scheme to compensate for uncertainties between the physical model and the idealized dynamic model of the robot such as the different drag coefficients of each joint, and the wearing of components (Jung & Hsia, 1996). Furthermore, control solutions generated using neural networks can be highly generalizable, which makes them ideal for dealing with unstructured environments. The correct selection of sensors and actuators is vital for the construction of a robust neural network control solution.

For the control of mobile manipulators in domestic environments, learning-based approaches are highly favored (J. Kim et al., 2019). The three most used learning approaches are:

- Learning from Demonstration, where first the intended behavior is inputted into the robot (via joint properties or computer vision), in a second stage these inputs are processed to generate trajectories via one of several mathematical methods, and finally, those trajectories and the learned behavior is used and extrapolated for the problem at hand;
- Supervised Learning, where a Neural Network is used to map the relations between inputs and outputs based on a training data set. By using sensor information as input data and actuator values as output data, generalizable trajectory control solutions for specific manipulation tasks can be obtained;
- Reinforcement Learning is where a set of rewards are used to define the goals of the system. By testing different actions at each state, and analyzing the resulting rewards, the neural network is trained to pick the most optimal actions at each given time.

One of the main current demands in the development of robotic systems is the increase in their complexity and the desire to increase their versatility. In turn, this increases the popularity of Reinforcement Learning as an appealing solution, since it results in adaptable control systems more suited to deal with dynamic environments (Polydoros & Nalpantidis, 2017). Reinforcement Learning has been used with great success for the control of fixed manipulators, however, most of the proposed solutions are not yet applicable to mobile manipulators due to the added complexity of the problem (C. Wang et al., 2020). Some recent works dealing with the neural network control in manipulator robots include using Deep Reinforcement Learning to control a 7-DOF KUKA robot arm for assembly tasks in industrial environments (F. Li et al., 2019), developing a Deep Reinforcement Learning control scheme for both the navigation and manipulation of a mobile manipulator robot which could enhance its applicability in industrial environments (Iriondo et al., 2019), obtaining a Reinforcement Learning solution

for the interactive navigation (which includes opening doors or pushing obstacles) of mobile manipulator robots (Chengshu Li et al., 2019), and achieving a Reinforcement Learning control system to learn complex object manipulation and grasping strategies in cluttered scenes with obstructed objects (Fujita et al., 2020).

Model-based machine learning is one of the current key areas of research within Reinforcement Learning (Chong Li, 2019). The principle is to use a virtual model (such as a multibody simulation software) to train an agent in a "consequence-free" environment, and then transfer this learned neural network into the physical model for a final training step.

The research of the state of the art of these three key fields: multibody dynamics, robot design, and neural network control, shows how they are interconnected. These three fields are a current key focus of the scientific community for the development of new and more versatile robotic systems.

1.4. Structure of the Thesis

This thesis aims at the design, development, and control of a domestic human-inspired mobile manipulator robot. This includes the mechanical project and design and the robot, followed by the production of a physical prototype, the modeling of the robot's key kinematics and dynamics in a computational multibody simulator, and the conception of a neural network control solution for a versatile control of the mobile manipulator system.

The first chapter presents the main motivations and objectives of this research work. This information is complemented and supported by a state-of-the-art analysis focusing on the key topics of the research in the field of robotics. Finally, the main contributions of this study are highlighted.

Chapter Two provides a more in-depth literature review looking at the main theoretical concepts approached in this work. First, the main theoretical considerations regarding the development of mobile manipulator robots are approach. This information is complemented by a study of the key required concepts of mechanism design. The methodologies used for the simulation of the dynamics of multibody systems in robotics are then listed, compared, and analyzed. The use of neural networks for the control of mobile manipulators is then studied to determine the most appropriate choice for the CHARMIE robot.

In Chapter Three, the main decisions regarding the morphology and optimization of the robot's physical model are discussed. This process starts with establishing the main specifications of the robot,

which then are used to determine its structural design. The robot's mechanisms are then optimized using computer simulation and mathematical formulations. Static balancing solutions are achieved that reduce the robot's energy consumption. After the robot's computational model is optimized, the construction of its physical prototype for testing is addressed.

The focus of Chapter Four is the development of the multibody model of the CHARMIE robot. The process is initiated by fully characterizing and describing the robot's bodies and joints. Then the Equations of Motion are determined by a Forward Kinematic analysis and an Inverse Dynamic analysis which both employ recursive algorithms. The simulator is completed by adding environment interaction to the robot's computational model.

Chapter Five deals with the development of the Reinforcement Learning trajectory control solution for the CHARMIE robot, and its implementation and training in a computational environment. This is started by establishing the specific environment where the neural network will be tested. Then, the preliminary architecture of the Deep Proximal Policy Optimization Network to be used is established. The agent is then trained in the simulation environment, which allows an analysis of the results to further improve the neural network's performance.

Finally, Chapter Six presents the key conclusions of the work. This chapter also proposes a set of future research lines for the further development of mobile manipulator systems and their control.

1.5. Contributions of this Work

This thesis presents a new human-inspired mobile manipulator robot that is being developed to work in domestic environments and assist people with mobility limitations in performing their daily tasks.

One of the key innovations of the human-inspired model is the static balancing mechanism that reduces the energy consumption of the hip and torso motions. These motions increase the workspace of the robot. Since the robot is a mobile manipulator, the energy supply is limited, and minimizing energy consumption is one of the main concerns. The static balancing mechanism is optimized using a parametric optimization approach which employs the multibody dynamic simulation to compare the system's behavior under different geometric and spring constant parameters.

Concerning multibody dynamics, a novel seven-step systematic and generalizable methodology was developed. This methodology interconnects and expands a set of mathematical formulations already

well-known in the literature and optimizes them for tackling branching kinematic chains which also integrate closed and over-constrained loops. This methodology was used to develop an own-code multibody simulator of CHARMIE, whose results were validated by the physical prototype.

The developed multibody simulator using the seven-step methodology is computationally efficient and hardware-independent, allowing it to be integrated into the robot's embedded computer. This allows the multibody model to be updated online using sensor and actuator data, providing more accurate results over time.

The efficient computational model of CHARMIE was used as the basis for the model-based Machine Learning training of the neural network. This neural network is responsible for controlling the trajectory of the CHARMIE robot. The obtained control solution is adaptable and versatile, becoming more optimal at tackling unstructured indoor environments, a common limitation of domestic robots.

"*Surprising what you can dig out of books if you read long enough, isn't it?*"
Robert Jordan

2. LITERATURE REVIEW

Outside of controlled environments, mobile manipulator robots still have very limited applicability (Kemp et al., 2007). For applications in domestic environments, two main requirements must be fulfilled: the design of the robot must be adequate to be accepted by humans (Beer et al., 2012), and the control solution must be adequate for dealing with cluttered and unstructured environments, such as using Reinforcement Learning approaches (J. Kim et al., 2019). The development of multibody computational models is highly advantageous for both the robot's design and the training of the neural networks in a virtual environment. Mobile robots are limited by their finite energy supply (Canfield et al., 2019), so mechanism design can be employed alongside the results from the multibody model to develop mechanical systems to reduce energy consumption. This chapter presents a literature review overarching these main interconnected topics, essential for the development of a robotic system.

2.1. Mobile Manipulator Robot Design

The design of a robot is mainly determined by its intended application. In a social application, such as for domestic robots, the design must not only consider the functionality of the robot but also its social acceptance and the willingness of humans to interact with it (Beer et al., 2012). In the context of this work, the design of the robot incorporated four key decisions: the morphology and shape of the robot dictated by the application, the choice of adequate locomotion system determined by the work environment, the configuration of the trunk defined by the requirements established for the robot, and the type of manipulation arms to be used which define the robot's range of motion and the types of objects it can interact with.

2.1.1. Robot Morphology

The morphology of robots is greatly influenced by their main tasks, in CHARMIE's case this is to perform chores in domestic environments to assist humans with mobility limitations.

The robot must be designed to fulfill a need. Survey data shows that older adults have a great interest in robots for manipulating objects (Smarr et al., 2014). These tasks include transporting heavy items and especially picking up objects from the ground, where the required motions can be hazardous for the health and comfort of older adults (Beer et al., 2012). This emphasizes the need for mobile manipulation in domestic environments.

The robot will perform in a human-engineered environment, created by and for humans. Contrary to an industrial environment, it is the robot that must adapt to the workplace it is inserted into. The performance of the robot can be greatly improved by using a human-inspired design (Radford et al., 2015) with human-sized proportions (Smarr et al., 2014).

The acceptance and social interaction of robots is also boosted by using a human-inspired design (Fink, 2012). The goal is not to create a perfect replica of a human but to use human characteristics to boost acceptance and manage the expectations of the robot. The work by (DiSalvo et al., 2002) enunciates three key characteristics for the design of robot heads, that can be applied to the entire design of social robots: “the need to retain an amount of robot-ness so that the user does not develop false expectations of the robots emotional abilities but realizes its machine capabilities, the need to project an amount of humanness so that the user will feel comfortable socially engaging the robot, and the need to convey an amount of product-ness so that the user will feel comfortable using the robot. Together, these suggestions create a balance between what we expect of a human, a robot, and a product for an effective design.” Considering the human-like design of these robots, the motion of the robot must also correspond to expectations, or else the Uncanny Valley effect would hinder human interaction instead of enhancing it (Mori et al., 2012).

Confirming the trends established in the State-of-The-Art review, the literature review confirms that the ideal morphology for the robot will be to adopt a human-inspired design with human proportions.

2.1.2. Locomotion System

The three most common choices for robotic locomotion are: i) legged locomotion, ii) tracked locomotion, iii) wheeled locomotion, or a hybrid solution between the three aforementioned options (Bruzzone & Quaglia, 2012). The best solution depends on the navigation environment.

Domestic robots must operate in unstructured indoor environments. Regarding mobility, legged robots are the optimal solutions, capable of climbing ladders and traversing obstacles. However, legged robots have very high energy consumption, and are mostly not statically stable, which can be prone to serious accidents for a domestic robot. Tracked robots are not ideal for indoor environments since the tracks can easily damage the floors or get stuck in fabrics from carpets. Therefore, to minimize energy consumption, maximize safety, and ease the assembly and maintenance of the robot, wheeled locomotion is the ideal choice (Bruzzone & Quaglia, 2012). This choice accepts the compromise that the robot will not be able to traverse steps.

The use of conventional wheels is the wheel type which provides higher speed, simplicity, and robustness while allowing the robot to traverse higher obstacles. However, when dealing with confined indoor spaces, omnidirectional wheels (such as the Swedish omni-wheel and the Mecanum wheel) provide much higher maneuverability, allowing the robot to rotate upon itself or initiate a linear movement in any direction at any given time (Shabalina et al., 2018).

The two main choices for omnidirectional wheeled locomotion are Swedish omni-wheels or the use of Mecanum wheels. To navigate flat floors, as common in indoor environments, Swedish wheels are preferable due to the lower vibrations generated during locomotion (Hijikata et al., 2023). Theoretically speaking, three Swedish omni-wheels are sufficient to establish omnidirectional locomotion with full control, however, the use of four or more wheels is associated with higher stability for the robot and a greater payload for the locomotion system (Hijikata et al., 2023). When 4 or more omnidirectional wheels are used, implementing a suspension system is highly advantageous to facilitate the robot's trajectory control by guaranteeing continuous contact of all four wheels with the floor plane (Siegwart et al., 2011).

2.1.3. Trunk

The trunk of a human-inspired mobile manipulator robot performs three main roles: to house some of the key electronic components, to give the robot a more human-like shape, and to perform the physical and electronic connection to the robot's different limbs and head.

In cases where simplicity is the pivotal focus, the trunk of the robot can be a simple fixed structure, with no degrees of freedom associated with it (the degrees of freedom only start at the connection with the limbs/head), such is the case for the robots in (Bajones et al., 2018; Graf et al., 2004). However, as the robot's morphology approaches anatomical proportions, the workspace of the robot diminishes, and the capability to interact with objects on the floor is lost without adequate torso and leg motions.

As discussed in section 2.1.1, interaction with objects on the floor is one of the main requirements of the robot. When wheeled locomotion is used, this is achieved by using an articulated trunk. The use of a flexible spine is possible (Mizuuchi et al., 2002), but challenging and complex. The most common solution is to use a mechanism at the base of the trunk, referred to in the literature as torso mechanism, robot waist, or robot hip, which can have one (Cheng & Ji, 2016; Negrete et al., 2018; Sturm et al., 2010), two (Agravante et al., 2013), three (Asfour et al., 2008) or more (Fiorio et al., 2017; Fuchs et al., 2009) DOFs.

The number of DOF from the robot hip is determined by the desired range of motion, the intended applications, how detrimental the increase in complexity is for the performance of the robot, and the degree of anthropomorphism desired from the robot.

2.1.4. Manipulator Arms

Manipulator arms are often categorized based on their number of DOFs. Considering the locomotion capacities of a mobile manipulator, three, four, or five DOFs are sufficient to obtain a robot with a wide workspace (Pandey & Gelin, 2018; Teng et al., 2020). However, by adding more DOF, redundant manipulators can avoid collisions, optimize trajectories and the object interaction itself, achieve higher dexterity, and attain wider workspaces (Long Jin et al., 2018). The goal for each application is to find the ideal compromise between complexity, desired workspace and dexterity, and the degree of human likeness.

The human arm can be described as having seven revolute DOFs, and the musculoskeletal structure is often approximated as a three-revolute DOF spherical joint shoulder, a single revolute DOF elbow, and a three revolute DOF spherical joint wrist (Yang et al., 2005). For robotic manipulator manufacturing, these joints are often converted into single-DOF revolute joints, resulting in the serial manipulator shown in Figure 2.1.b. However, if the goal of a system is to completely mimic the motion of the human body, the scapulohumeral rhythm motion of the inner shoulder must be considered, which can increase the number of DOF to 8 or even 9 (see Figure 2.1.a) (J. Lee & Cho, 2023). On the other hand, if the focus is simplifying the mechanism, the wrist ulnar deviation (Joint J_8) can be neglected due to its limited range of motion (Leal-Naranjo et al., 2017), resulting in a 6-DOF arm (see Figure 2.1.c).

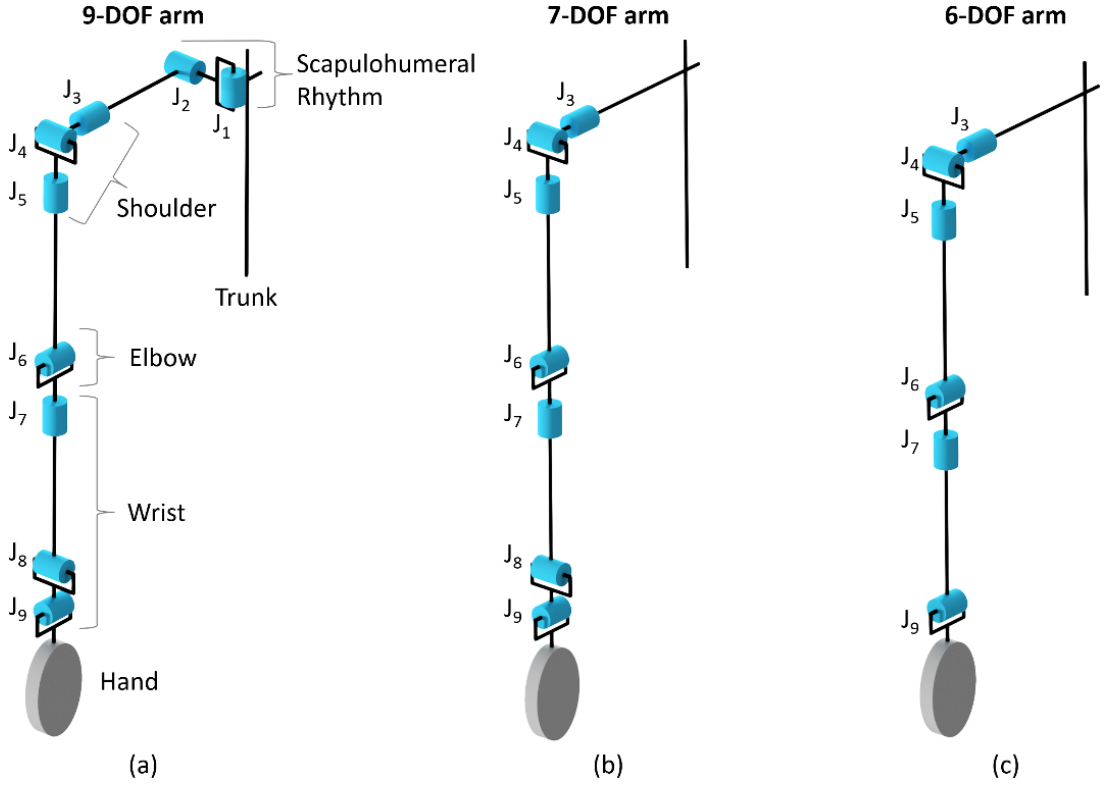


Figure 2.1 – Comparison of the kinematics for a) 9-DOF, b) 7-DOF, and c) 6-DOF robotic manipulator arm.

From the examples above, seven-DOF arms are the most common for human-inspired mobile manipulator robots, such as in the robots Justina (Negrete et al., 2018), ARMAR-III (Asfour & Dillmann, 2003), TIAGo (Bajrami et al., 2024), and REEM-C (Diaz-Boladeras et al., 2015). Some developed robots found it advantageous to simplify to six-DOF arms, for example, Moro (Bogue, 2017), Orihime-D (Takeuchi et al., 2020), and DRACO 3 (Bang et al., 2023). The alternative of increasing the number of DOFs above seven is not commonly used for practical implementations, only for specific research topics.

2.2. Mechanism Synthesis

The synthesis of mechanisms is the process of determining all parameters that define a mechanism: morphology, number of bodies, dimensions, joints, ... to obtain a desired output motion. The synthesis of mechanisms is divided into three stages (Hartenburg & Denavit, 1964):

- Type Synthesis refers to determining which mechanism configuration will be used, for example, a cam mechanism, a gear system, or a linkage. This step is often governed by non-mathematical steps, such as the knowledge and experience of the designer, the desired motion, the intended application, and the costs of the system.
- Number Synthesis is the process of using design tools to determine the optimal number of links and joints that result in the desired motion.
- Dimension Synthesis is when the categorization of the mechanism is finalized by determining the morphology and properties of each link and joint. This step can easily be modeled as a mathematical optimization problem, and it is the stage of mechanism design where the use of computational tools is more prevalent (Fomin et al., 2021).

This subsection tackles the literature review supporting the type synthesis and the number synthesis of two mechanical subsystems vital for the CHARMIE robot, namely the suspension system, and the static balancing system which was employed in the robot's trunk motion.

2.2.1. Suspension System

The goal of a suspension system is twofold: i) to keep all the wheels of a vehicle/robot in contact with the floor by increasing the mobility of the system, and ii) to dampen the effects of obstacles and irregularities in the robot's stability and reduce the propagation of vibrations within the system.

Most developments in suspension mechanisms for wheeled locomotion originate from the automotive industry. Here, suspension systems are often grouped into two main categories. Dependant, or rigid axle, suspensions use the structure of automobiles to simplify the suspension by acting upon the axle that supports a pair of wheels. On the other hand, independent suspensions allow each wheel to move independently, providing greater comfort and increased efficiency in reducing vibrations.

Mobile robots for confined indoor spaces mostly use omnidirectional wheels. When Swedish omni wheels are used, they are distributed in an axisymmetric configuration. In these cases, for construction

simplicity, each wheel is linked to an independent axle. The use of individual axles motivates the use of independent suspension systems.

One of the main distinguishing characteristics of comparing suspension types for robotic or automotive applications is the wheel type. In automotive suspensions, control over the variation of the camber angle can be used to obtain idealized steering configurations, however, most omnidirectional wheels are intended for a constant neutral camber angle.

Standard spring-based suspension mechanisms can fall into four main types. The simplest configuration is to directly use a prismatic joint and dampen its motion using a compression spring (Figure 2.2.a). This method is not used in the automotive industry since high radial loads on the prismatic joint could damage it, and it imposes a neutral camber angle. Furthermore, the use of prismatic joints tends to result in harder-to-manufacture systems. A MacPherson strut (Figure 2.2.b) is the most compact and easy-to-manufacture solution, however, it imposes a variable camber angle on the wheel (Reddy et al., 2016). The geometry of a Double Wishbone suspension (Figure 2.2.c) can be altered to obtain the desired variation of camber angle, including maintaining it neutral and constant. The main drawback of using the double wishbone system is the increased complexity and size of the system, which is not viable for some robotic applications. The fourth mechanism type is to use a multi-link suspension. This name is given to any independent suspension with three or more control links per wheel. This is the biggest and most complex solution, ideal for high-performance automotive applications, but inadequate for most simple robotic systems.

The main advantages and disadvantages of each of the four suspension systems are displayed in Table 2.1. Considering the significance of maintaining a neutral camber angle and reducing the dimensions of the suspension system, the prismatic joint and Double Wishbone suspensions are the most adequate for a domestic mobile manipulator robot.

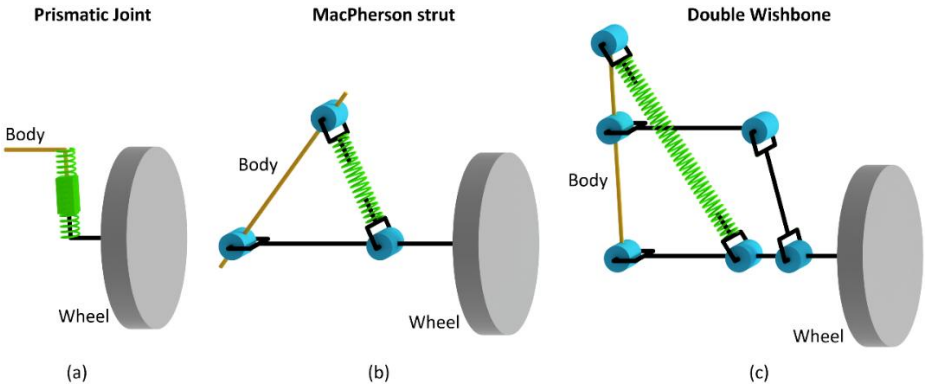


Figure 2.2 – Kinematics of the a) Prismatic Joint, b) MacPherson strut, and c) Double Wishbone suspensions.

Table 2.1 – Comparison of the different independent suspension types considering the intended application.

Suspension Type	Advantages	Drawbacks
Prismatic Joint	• Reduced Dimensions • Simplicity • Linear Wheel Motion	• Prismatic Joints
MacPherson strut	• Simplicity • Only Revolute Joints • Reduced Dimensions	• Variable Wheel Angle
Double Wishbone	• Only Revolute Joints • Configurable Wheel Angle	• Higher Complexity • Increased Dimensions
Multilink	• Configurable Wheel Angle	• Very High Complexity • Increased Dimensions

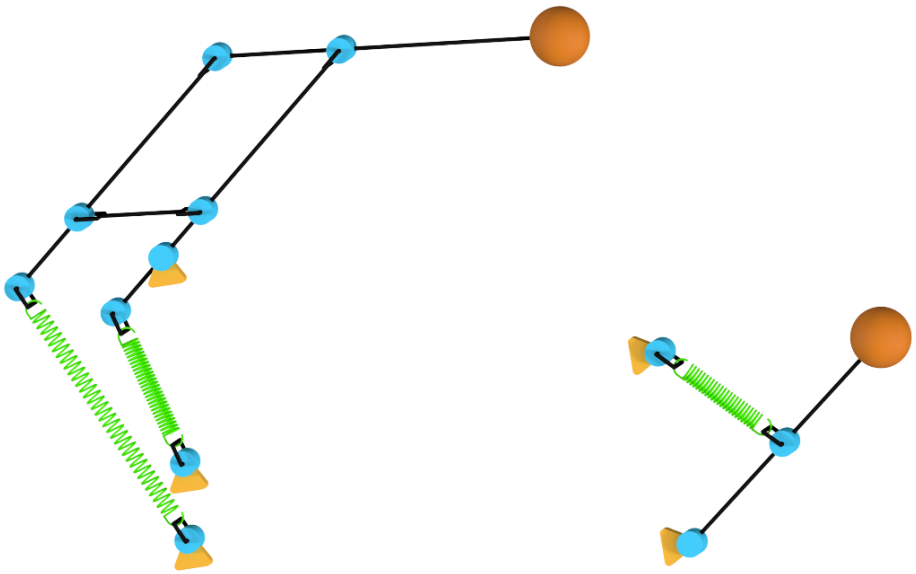
2.2.2. Static Balancing

One of the downsides of mobile robots is their finite energy supply (Canfield et al., 2019). In mechanical systems, energy consumption can be reduced by implementing energy-efficient designs that minimize the required actuation power. Statical balancing — sometimes referred to in the literature as gravity balancing (Nguyen, 2022b) or gravity compensation (Gosselin, 2008) — is a mechanism design methodology that aims at decreasing actuator loads by compensating the external forces applied in the system (J. Wang & Kong, 2019).

A mechanism is statically balanced when its potential energy is kept at a constant value throughout its complete range of motion (Hoetmer et al., 2010). Elements are added to the system to store and release potential energy, such as gravitational potential energy via the use of counterweights (Lacasse et al., 2013; Nakamoto & Matsuhira, 2017), elastic potential energy via the use of springs (Juang & Chen, 2022; Martini et al., 2015; Nguyen, 2022a; Vermeulen & Wisse, 2010; J. Wang & Kong, 2019) or compliant mechanisms (Blad et al., 2022; G. Chen & Zhang, 2011; Dunning et al., 2011; Kuppens et al., 2021; Tolou & Herder, 2009), or even potential magnetic energy via the use of magnetic elements (Boisclair et al., 2017).

The use of counterweights or tension springs are the two most common solutions used in robotics. When considering mobile robots, counterweights increase the overall mass and inertia of the system, which in turn results in higher energy consumption for the locomotion system (Gosselin, 2008).

Theoretical perfect static balancing solutions have been obtained for widely used practical applications using tension springs, such as for a single revolute DOF mechanism (the spring-and-lever mechanism Figure 2.3.b) and a 2 revolute DOF planar mechanism (the Anglepoise lamp Figure 2.3.a) (French & Widden, 2000).



(a) (b)

Figure 2.3 – a) Anglepoise lamp and b) Spring-And-Lever static balancing mechanisms

When considering most robotic applications, where 3D systems with long kinematic chains are used, the possible solutions for static balancing become larger (D. Lee & Seo, 2017), impractical, and costly. If the disadvantages of devising a perfect static balancing mechanism far outweigh the gains, then static balancing can be modeled as an optimization problem, in which the goal is the minimization of actuator loads without implementing mechanisms with excessive complexity and costs. Static balancing is also advantageous for improving the safety in human-robot interactions (Vermeulen & Wisse, 2010).

On mobile manipulator robots, the three main systems where static balancing can be implemented are: i) the redundant manipulator; ii) the robot's head, and iii) the hip mechanism. Redundant manipulators are the most used robotic systems, thus they have been the target of several static balancing studies such as (Boisclair et al., 2017; H.-S. Kim et al., 2022; Nguyen, 2023; Vermeulen & Wisse, 2010) for serial manipulators and (Baradat et al., 2008; Lixing Jin et al., 2022; Nguyen et al., 2020; Russo et al., 2005) for parallel manipulators. Mobile manipulator robots can include a head with sensors and features to boost human interaction (Fink, 2012). The head is often mobile, however, since its mass is reduced, the generated loads are also diminished. Thus, implementing static balancing solutions in these subsystems is often superfluous. The hip mechanism (sometimes also referred to as a torso mechanism, mobile trunk, or waist) is often introduced into human-inspired mobile manipulators to ensure that they can interact with objects on both the floor and high-to-reach planes. In the work by (Fuchs et al., 2009) they report that the motion of the torso represents the highest energy consumption of their robot due to the amount of mass to be moved. Most robotic hips are not standardized, each system

designs their own hip with one (Cheng & Ji, 2016; Negrete et al., 2018; Sturm et al., 2010), two (Agravante et al., 2013), three (Asfour et al., 2008) or more (Fiorio et al., 2017; Fuchs et al., 2009) degrees of freedom.

Since specific hip mechanisms are used for each robot, they are often not statically balanced and not the target for any generalizable static balancing study. However, some mechanisms developed towards this end include a gravity compensation system for varying masses of a three-DOF robotic waist (Yun et al., 2019) and a spring and pulley system for the gravity compensation of a three-DOF anthropomorphic robotic torso (Reinecke et al., 2020).

2.3. Multibody Dynamics in Robotics

Multibody dynamics plays a key role in the research of robotic systems. Multibody simulators are used to study the motion and trajectories of robots, to compare the efficiency of different control solutions in computational environments, and to perform the design optimization of the robot's morphology, structure, and mechanisms (Siciliano et al., 2009).

The methods for solving the dynamics of robotic systems can be grouped into two categories: a) to use commercial or open-source software to create and simulate a virtual model of the robot, or b) to implement the equations of motion describing a system's dynamics into an integrated development environment to simulate how the motion of the robot progresses over time. This literature review will focus on the manual implementation of the equations of motion, the method used for CHARMIE due to the higher computational efficiency, broader software compatibility, and for generating hardware-independent solutions that can be run on the robot's embedded computer.

The type of kinematic and dynamic studies depends on the desired application for the simulator. In the study of the CHARMIE robot, the goals are to determine actuator loads, to train neural network solutions for the trajectory control of the robot, and to perform the parametric optimization for the static balancing system. To achieve this end, the formulations for studying forward kinematics and inverse dynamics will be discussed. Furthermore, the algorithms that incorporate both studies into a systematic notation to achieve the multibody model of robotic systems will also be presented.

2.3.1. Forward Kinematics

The forward kinematics analysis is the process of obtaining the position and orientation of the bodies of the system, and how these properties vary over time, from known joint configurations.

A rigid body in 3D space can be defined by its orientation and the position of one of its points. The position of a point in Euclidian space can be fully described with three Cartesian coordinates: x, y, and z. However, defining three-dimensional rotations, denoted SO(3), is not straightforward. Below, the main formalisms used in robotics are presented and discussed.

Rotation matrices are [3 × 3] matrices used to uniquely define the rotation between two coordinate frames i and j. The rotation matrix $\mathbf{R}_j^i$ describes the rotation from frame i to frame j, which corresponds to the dot product of the basis vectors of the two coordinate frames (Waldron & Schmiedeler, 2008):

$$\mathbf{R}_j^i = \begin{bmatrix} \hat{\mathbf{x}}_i \cdot \hat{\mathbf{x}}_j & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{x}}_j & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{x}}_j \\ \hat{\mathbf{x}}_i \cdot \hat{\mathbf{y}}_j & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{y}}_j & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{y}}_j \\ \hat{\mathbf{x}}_i \cdot \hat{\mathbf{z}}_j & \hat{\mathbf{y}}_i \cdot \hat{\mathbf{z}}_j & \hat{\mathbf{z}}_i \cdot \hat{\mathbf{z}}_j \end{bmatrix} \tag{2.1}$$

If the Cartesian coordinates of a point P are placed into a [1 × 3] vector, and then this vector is multiplied by the $\mathbf{R}_j^i$ matrix, then point P is transformed following the rotation between the two considered frames. This process enables the conversion of coordinates between references with distinct orientations.

Rotation matrices can be combined via their multiplication; this property can be harnessed to represent any number of sequential rotations:

$$\mathbf{R}_k^i = \mathbf{R}_j^i \cdot \mathbf{R}_k^j \tag{2.2}$$

The simplicity and computational efficiency of manipulating rotation matrices motivates their use in several formalisms, such as to describe rotations obtained from Euler Angles (Aye, 2011).

In the field of robotics, one of the most widely used notations for the analysis of manipulator kinematics is the Denavit-Hartenberg convention. The transformations between consecutive bodies in the kinematic chains are fully characterized by four parameters: $[a_i, \alpha_i, d_i, \theta_i]$. The parameters a_i and α_i describe the length and twist of the link, while d_i and θ_i define the joint's offset and angle (Corke, 2007). Application of the Denhavit-Hartenberg convention implies attaching a set of reference axes to each link of the kinematic chain. Authors such as (Rocha et al., 2011) propose formal methodologies in the form

of a set of rules and steps to be followed to guarantee cohesion in the resulting Denhavit-Hartenberg parameters.

In mathematical operations using the Denavit-Hartenberg parameters, rotations, and translations are combined into a single calculation. The four parameters are used to construct [4 × 4] homogeneous transformation matrices:

$$\mathbf{H}_{j}^{i}=\begin{bmatrix}\cos\theta_i & -\sin\theta_i\sin\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i\\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i\\ 0 & \sin\alpha_i & \cos\alpha_i & d_i\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{2.3}$$

These matrices can be combined and manipulated equal to simple rotation matrices. The main advantages of this methodology are twofold: robot kinematics are expressed in a compact form, and a vast number of algorithms have already been developed to use the Denavit-Hartenberg parameters as inputs. Works using this notation can be found in (Ding & Liu, 2018; Klug et al., 2019; Thomas et al., 2020).

The three-dimensional orientation of an object can be represented as a sequence of three consecutive rotations. These rotations can either be extrinsic (concerning the original fixed and motionless coordinate axes) or intrinsic (concerning the rotation coordinate axes of the considered body). When the first and third rotations are around the same axis, then Euler Angles are being used, otherwise, if all three rotations are around different axes, Tait–Bryan angles are used. The rotation matrix originated from intrinsic ZXZ Euler rotations, defined by the three angles Z_1, X_2, and Z_3, is obtained from the formulation (Aye, 2011):

$$\mathrm{ZXZEuler}\,(\mathrm{Z}_1,\mathrm{X}_2,\mathrm{Z}_3)=\begin{bmatrix}c_1c_3-s_1c_2s_3 & -c_1s_3-s_1c_2c_3 & s_1s_3\\ s_1c_3+c_1c_2s_3 & c_1c_2c_3-s_1s_3 & -c_1s_2\\ s_2s_3 & s_2c_3 & c_2\end{bmatrix} \tag{2.4}$$

where c and s represent the cosine and sign functions respectively, and the indexes 1, 2, and 3 refer to each of the three consecutive rotations.

The use of Euler angles is associated with two limitations. The first is the Euler angle singularity: when the angle of the second rotations is either $\pi/2$ or $-\pi/2$, the first and third angles are independent variables controlling the same DOF, so they can assume an infinite combination of values. The second problem is gimbal lock: when two rotation axes are aligned, a degree of freedom is lost, preventing the system from immediately evolving to a determined state, which may result in infinite velocities. These

limitations can be mitigated and addressed (Hemingway & O'Reilly, 2018), or become severe problems, depending on the practical application of the system. Works implementing the Euler-Angle notation are available in (Perrusquía & Yu, 2020; Ran & Cheng, 2016; Schappler et al., 2019).

Quaternions are four-dimensional vectors represented by 4 scalar entities:

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} \tag{2.5}$$

where a, b, c, and d are real numbers, and $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are unit vectors along three spatial axes.

Quaternions can be harnessed to compute rotations on points and vectors. They are one of the major alternatives to rotation matrices due to their very high computational efficiency, ease of interpolation, and avoiding both limitations associated with the Euler Angles. For a deeper understanding of quaternions, the works by (Goldman, 2011; Hamilton, 2010; Kuipers, 2012) are recommended. Some works using quaternions include (Valverde & Tsiotras, 2018; Zupan et al., 2013).

Another method which can be implemented for the computational analysis of multi-body kinematics is screw theory. Two three-dimensional vectors can be used to represent either the position and orientation of a rigid body or its linear and angular velocities (Huang et al., 2013). The two vectors fully characterize a screw by defining the Plücker coordinates of a line in space — which in turn defines the position and direction of the screw axis — as well as the magnitude and pitch of the screw (Huang et al., 2013). Screw theory can be conjugated with Lie algebra to develop efficient recursive algorithms for solving forward kinematics of robotic systems (Müller, 2018). Some works that implement Screw Theory for kinematic applications can be found in (Q. Chen et al., 2015; Su et al., 2009; Wu et al., 2015).

2.3.2. Inverse Dynamics

Dynamics refers to the study of forces, movements, and the interaction between these two elements. When the goal is to determine the forces applied between a set of bodies based on their known kinematics and physical properties, the problem is called an inverse dynamics study (Otten, 2003). The aim is to define the Equations of Motion of a system, which describe the motion of each body as a function of time. In robotics, four main notations can be used to obtain the equations of motion: the Newton-Euler formulation (Ghaedrahmati et al., 2019), the Lagrange formulation (Emam, 2020), the Hamilton Formulation (Turno & Malczyk, 2023), and Kane's dynamical equations (Kane & Levinson, 1983). The first three formulations mentioned are interpretations of classical mechanics, while Kane's dynamical equations of motion are mathematical formulations derived specifically to tackle the computational

analysis of dynamic systems (Roithmayr & Hodges, 2016). All four formulations can be manipulated to originate algorithms discussed in section 2.3.3., but this subsection will focus on the base formulations.

The Newton-Euler formulation expands Newton's Second Law of Motion — which relates forces to acceleration, and therefore motion — to mathematically describe the combined translational and rotational dynamics of rigid bodies. This law is defined by the notation expressed in Equations 2.6—2.7.

$$\mathbf{F} = \frac{d}{dt}(m\mathbf{v}) \tag{2.6}$$

for linear motion, where m is the mass of a body, $\mathbf{v}$ its linear velocities, $\mathbf{F}$ the total forces applied on the body; and:

$$\mathbf{M}_{\mathrm{T}} = \frac{d\mathbf{L}}{dt} \tag{2.7}$$

for angular motion; $\mathbf{L}$ is the angular momentum of a body, and $\mathbf{M}_{\mathrm{T}}$ the sum of moments applied to it.

Works using the Newton-Euler formulation include studying robots with closed-loops, parallel robots, and mobile robots with elastic joints (Wisama Khalil, 2011), the study of open-loop robotic systems and manipulators (Aloulou & Boubaker, 2012; Giacaglia & de Queiroz Lamas, 2012), and the analysis of the serial manipulator of omnidirectional mobile manipulators (W. Jia et al., 2017).

The Lagrange formulation is centered around the study of the total energy of a system by analyzing the potential energy U and kinetic energy T. The basic Lagrange function (L) considers the following energetic balance (Menzel & Peshkin, 1955):

$$L = T - U \tag{2.8}$$

The physical properties of bodies can be associated with the desired kinematic values (acceleration, velocities, and positions) by solving the Lagrange equations of the second kind, known as the Euler-Lagrange equations (Menzel & Peshkin, 1955):

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{\mathbf{q}}_{\mathrm{k}}}\right) - \frac{\delta L}{\delta \mathbf{q}_{\mathrm{k}}} = \mathbf{F}_{\mathrm{k}} \tag{2.9}$$

where $\mathbf{q}_{\mathrm{k}}$ represents the generalized coordinates of a system, $\dot{\mathbf{q}}_{\mathrm{k}}$ the generalized velocities, and $\mathbf{F}_{\mathrm{k}}$ any additional nonconservative forces.

The Lagrange formulation has been used for studying either only the omnidirectional locomotion of a mobile manipulator robot (W. Jia et al., 2017) or the dynamics of the entire system including

manipulation (Xu et al., 2007), as well as to obtain generalized methods to simplify the study of multi-DOF manipulators (Dong et al., 2019).

The Hamiltonian formulation is a reframed version of the Lagrange formulation; therefore, it is based on the energy balance of the system. The distinctive feature of the Hamiltonian formulation is that generalized momenta $\mathbf{p}_k$ are used in place of the generalized velocities $\dot{\mathbf{q}}_k$. The conjugate momenta can be determined from Lagrange's equations with the following expression (Hand & Finch, 1998):

$$\mathbf{p}_k(\mathbf{q}_k, \dot{\mathbf{q}}_k) = \frac{\delta L}{\delta \dot{\mathbf{q}}_k} \tag{2.10}$$

The Equations of motion based on Hamiltonian mechanics are determined by the Equations 2.11–2.13 (Hand & Finch, 1998).

$$\frac{d\mathbf{q}_k}{dt} = \frac{\delta H}{\delta \mathbf{p}_k} \tag{2.11}$$

$$\frac{d\mathbf{p}_k}{dt} = -\frac{\delta H}{\delta \mathbf{q}_k} \tag{2.12}$$

$$\frac{dH}{dt} = -\frac{\delta L}{\delta t} \tag{2.13}$$

where H is the total energy of the system ($T + U$), L is the Lagrange function, and $\mathbf{q}_k$ and $\mathbf{p}_k$ the generalized coordinates and its conjugate momenta respectively.

In the work (Záda & Belda, 2017) Hamiltonian mechanics describe the dynamics of an industrial manipulator. The authors concluded that Hamiltonian mechanics are suitable for smaller control actions and more convenient for systems with a higher number of DOFs.

Kane's method of obtaining the equations of motion is a specific case of the Euler–Jourdain equations (Müller, 2018). Kane's method is a vector-based approach in which the mathematics are simplified by eliminating non-relevant forces and torques at the start of the analysis (Hussain & Azlan, 2016). The implementation of Kane's method can be described as a 7-step methodology that results in the Equations of Motion in a matrix form:

$$\mathbf{M}\ddot{\mathbf{Q}} = \mathbf{T} + \mathbf{V} + \mathbf{G} + \mathbf{E} \tag{2.14}$$

where $\mathbf{M}$ is the mass matrix, $\ddot{\mathbf{Q}}$ is the acceleration vector, $\mathbf{T}$ is the vector of applied torques, $\mathbf{V}$ is the vector containing the moments from the centrifugal forces, $\mathbf{G}$ is the vector containing the moments from the gravitational forces, and $\mathbf{E}$ is the vector containing the moments from external forces and torques.

2.3.3. Algorithms for Generating Multibody Models

In the field of robotics, recursive Newton-Euler algorithms (Luh et al., 1980) are commonly used due to their computational efficiency, low complexity, and the fact their mathematical complexity increases linearly with the number of bodies of the system (Müller, 2018). Recursive Newton-Euler algorithms are divided into two stages: first, the forward kinematics of the bodies are determined by analyzing the kinematic chain from ground reference to end-effector; then, the inverse dynamic analysis determines the forces and torques applied by each pair of bodies by analyzing the end-effectors first and then progressing towards the ground reference. Some of the most efficient Newton-Euler algorithms (W. Khalil et al., 1986) can match the computational efficiency of Kane's equations (Angeles et al., 1989). The work by (Müller, 2018) derives Newton-Euler recursive algorithms which employ screw theory and Lie algebra to achieve high computational efficiency.

The work by (Hollerbach, 1980) presents a recursive Lagrangian formulation for manipulator analysis but reports that the obtained recursive algorithm has lower efficiency when compared to recursive Newton-Euler algorithms. The Lagrange Equations of motions are often augmented and used alongside Divide-and-Conquer Algorithms (DCA) presented by (Featherstone, 1999), which are recursive by nature, to be able to solve both open systems, as well as closed-loop constrained systems (Poursina & Anderson, 2013).

Hamilton's formulation is employed in (Krzysztof Chadaj et al., 2017) to develop a parallel HDCA (Hamilton Divide-and-Conquer Algorithm) which is advantageous for the study of serial kinematic chains. The work by (Turno & Malczyk, 2023) focuses on the parallel computing capabilities of HDCA algorithms, reporting that they may lead to substantial improvements in performance in specific cases.

Kane's Equations of Motion are commonly used in the field of robotics to obtain algorithms, and they are more efficient than using the Lagrange formulation and can be as efficient as the Newton-Euler algorithms (Angeles et al., 1989). The work by (Singh et al., 1985) presents a formulation for the Equations of Motion of tree-like kinematic chains with an emphasis on the computational efficiency of the obtained solution. The work (Rosenthal & Sherman, 1986) combines Kane's Equation with symbolic equation manipulation techniques to develop a high-performance computer simulator program with an emphasis on the user interface. The work (Rone & Ben-Tzvi, 2014) employs Kane's method to solve the equations of motion of a continuum robot, which has a theoretically infinite number of joints and degrees of freedom.

2.4. Neural Networks

Artificial Neural Network (ANN) are connectionist systems (see Figure 2.4) — composed of nodes (neurons) and their connections (synapses)— which are inspired by biological neural networks (Y.-Y. Chen et al., 2019). The weights associated with each synapse can be shifted to change the behavior of the neural network. Neural networks can assume different configurations and architectures such as Feedforward Neural Networks, Recurrent Neural Networks, Spiking Neural Networks, etc. (S. Li et al., 2017). Neurons in the neural network belong to three different types of layers: input layers, where each neuron receives an input value; hidden layers, where any number of neurons can be placed and interconnected to transmit and process information; and output layers, where each neuron will output a value. After implementation, neural networks can be regarded as a black box, where relations between inputs and outputs are established based on the methodology used and the desired behaviours.

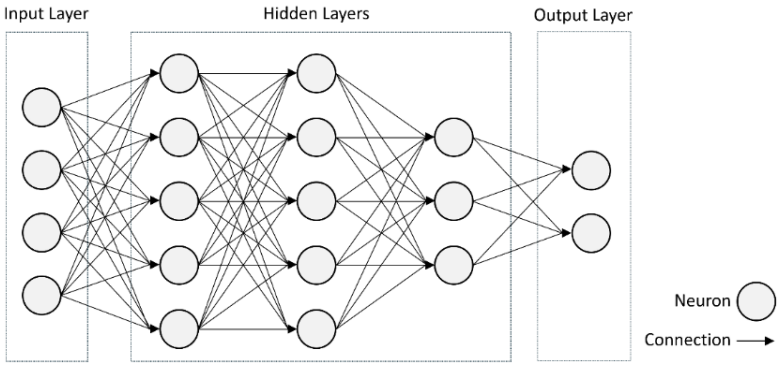


Figure 2.4 – Architecture of a Fully Connected Deep Feed Forward Neural Network

Neural Networks are deeply tied to Machine Learning (ML) algorithms. ML is a research field in which statistic-based algorithms analyze patterns and trends in data to attain optimization without explicit programming, resulting in computer systems that improve their performance through experience (Jordan & Mitchell, 2015). The three main branches of ML are: i) Supervised Learning, the study of labeled data to find patterns and predict outcomes; ii) Unsupervised Learning, the study of unlabelled data to identify patterns, and clusters and to analyze big data; and iii) Reinforcement Learning (RL), in which strategies are defined to train a computational system so that its behavior converges towards the desired performance. In the field of robotics, RL is providing the most significant contributions towards the development of autonomous systems; these developments were greatly enhanced by the employment of deep neural networks, which originated Deep Reinforcement Learning (DRL) systems (Arulkumaran et al., 2017).

The crux of RL is learning through experience and interaction (Arulkumaran et al., 2017). An agent, that exists in a determined state, interacts with its surroundings by taking a certain action. A reward function is then used to judge the adequacy and effectiveness of the used action based on the newly achieved state: if the action produced a positive outcome, the system is rewarded, if the action produced a negative or detrimental outcome, the system is punished. The system then adapts based on the reward function, mimicking the learning processes of behaviorist psychology. DRL is an expansion of RL, where ANN with multiple layers of neurons (an input layer, several hidden layers, and an output layer as shown in Figure 2.4) provide the basis architecture for the learning machine. Deep learning provides significant performance improvements when compared to traditional ML processes (Sarker, 2021). Some examples of DRL solutions include Multilayer Perceptron, Long Short-Term Memory Recurrent Neural Network, and Convolutional Neural Network.

DRL has already achieved previously impossible tasks in the field of computation, such as defeating humans in the game of Go (Silver et al., 2017) and Grandmaster-level performance in the videogame Starcraft II (Vinyals et al., 2019). In the field of robotics, DRL has achieved success in several tasks such as object movement, object manipulation, and locomotion control (Lillicrap et al., 2015). Model-based DRL methods, which often employ simulators for the training and learning process of the neural network, achieve much higher sample efficiency and flexibility when compared to model-free DRL (Nagabandi et al., 2018). The work by (Chai et al., 2024) employs DRL to achieve high-efficiency trajectory planning for a mobile robot navigating through unknown environments. With a complete computational model of a robotic system, DRL algorithms can be developed, implemented, trained, and tested in the safety of computer simulations, before being transferred into the physical prototypes. Of all available DRL approaches, Proximal Policy Optimization reports the highest speed of convergence with good stability (Zhou et al., 2022).

2.5. Summary and Discussion

In this Literature review, a set of trends in the development, analysis, and control of mobile manipulator robots were identified and described.

The study of Mobile Manipulator Robot Design and Mechanism synthesis grants a foundation for the morphology of the CHARMIE robot. CHARMIE will follow a human-inspired shape due to its intended human interaction and operation in human-engineered environments but employ 4 Swedish Wheel which is optimal for indoor navigation. The robot's range of motion will be expanded by employing an articulated torso. The energy consumption of this torso will be minimized via the application of static balancing mechanisms. Six DOF arms will be implemented for object manipulation, guaranteeing human-like mobility without significantly increasing the complexity of the robot.

The development of the robot will be highly supported by the creation of a multibody model and the corresponding simulation environment. The compatibility and efficiency of the multibody simulator will be maximized by creating an in-house code that models the robot's fundamental dynamics and implements them into an Integrated Development Environment (IDE). For the present research, this model must include the study of the Forward Kinematics and Inverse Dynamics of the robot. Following the trends in robotics, Recursive Newton-Euler algorithms will be implemented due to their high efficiency and simplicity. To split the rotation and translation operations of the kinematic analysis, avoiding the decrease in computational efficiency caused by using Denavit–Hartenberg parameters, the kinematic analysis is centered around the use of cartesian coordinates and rotations defined by Euler Angles.

Neural Network-based control solutions provide more versatile systems that do not require explicit programming, becoming more adaptable at dealing with unstructured work environments. Neural Networks are providing significant breakthroughs in the field of robotics, specifically via the use of Deep Reinforcement Learning (DRL). Significant improvements can be made to the learning process of the DRL Neural Networks by employing model-based learning. By implementing a Proximal Policy Optimization control solution to CHARMIE, the robot's efficiency will improve, and important scientific contributions will be made to this relevant field of research.

"There is nothing impossible to him who will try."
Alexander the Great

3. DESIGN AND CONSTRUCTION OF THE PROPOSED CONCEPT

One of the most solid proofs of concept for the development of a new robot is the construction of a fully functional prototype. The advantages of achieving a prototype are manyfold: it provides a practical validation of the theoretical concepts, it permits the verification of the structure and dynamics of the robot, and it allows a platform to test and confirm control solutions in actual physical unstructured environments.

The prototype was attained by following four sequential steps: i) determining the specifications of the robot; ii) utilizing those specifications and information from the literature to obtain a preliminary design of the robot by resorting to CAD software; iii) optimizing the mechanism designs of the robot with the assistance of the multibody model described in Chapter 4; and iv) creating the technical drawings of the prototype's components, manufacturing them and assembling the final prototype.

3.1. Robot Specifications

The specifications for CHARMIE's prototype are tied to three main requirements: the purpose of the robot (performing domestic tasks), the work environment of the robot (unstructured indoors environments), and the goal of maintaining the robot's manufacturing low-cost and replicable.

Starting with the overall requirements for the robot, human-like shape, proportions and ranges of motion will be adopted to enhance the quality of human interaction (Fink, 2012). In Human-Robot Interaction, safety is always a main concern. Accidents can be avoided by always guaranteeing the static stability of the locomotive system and the robot's structure. Uncontrolled joints with indeterminate behaviors must also be avoided. The dimensions of the robot are determined by its work environment. The base diameter must not surpass 650 mm (so that the robot can traverse through door frames), the height must not exceed 1800 mm to avoid collision and not be overly intimidating, and the robot's total mass should not exceed 120 kg. For safety reasons, the protection of hardware, and to enhance interaction, the main mechanical and electrical components should not be visible to users. This prototype

will not only be a step towards producing a finished product but also a contribution to the scientific community; the use of replicable and reproducible manufacturing technologies and design choices must be ensured in the robot's project.

When analyzing the three main options for mobile manipulator robot locomotion, legged locomotion breaks the static stability requirements, while tracked wheels cause damage to floors and mats in indoor environments, therefore, wheeled locomotion will be the only viable option (Bruzzone & Quaglia, 2012). With the confined navigation conditions of domestic spaces, omnidirectional locomotion is heavily favored, using four wheels to enhance the robot's stability. The performance of four omnidirectional wheel solutions is greatly enhanced by employing a suspension system. The dimensional limitations of the robot's base favor the use of independent suspensions. The use of omnidirectional wheels requires a constant neutral camber angle for stable and optimal locomotion.

When inquired regarding their preferences for domestic robots, the elderly expressed a high desire for the robot to be able to grab objects from the floor (Beer et al., 2012). Interaction with objects on both the floor and high to reach places while mimicking human motion and anatomy requires an articulated torso. Two DOF will be added to the torso, one prismatic and one rotational, to mimic the human's squatting and bending forward motions. Torso mechanisms have high energy consumption (Fuchs et al., 2009), which is harmful to the efficiency of mobile robots. This limitation can be mitigated by employing self-locking mechanisms for the torso mechanisms actuation and a static balancing solution to reduce the power required to move the torso.

The robot's manipulation system aims to find the optimal compromise between simplicity, efficiency, human likeness, and range of motion. Two 6-DOF arms will be employed for manipulation. The arms must avoid continuous 360º rotations (which human arms appear to achieve) to avoid the entanglement of wires and decrease the system's complexity. The robot's main goal is to perform manipulation tasks; to guarantee its applicability in a range of different item transportation tasks, the arms must support a maximum payload of up to 1 kg each.

Copying human nature, therefore allowing for more natural interaction, the robot's main sound and image inputs will be placed on its head. Two degrees of freedom will be granted to the robot's head to both allow communication via head movement, as well as to increase the range of the head sensors without requiring full locomotion.

All established requirements for the design of CHARMIE have been listed in Table 3.1.

Table 3.1 – Requirements established for the design of the CHARMIE robot.

Number	Requirement
a)	Overall Requirements
a.1)	Human-like morphology
a.2)	Human-like proportions
a.3)	Human-like range of motion
a.4)	Static stability must be always guaranteed
a.5)	No uncontrolled joints
a.6)	Base diameter < 650 mm
a.7)	Total height < 1800 mm
a.8)	Total mass < 120 kg
a.9)	Mechanical/electrical components must not be visible
a.10)	Fully replicable design
b)	Locomotion and Suspension
b.1)	Wheeled locomotion
b.2)	Omnidirectional locomotion
b.3)	Four-wheeled locomotion
b.4)	Independent suspension system
b.5)	Constant neutral camber angle
c)	Torso
c.1)	2-DOF mobility
c.2)	Self-locking actuation
c.3)	Static balancing of both DOF
d)	Arms
d.1)	6-DOF arms
d.2)	No continuous rotation
d.3)	1kg payload carrying capacity
e)	Head
e.1)	2-DOF neck

3.2. Design of the CHARMIE Robot

The CHARMIE robot was designed as a modular robot with interconnected subsystems. The advantages of this philosophy are threefold: each system can be studied and optimized individually, accelerating research and fabrication; the solutions implemented into each subsystem can be adapted and partially applied to other robots; and since prototyping and product development in an interactive

and continuous process, modular approaches allow new versions of subsystems to be replaced without compromising the robot. The preliminary CAD model of the CHARMIE robot is shown in Figure 3.1.



Figure 3.1 – CAD model of CHARMIE's preliminary design

An octagonal steel sheet base was used as the initial structure, and then the robot's remaining systems were developed to expand upon this base. Reflecting the modular system development, each of the following five subsections will tackle a subsystem of CHARIE: i) the locomotion and suspension systems; ii) the squatting motion mechanism; iii) the trunk flexion motion mechanism; iv) the development of the manipulator arms; v) and the study of the robot's head.

3.2.1. Locomotive and Suspension System

Considering the established requisites, there are two possible wheel choices for CHARMIE: Swedish 90° wheels or Mecanum wheels. Swedish 90° wheels were used since their manufacturing is simpler, their costs are lower and the main advantage of Mecanum wheels (higher carrying capacity and traction forces for a preferential direction) will not be relevant for CHARMIE's indoor tasks with limited payloads (Massoud et al., 2021). The four wheels are distributed in an axisymmetric disposition, with their corresponding rotation axis displaced angularly by 45° about the front of the robot. In this

configuration, all wheels are engaged in active motion when the robot moves forward, providing higher stability.

Swedish 90º wheels have two main motions, an active motion and a passive motion. Passive motion does not have any actuation, it is permitted by placing roller components along the wheel's outer surface. Adequate active motion is ensured by associating an independent motor to the axle of each wheel. The robot was simplified by implementing the same support system for each wheel comprised of four main components: the Swedish 90º wheel, the actuating DC motor, a bearing to reduce radial loads on the motors, and a structural component connected to both the DC motor and the bearing (see Figure 3.2).

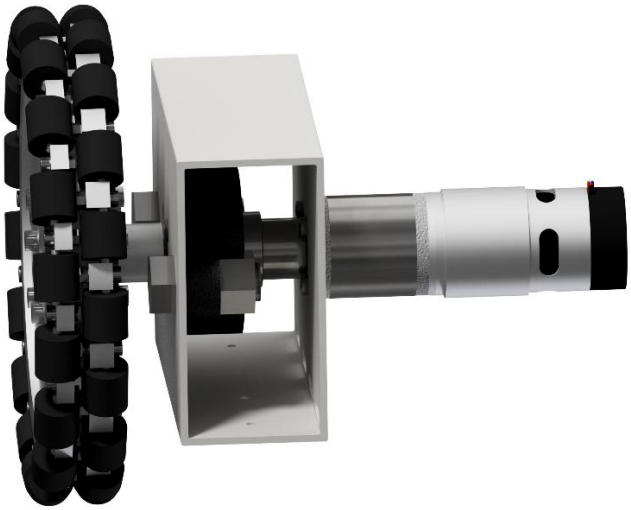


Figure 3.2 – CAD model of the wheel-motor-bearing assembly

The locomotion system requires each of these four units to connect to the robot's base via an independent suspension which does not allow angular motion for a constant camber angle. Space limitation is one of the main concerns for the CHARMIE robot. Employing a simple vertical prismatic joint is the most compact solution. This option has two main drawbacks, the first is requiring the production of prismatic joints, which are more costly and require tighter dimensional tolerances, and the second is the fragility of the system to loads orthogonal to the prismatic suspension motion. The first drawback was accepted as a necessity for the compactness of the robot, while the second is mitigated by the robot's low locomotion speeds, which will not originate abrupt impacts on the scale that would damage the suspension. The system's stability and ease of assembly were increased by using two parallel springs for each independent wheel (see Figure 3.3).

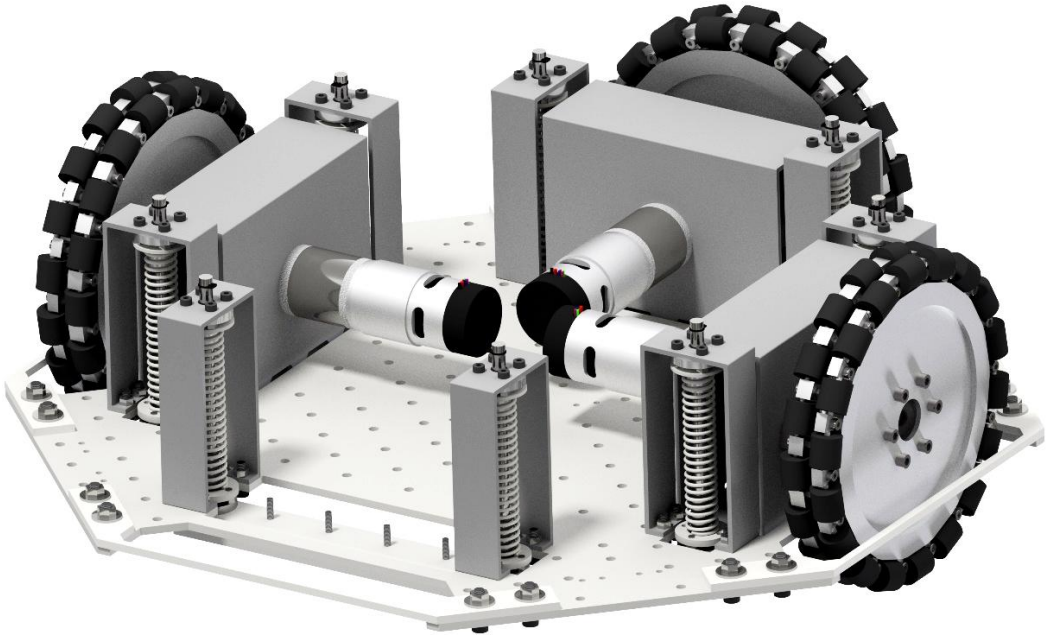


Figure 3.3 – CAD model of the suspension system assembly. One of the wheel-motor-bearing groupings is unassembled for better visibility.

The last consideration required to complete the dimensioning of the suspension system is the spring choice. Compression springs will be used. Five main dimensional limits must be respected: The coil's outer diameter D_{E} must not surpass 36 mm to avoid collision with the robot's base; the coil's inner diameter D_{I} must be higher than 12 mm to avoid collision with the suspension rod; the springs' maximum uncompressed extension L_0 must be greater than the suspension's maximum extension $y_{\max}$ defined by a locking washer on the suspension rod; the springs' minimum extension L_{S} must be smaller than the suspension's minimum extension $y_{\min}$ limited by the collision of the structural support of the motor system with the robot's base; and finally, under normal balance conditions, when the suspension is compressed Δy_0, the spring length must be contained within the minimum and maximum extension. These conditions are summed up by Equations 3.1–3.5.

$$D_{\mathrm{E}} < 0.036 \tag{3.1}$$

$$D_{\mathrm{I}} > 0.012 \tag{3.2}$$

$$L_0 > y_{\max} \tag{3.3}$$

$$L_{\mathrm{S}} < y_{\min} \tag{3.4}$$

$$y_{\min} < L_0 - \Delta y_0 < y_{\max} \tag{3.5}$$

When considering that eight springs in parallel support the robot's entire mass, the spring deformation for each spring under balanced conditions is given by the expression (Budynas & Nisbett, 2011):

$$\Delta y_0 = \frac{m_{\text{total}} g}{8k} \tag{3.6}$$

where m_{total} is the robot's total mass, g the acceleration of gravity, and k the spring constant.

After a spring is selected, the maximum $F_{\max}$ and minimum $F_{\min}$ force each suspension arm can exert is given by the pair of expressions:

$$F_{\min} = (L_0 - y_{\max})k \tag{3.7}$$

$$F_{\max} = (L_0 - y_{\min})k \tag{3.8}$$

A higher spring constant results in a wider range of forces applied, but a stiffer suspension with less mitigation of vibration and the effect of floor irregularities, while lower spring constants result in softer suspension with a narrower range. The goal is to find the ideal compromise of characteristics.

The compression springs were chosen based on the robot's current total mass of 100 kg. The properties of the selected springs are shown in Table 3.2. In the current configuration, these springs will be able to support up to 130 kg of robot mass.

Table 3.2 – Properties of the compression springs used for CHARMIE's suspension system

Property	**Value**
L_0	160 mm
D	23.2 mm
d	2.8 mm
N_{A}	20
k	2225 N/m
Material	AISI 302 Stainless Steel

3.2.2. Squatting Motion Mechanism

The goal of the squatting motion mechanism is to start expanding on the robot's base, adding some verticality to the robot to mimic human anatomy, while including a vertical DOF to maximize the system's workspace.

The development of the squatting mechanism was initiated by adding the fundamental degree of freedom, steel columns were added to the center of the base to support a prismatic joint (see Figure 3.4). The prismatic joint guides the vertical displacement of a stainless-steel rod, the first component of the robot's hip.

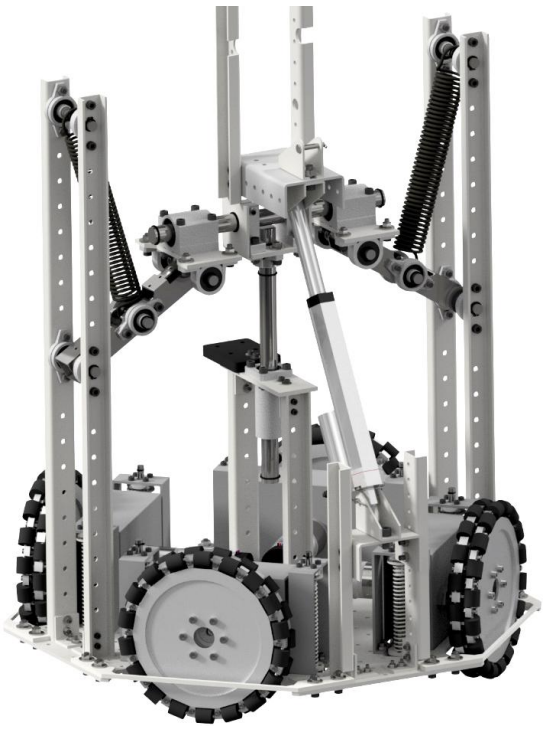


Figure 3.4 – CAD model of the robot's hip with the squatting motion mechanism

The movement of this vertical degree of freedom must be compensated by static balancing. The use of constant force springs was avoided by employing a static balancing solution similar to the spring-and-lever mechanism (described in Section 2.2.2). Two linear bearings were added in a horizontal configuration, which will solely support the vertical loads of the robot's upper body. By placing these bearings at the end of a lever, a single rotating lever with a vertical constant load applied to its end is obtained on each side of the robot, just like in the spring-and-lever mechanism. The final step is adding a tension spring to exert the counterbalancing force (see Figure 3.5). The optimization of this mechanism's geometry and the correct choice of springs is tackled in Section 3.3.1.

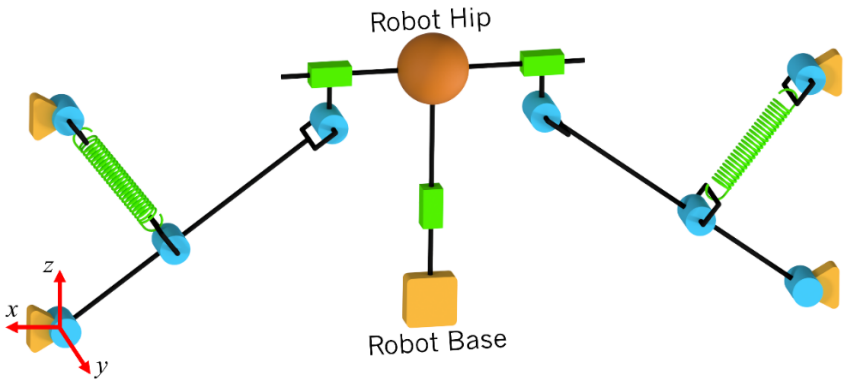


Figure 3.5 – Schematic representation of the robot's squatting motion static balancing mechanism

The squatting motion mechanism is concluded by adding a linear actuator (which is self-locking) between the robot's base and its hip. For maximum efficiency, this actuator is placed as vertically as possible (while avoiding collision with other components). The most optimal solution was placing this actuator at the front of the base, as shown in Figure 3.4.

3.2.3. Trunk Flexion Motion Mechanism

The trunk flexion motion corresponds to a single rotating DOF between the robot's hip and torso. The static balancing of this system is a direct application of the spring-and-lever mechanism achieved by placing tension springs between the robot's hip and torso. To increase the actuator efficiency, the linear actuator responsible for the trunk flexion is placed in the front of the robot connecting the hip to the robot's chest. (see Figure 3.6). The selection of springs and optimization of the mechanism's geometry is further described in Section 3.3.1.



Figure 3.6 – CAD model of the robot's trunk flexion motion mechanism

3.2.4. Manipulator Arms

CHARMIE's arms were designed as a 6-DOF serial manipulator, with a claw-like end-effector with an additional DOF for object interaction. Three main concerns had to be addressed when designing the arms: i) mechanical resistance to carry 1 kg payloads, ii) placement of the motors to maintain an anthropomorphic appearance, and iii) smoothness of motion. The selection of actuators, rod sizes, and the use of bearings and fixed rod supports were key in developing the arm. The manipulator arm can be divided into four smaller mechanism types: the shoulder joint, the mechanical claw, the arm z-axis rotations, and the arm x-axis rotations.

The shoulder joint (Figure 3.7.a) is responsible for the shoulder abduction\adduction and flexion\extension motions. This is achieved by the direct actuation of two DC motors on a cylindrical shaft. The shaft is supported by revolute bearings on the preceding body of the kinematic chain, and connected by shaft supports (which rotate synchronized with the shaft) to the succeeding body. The two axles of rotation are not aligned with each other due to the geometrical constraints and limitations of the robot's shoulder.

The robot's mechanical claw (Figure 3.7.b) includes some polymer bevel gears so that the actuating DC motor can be placed inside the claw's body. Polymer spur gears then transmit this motion so that a single motor is responsible for opening and closing the two halves of the claw in a synchronized way.

The rotations around the z-axis of the arm (which is aligned with the arm's length) are achieved by dividing the arm's structure into two sections (Figure 3.7.c). A DC motor is placed inside the preceding body, which will be responsible for controlling the rotation of the succeeding body.

The rotations around the x-axis of the arm (Figure 3.7.d) required the use of metal bevel gears. These gears are responsible for allowing the DC motors to be hidden inside the arm's structure, to avoid them from protruding outside of the structure. Due to geometrical limitations, the two consecutive bodies in these joints must be misaligned.

The four subsystems were used to obtain the arm's desired DOF (see Figure 3.8).

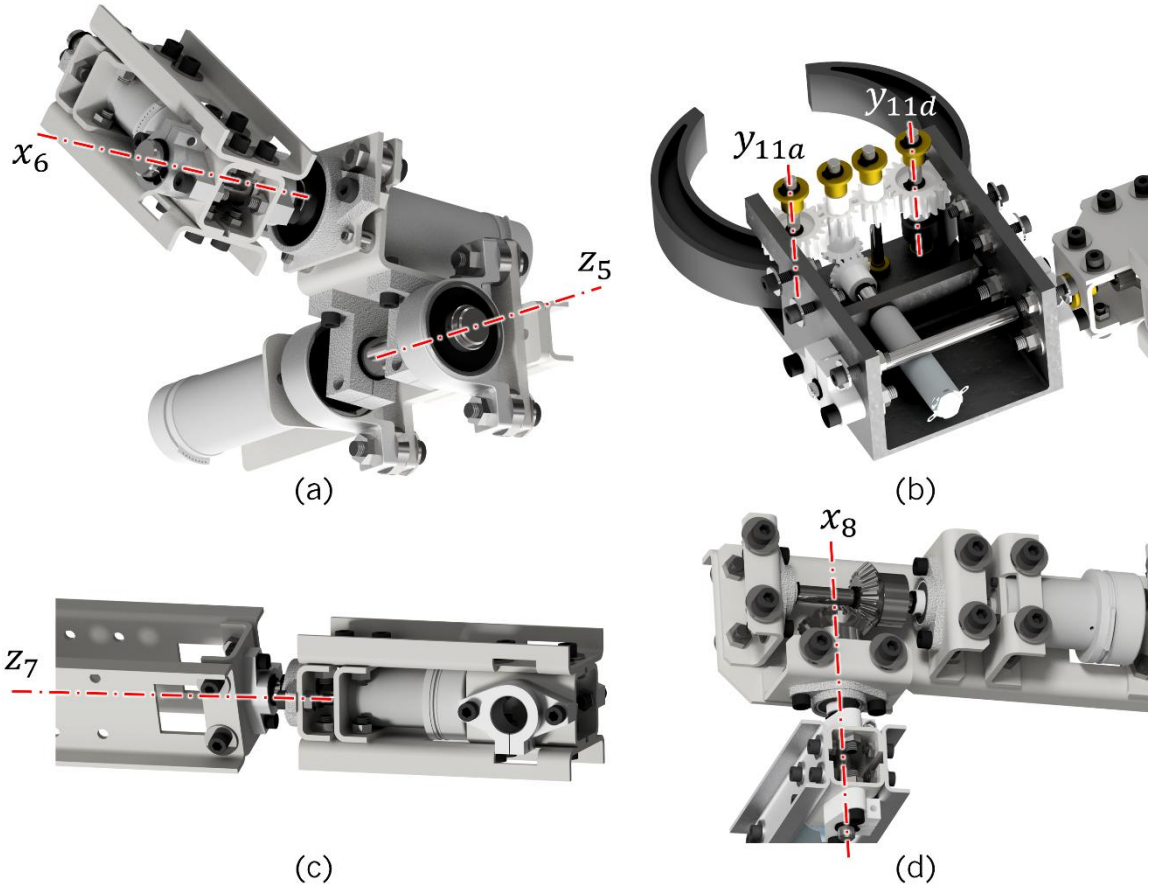


Figure 3.7 – CAD model of the subsystems of the manipulator arm with the output degrees of freedom highlighted: a) shoulder joint; b) mechanical claw; c) arm z-rotation; and d) arm x-rotation.

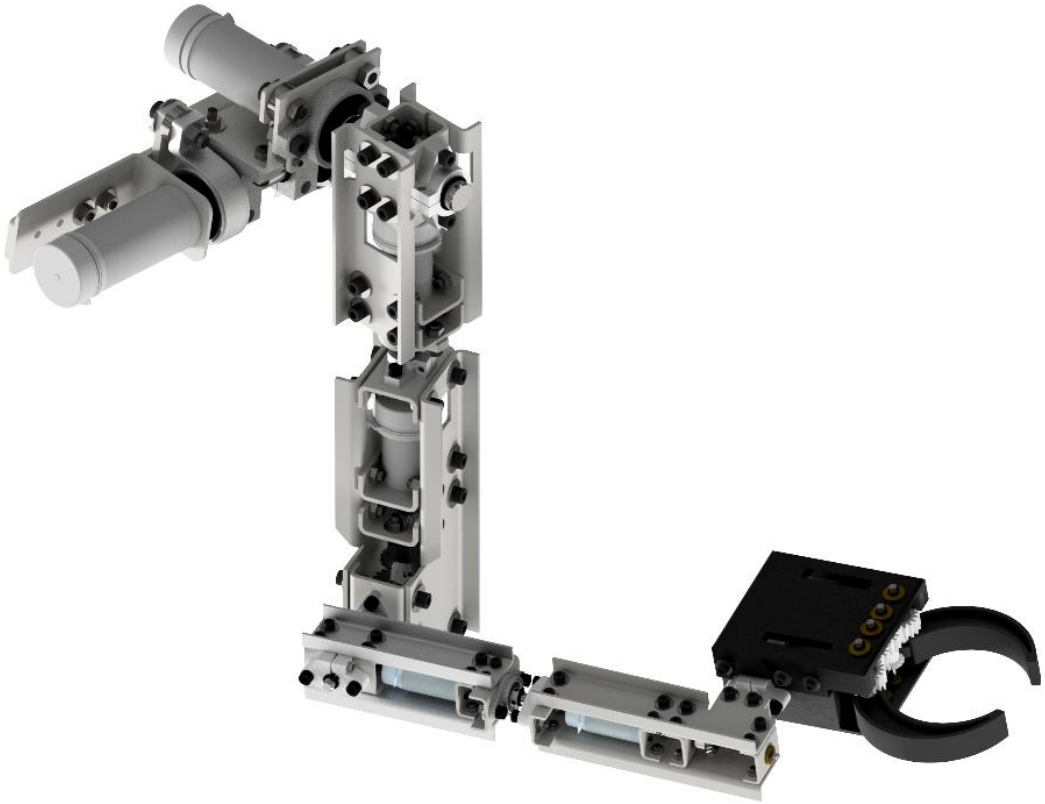


Figure 3.8 – CAD model of the robot's manipulator arms

3.2.5. Head of The Robot

The main structure of the robot's head is a set of square profile aluminum tubes in a parallelepipedal configuration with a mesh of holes for including any additional components (see Figure 3.9.a). The neck of the robot contains two degrees of freedom, the first one is responsible for the head's rotation, and the second one for the neck flexion and extension motions (see Figure 3.9.b).

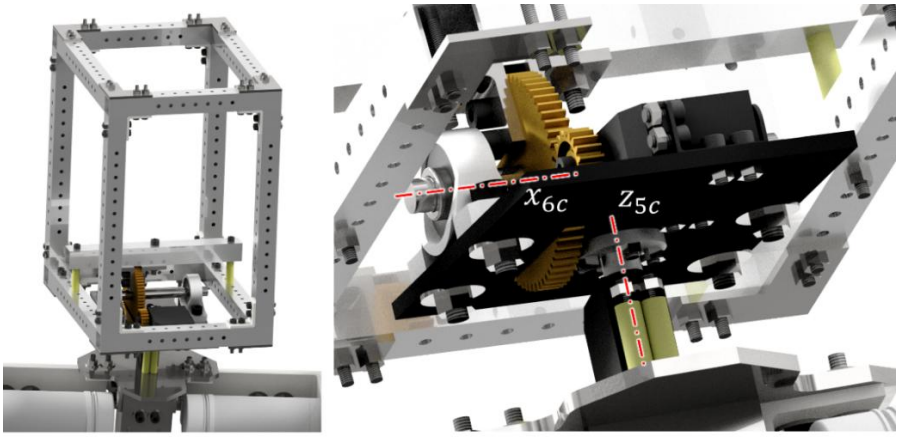


Figure 3.9 – a) CAD model of the robot's head and b) detail highlighting the neck's DOF

3.3. Mechanism Optimization

After the initial design, two of CHARMIE's subsystems required a more in-depth analysis using the multibody dynamics model developed in section 4. The design of these physical models and the multibody model was iterative and interactive. The current section refers to the mathematical formulation of the analysis conducted on the static balancing of the squatting and trunk flexion motions, as well as the structural analysis of the robot's arms.

3.3.1. Static Balancing of the Squatting and Trunk Flexion Motions

The optimization of the static balancing mechanism used a parametric optimization approach centered around the dynamic behavior of the system. The parametric approach requires three main preparatory stages: defining the parameters that will be varied; creating evaluation scenarios to study the solutions; and defining the objective function to evaluate each solution.

Two mechanisms, with minimal influence between each other, are being considered: the mechanism for the squatting motion, and the mechanism for the trunk flexion motion. These mechanisms mimic the action of the spring-and-lever static balancing scheme.

The geometry of a spring-and-lever mechanism is determined by three measurements: the distance between the revolute joints connecting both the spring and the lever to the fixed body; the distance between the two revolute joints in the lever connecting to the fixed body and the spring respectively; and the distance between the lever's revolute joint and the mass to be balanced. Since the position of the mass is defined by the robot's geometry, only two parameters can be changed. Furthermore, in the squatting mechanism, dimensional limitations restrict the position of the revolute joint in the middle of the lever. Therefore, the relevant dimensional parameters that can be studied are a_i, c_j, and d_j shown Figure 3.10.

The other two relevant parameters are the choice of extension springs, associated with the variables b_i and e_j.

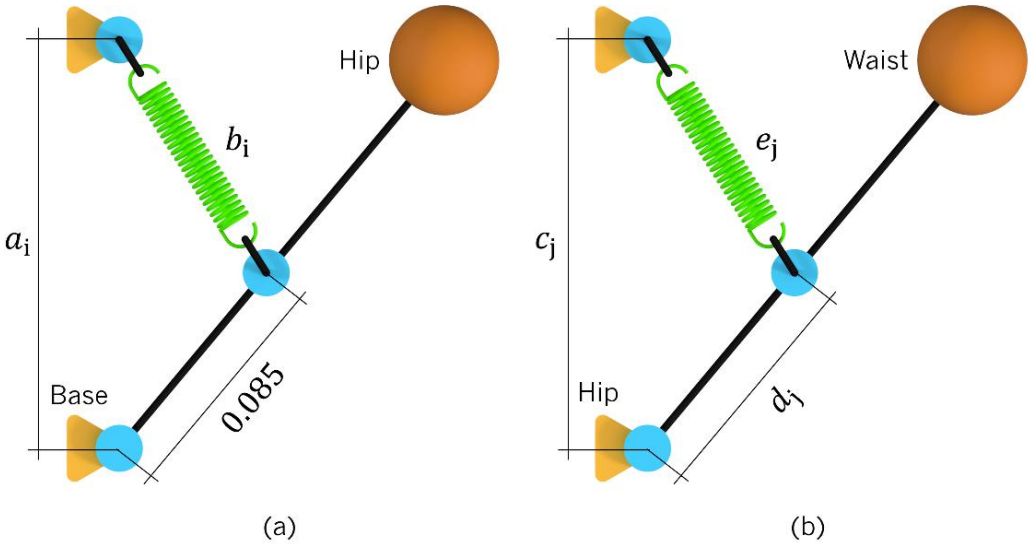


Figure 3.10 – Parameters for the optimization of the static balancing mechanisms of the CHARMIE robot for the a) squatting mechanism and b) trunk flexion mechanism

The result is a total of 5 parameters, 2 for the squatting mechanism, and 3 for the trunk flexion mechanism. The next step is the discretization of these parameters. The springs were chosen from a catalog of off-the-shelf tension springs (Annex I). Four additional springs are included for the trunk flexion motion since the ideal solution was the use of more than one of the springs in a parallel configuration. The maximum and minimum limits for the dimensional parameters were based on avoiding component collision, as well as the desired scale for the mechanism's dimension. After maximum and minimum dimensions were defined, a step of 20 mm was used to discretize the possible values. The possible combinations of input variables for the parametric optimization are shown in Table 3.3.

Table 3.3 – Input variables for the parametric optimization of the static balancing mechanisms

Parameter	Min	Step	Max	Possible Values
a_{i}	130	20	330	11
b_{i}	0	1	329	330
c_{j}	130	20	330	11
d_{j}	70	20	270	11
e_{j}	0	1	333	334

The evaluation scenarios to test each of the solutions employ the multibody dynamics simulation. The goal is to study the robot during non-specific general motions. If a single scenario is used to study the robot, the obtained solution may not be optimal during general motion, but rather an overfitted solution to a specific motion. For this end, a way to create realistic and constrained, yet random, trajectories was devised.

CHARMIE's trajectory will be controlled by a neural network which creates a sign variable s_{k} (with possible values -1, 0, or 1) for each joint k. These variables are then multiplied by a reference acceleration $\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}$, defining the acceleration of the joint for the given timestep. To obtain random trajectories, these s_{k} sign variables are randomized. To obtain realistic trajectories, constraints are added to the variables post-randomization.

Due to the physical limitations of the motors, each joint is limited by a maximum velocity $\dot{\vartheta}\mathrm{max}_{\mathrm{k}}$. This condition can be mathematically imposed on the trajectory generation by the following formulation:

$$\left|\dot{\vartheta}_{\mathrm{k}}(t)+s_{\mathrm{k}}(t)\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}\Delta t\right|>\dot{\vartheta}\mathrm{max}_{\mathrm{k}}\Rightarrow s_{\mathrm{k}}(t)=0 \tag{3.9}$$

where $\dot{\vartheta}_{\mathrm{k}}$ is the current velocity of joint k, t the current time frame, and Δt the simulation timestep.

For the position constraints, two limitations must be addressed: the system cannot enter a position that exceeds the maximum $\vartheta\mathrm{max}_{\mathrm{k}}$\ minimum $\vartheta\mathrm{min}_{\mathrm{k}}$ limits and the system cannot enter a state where its breaking distance is insufficient to avoid entering a limit-breaking situation. Both conditions can be imposed by Equations 3.10—3.11.

$$\vartheta\mathrm{max}_{\mathrm{k}}-\left[\vartheta_{\mathrm{k}}(t)+\dot{\vartheta}_{\mathrm{k}}(t)\Delta t+s_{\mathrm{k}}(t)\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}(\Delta t)^2\right]<\left|\frac{\left[\dot{\vartheta}_{\mathrm{k}}(t+1)\right]^2}{2\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}}\right|\Rightarrow s_{\mathrm{k}}(t)=-1 \tag{3.10}$$

$$\left[\vartheta_{\mathrm{k}}(t)+\dot{\vartheta}_{\mathrm{k}}(t)\Delta t+s_{\mathrm{k}}(t)\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}(\Delta t)^2\right]-\vartheta\mathrm{min}_{\mathrm{k}}<\left|\frac{\left[\dot{\vartheta}_{\mathrm{k}}(t+1)\right]^2}{2\ddot{\vartheta}\mathrm{act}_{\mathrm{k}}}\right|\Rightarrow s_{\mathrm{k}}(t)=1 \tag{3.11}$$

where ϑ_{k} is the current position of joint k, and $\dot{\vartheta}_{\mathrm{k}}(t+1)$ the velocity of joint k at the next timestep estimated from the acceleration of the current timestep before any constrained-based corrections.

The last step for trajectory generation is defining the limits and characteristics of these joints, which can be confirmed in Table 4.5.

This method was used to generate 10 one-minute trajectories for the CHARMIE robot. The same 10 trajectories will be used to evaluate all parametric options so that no choice is favored due to the generation of simpler trajectories.

The third preparatory stage for the parametric analysis is defining the objective function to evaluate each solution. The objective function is divided into two parameters, rewards that incentivize the solution's approximation to a theoretically perfect performance, and punishments that penalize the choice of unapplicable solutions. Since two mechanisms are being considered simultaneously, a variable was associated with each possible solution for each mechanism, variable i relates to the 3630 (11x330) configurations for the squatting mechanism, and variable j to the 40414 (11x11x334) configurations for the trunk flexion mechanism. The objective function evaluates the average of the performances between these two mechanisms as follows:

$$f(\mathrm{i},\mathrm{j}) = \frac{\mathrm{Rwd}(i) - \mathrm{Pnt}(i) + \mathrm{Rwd}(j) - \mathrm{Pnt}(j)}{2} \tag{3.12}$$

$$\max\left\{ f(i,j) \,|\, \left\{ i \in \mathbb{Z} \,|\, i \in [0,3630] \wedge j \in \mathbb{Z} \,|\, j \in [0,40414] \right\} \right\} \tag{3.13}$$

The mechanisms for the static balancing of the squatting and trunk flexion motions are identical, therefore, the reward and punishment functions used for both will be identical. In this section, all formulations will use the variable i referring to the squatting mechanism to avoid repetition.

The performance of the static balancing solution is rewarded on two factors: the minimization of the maximum force that must be applied by the actuator (F_{Max}); and the minimization of the average force (in modulus) applied by the actuator along the robot's trajectory (F_{Avg}). Two score variables (Sc_{Max} and Sc_{Avg}) evaluate the performance of the solution regarding these two situations by determining the percentual reduction of maximum and average force in comparison to the mechanisms' performance without any static balancing (the solution with index i = 0).

$$Sc_{\text{Max}}(i) = \frac{F_{\text{Max}}(0) - \left|F_{\text{Max}}(i)\right|}{F_{\text{Max}}(0)} \times 100 \tag{3.14}$$

$$Sc_{\text{Avg}}(i) = \frac{F_{\text{Avg}}(0) - \left|F_{\text{Avg}}(i)\right|}{F_{\text{Avg}}(0)} \times 100 \tag{3.15}$$

The forces required for these calculations are direct outputs from the inverse dynamic simulation. Using these two scores, the reward for solution i is calculated by using Equation 3.16.

$$\text{Rwd}(i) = \frac{Sc_{\text{Max}}(i) + Sc_{\text{Avg}}(i)}{2} \tag{3.16}$$

The next step is to determine the penalties for the objective function. These penalties exclude solutions where structural integrity is compromised, either due to excessive loads applied to the springs or the springs operating outside the dimensional limits imposed by their geometry.

Studying the tension springs starts by calculating their deformation y_{spring}. This is obtained from the coordinates of the spring ends A_s and B_s, and the spring's initial length L_0:

$$y_{\text{spring}} = \sqrt{\left(A_s x - B_s x\right)^2 + \left(A_s y - B_s y\right)^2 + \left(A_s z - B_s z\right)^2} - L_0 \tag{3.17}$$

Tension springs always have a pretension applied to them due to their manufacturing process (Budynas & Nisbett, 2011). In this study, each spring is assumed to be pre-stressed with the average recommended stress. The force F_{spring} exerted by the tension springs at each given point of the simulation depends on this pre-tension and the multiplication of the spring deformation with the spring constant. By expressing the spring force with known input variables, Equation 3.18 is obtained:

$$F_{\text{spring}} = \frac{\pi d^2 \left(\dfrac{230.97 \times 10^6}{e^{0.105\frac{D}{d}}} \right)}{8D} + \frac{d^4 G}{8D^3 \left(\dfrac{L}{d} + \dfrac{G}{E} \right)} y_{\text{spring}} \tag{3.18}$$

where D is the spring mean diameter, d is the spring wire diameter, L is the spring coiled length, G is the spring's shear modulus with a value of 69 GPa for AISI 302 stainless steel, and E is the Young's modulus of the steel with a value of 193 GPa.

With this expression, it is possible to determine the force exerted by any of the springs at any point during the multibody simulation. This force is used to verify the structural integrity of the springs.

The critical points, each with its safety factor, must be considered for studying tension springs: n_{body} is the safety factor for the shear stress in the main body of the spring, n_{A} is the safety factor for the point of the spring hook with the maximum bending and axial stresses, and n_{B} is the safety factor for the point of the spring hook with the maximum shear stress. These three safety factors are calculated by the following expressions:

$$n_{\mathrm{body}} = \frac{b_{\mathrm{body}} \pi d^3 A \left(4\frac{D}{d} - 3 \right)}{8 F_{\mathrm{spring}} d^{\beta} D \left(4\frac{D}{d} + 2 \right)} \tag{3.19}$$

$$n_{\mathrm{A}} = \frac{b_{\mathrm{A}} \pi d^2 A}{F_{\mathrm{spring}} d^{\beta} \left[\frac{4\left(\frac{2r_1}{d}\right)^2 - \frac{2r_1}{d} - 1}{4\frac{2r_1}{d}\left(\frac{2r_1}{d} - 1\right)} \frac{16D}{d} + 4 \right]} \tag{3.20}$$

$$n_{\mathrm{B}} = \frac{b_{\mathrm{B}} \pi d^3 A \left(8\frac{r_2}{d} - 4 \right)}{8 F_{\mathrm{spring}} d^{\beta} D \left(8\frac{r_2}{d} - 1 \right)} \tag{3.21}$$

where r_1 and r_2 are the radii that define the spring hook geometry, estimated to be half of the spring diameter D and 2.5 times the spring wire diameter d respectively, b_{body}, b_{A} and b_{B} are correction factors to estimate the maximum allowable stress for each critical point (see Table 3.4), and A and β are constants used to estimate the minimum tensile strength of the spring's material, whose value depends on the spring wire diameter (see Table 3.4).

Table 3.4 – Values of the constants β and A to estimate the minimum tensile strength of AISI 302 stainless steels, and the correction factors n_{body}, n_{A}, and n_{B} to estimate the maximum allowable stresses for the safety factors calculations, adapted from (Budynas & Nisbett, 2011)

b_{body}	b_{A}	b_{B}	**Wire diameter d [mm]**	**Exponent β**	$A[\mathrm{Pa \cdot m^{\beta}}]$
0.35	0.55	0.3	]0.3, 2.5]	0.146	681.00×10^6
0.35	0.55	0.3	]2.5, 5.0]	0.263	335.68×10^6
0.35	0.55	0.3	]5.0, 10.0]	0.478	107.16×10^6

If any of the calculated safety factors is lower than the overall safety factor n_{spring} during any point of the robot's operation, the structural integrity of the system is considered compromised, therefore,

the associated solution must be immediately discarded. Three auxiliary variables C_{body}, $\mathrm{C_A}$ and C_{B} were used to verify these conditions in the following expressions:

$$\exists n_{\mathrm{body}}(i) < n_{\mathrm{spring}} \Rightarrow C_{\mathrm{body}}(i) = 1 \tag{3.22}$$

$$\exists n_{\mathrm{A}}(i) < n_{\mathrm{spring}} \Rightarrow C_{\mathrm{A}}(i) = 1 \tag{3.23}$$

$$\exists n_{\mathrm{B}}(i) < n_{\mathrm{spring}} \Rightarrow C_{\mathrm{B}}(i) = 1 \tag{3.24}$$

After confirming the structural integrity of the spring, the final condition C_{ys} that must be verified is that tension springs cannot be compressed, which is confirmed by the expression:

$$\exists y_{\mathrm{spring}}(i) < 0 \Rightarrow C_{\mathrm{ys}}(i) = 1 \tag{3.25}$$

If any of the four previous conditions is infringed, the associated configuration must be immediately excluded. This was achieved by adding a weight of 1000 to the penalties. The final expression to determine the system's penalties is the following:

$$\mathrm{Pnt}(i) = \left[C_{\mathrm{body}}(i) + C_{\mathrm{A}}(i) + C_{\mathrm{B}}(i) + C_{\mathrm{ys}}(i) \right] \times 1000 \tag{3.26}$$

With the objective function fully defined, the final stage of this study is running the simulations and evaluating each solution for each scenario. To reduce computational strain, this analysis was conducted in two steps. The first step was analyzing the squatting (Table 3.5) and trunk flexion motions (Table 3.6) independently, using the auxiliary objective functions $f_1(i)$ and $f_2(j)$ to determine the ten best possible configurations for each subsystem independently on all ten trajectories.

Table 3.5 – Best combinations of inputs for the parametric optimization of the squatting mechanism

i	a_{i}	b_{i}	$f_1(i)$
3319	290	301	77.70
2975	230	270	42.26
2754	210	250	34.36
1901	310	172	22.17
1634	250	148	11.80
1732	230	157	10.73
1766	250	160	10.53
1633	230	148	10.34
1457	230	132	10.14
1731	210	157	9.40

Table 3.6 – Best combinations of inputs for the parametric optimization of the trunk flexion mechanism

j	c_j	d_j	e_j	$f_2(j)$
40060	310	70	331	77.89
39939	310	70	330	62.73
40181	310	70	332	53.54
36430	310	70	301	31.37
40302	310	70	333	23.87
32677	270	70	270	18.32
32553	210	70	269	17.28
32676	250	70	270	15.61
30256	250	70	250	15.48
30617	210	70	253	14.25

A satisfactory solution was determined for the squatting mechanism, with an average force reduction of over 70%. However, for the trunk flexion mechanism, the optimal objective function value for the original spring database had a mere 30% force reduction. This limitation was addressed by adding four new possibilities to the database, which correspond to the combination of 2—5 springs 301 in parallel. With these new options, an adequate solution was determined.

Both systems have a small influence on each other. The final step of the parametric optimization study was running the best-performing combinations of i and j parameters for all trajectories (see Table 3.7).

Table 3.7 – Results for the best-performing solutions of the static balancing parametric optimization study

$[i,\ j]$	$f(i,j)$ **for trajectory:**										$\mu_{f(i,j)}$
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
[3319, 40060]	77.8	71.1	79.0	79.8	78.9	77.3	77.8	76.6	75.0	72.2	76.6
[3319, 39939]	70.2	69.6	67.6	68.3	70.4	66.4	70.2	67.9	69.5	72.1	69.2
[3319, 40181]	65.6	57.4	69.1	70.1	68.7	66.9	63.6	62.8	63.0	54.9	64.2
[2975, 40060]	60.1	54.5	62.0	62.4	60.8	60.9	61.0	60.3	58.6	55.6	59.6
[2754, 40060]	56.1	50.5	58.0	58.5	56.9	56.9	57.0	56.2	54.5	51.6	55.6
[3319, 36430]	54.5	53.1	52.6	53.3	54.8	51.6	53.8	52.0	53.2	54.7	53.4
[2975, 39939]	52.5	53.0	50.7	50.9	52.3	50.0	53.4	51.5	53.0	55.4	52.3
[1901, 40060]	50.0	45.3	52.3	52.5	51.1	51.6	51.1	51.2	48.8	45.8	50.0
[3319, 40302]	50.8	42.6	55.5	57.1	56.0	53.3	48.6	46.3	48.6	37.1	49.5
[2754, 39939]	48.5	48.9	46.6	47.0	48.4	45.9	49.0	47.4	49.0	51.4	48.3

From the results displayed in Table 3.7, the best combination of solutions is the use of solution 3319 for the squatting mechanism and solution 40060 for the trunk flexion mechanism. These combinations of spring choices and geometries are displayed in Figure 3.4 and Figure 3.6 respectively. The simulation reports the following performance increase for the system:

- Squatting Actuator Maximum Load: 66.4% Reduction (258.3 N to 86.8 N)
- Squatting Actuator Average Load: 86.4% Reduction (196.1 N to 26.7 N)
- Trunk Flexion Actuator Maximum Load: 75.5% Reduction (1093.6 N to 278.4 N)
- Trunk Flexion Actuator Maximum Load: 79.0% Reduction (266.5 N to 55.9 N)

3.3.2. Structural Analysis of The Robot's Arms

The optimization of the robot's arms implies the adequate selection of three main components: DC motors that can exert the required torques, gears that can transmit the required torques, and shafts that can sustain the system's loads without structural damage. This optimization is an iterative process: selecting different components affects the system's mass, which affects the generated loads, requiring a new selection of components. This iterative development was achieved by alternating the optimization of the system between the CAD model and the multibody simulation.

The multibody simulation was key in calculating the loads generated by the robot during regular operation. The maximum loads in regular work conditions were determined by running a 10-minute-long randomized realistic trajectory where both arms were carrying the maximum payload of 1 kg.

Table 3.8 – Maximum joint forces and torques generated in the arms of the CHARMIE robot

Joint	F_x [N]	F_y [N]	F_z [N]	T_x [N.m]	T_y [N.m]	T_z [N.m]	$\|F\|$ [N]	$\|T\|$ [N.m]
5	82.70	80.71	10.08	17.03	16.68	<u>19.55</u>	83.04	19.89
6	62.32	58.88	59.34	<u>17.04</u>	19.77	4.84	62.61	19.94
7	45.25	43.71	42.02	9.96	11.48	<u>4.84</u>	46.06	12.53
8	22.25	19.53	19.55	<u>4.40</u>	5.32	1.91	24.62	6.11
9	15.46	20.73	15.57	2.93	3.34	<u>1.91</u>	20.96	3.49
10	10.08	9.81	15.63	<u>0.76</u>	1.01	0.26	16.37	1.03
11	5.75	5.78	12.03	0.22	<u>0.19</u>	0.01	12.47	0.22

The DC motors required both selecting the motor and the associated planetary gearbox. The required torques from each actuator set are underlined in Table 3.8. For the selection of adequate gearboxes, they must be able to withstand the maximum torque that will be exerted on them. For the

selection of motors, three things must be considered: after the gear ratio is applied, the motor speed should not be exceedingly slow at the maximum required torque, the maximum required motor torque (determined from the maximum load torque and the gearbox) should not exceed the stall torque at any point, and the motor's output shaft must be compatible with the selected shaft diameter. Table 3.9 lists the properties of the selected motors which fulfill all the required conditions. At no point will any of the motors produce speeds lower than 3 rpm, which is approximately the maximum joint velocity established in Table 4.5 for safety reasons.

Table 3.9 – Properties of the motor-gearbox assemblies selected for the arms of the CHARMIE robot

Joint	Gear Ratio	Stall Torque $[\mathbf{mN.m}]$	Max Required Motor Torque $[\mathbf{mN.m}]$	Max Output Speed $[\mathbf{rpm}]$
5	1:1296	918	15.09	3.99
6	1:1296	918	13.15	3.99
7	1:159	918	30.45	32.52
8	1:159	918	27.69	32.52
9	1:157	17.2	12.13	33.25
10	1:157	17.2	4.87	33.25
11	1:131	8.8	1.45	56.4

Joints 8, 10, and 11 employ bevel gears to convert the orientation of the output rotation of the motor. The gear selection requires the determination of the bending stress σ_{F} and strength σ_{FP}, as well as the contact stress σ_{H} and strength σ_{HP}, for the gear's teeth. These stresses were calculated using the ANSI/AGMA 2003-B97 (Budynas & Nisbett, 2011) method, expressed in equations 3.27—3.30.

$$\sigma_{\mathrm{F}} = \frac{W^{\mathrm{t}}}{b}\frac{K_{\mathrm{A}}K_{\upsilon}}{m_{\mathrm{et}}}\frac{Y_{\mathrm{x}}K_{\mathrm{H}\beta}}{Y_{\beta}Y_{\mathrm{J}}} \tag{3.27}$$

$$\sigma_{\mathrm{FP}} = \frac{\sigma_{\mathrm{F\,lim}}Y_{\mathrm{NT}}}{S_{\mathrm{F}}K_{\theta}Y_{\mathrm{z}}} \tag{3.28}$$

$$\sigma_{\mathrm{H}} = Z_{\mathrm{E}}\sqrt{\frac{W^{\mathrm{t}}}{bd_{\mathrm{p}}Z_{\mathrm{I}}}K_{\mathrm{A}}K_{\upsilon}K_{\mathrm{H}\beta}Z_{\mathrm{x}}Z_{\mathrm{xc}}} \tag{3.29}$$

$$\sigma_{\mathrm{HP}} = \frac{\sigma_{\mathrm{H\,lim}}Z_{\mathrm{NT}}Z_{W}}{S_{\mathrm{H}}K_{\theta}Z_{\mathrm{Z}}} \tag{3.30}$$

where the several factors necessary to determine the stresses and their calculations, are described in Appendix I.

The structural properties of the selected gears (which are compatible with the used steel shafts) are listed in Table 3.10. The values shown in this table already account for a 2.0 safety factor for the bending stress, and a 1.4 safety factor for the contact stress.

Table 3.10 – Properties of the bevel gears used on CHARMIE and corresponding stresses and strengths

Joint	Gear Teeth	Gear Modulus [mm]	Gear Material	Bending Stress [MPa]	Bending Strength [MPa]	Contact Stress [MPa]	Contact Strength [MPa]
8	22	1.5	Steel EN8	62.3	85.86	678.2	1242
10	18	1	Steel EN8	38.5	85.86	532.6	1242
11	16	1	HOSTAFORM C 9021	18.3	26.80	56.08	57.60

The last component that required a structural analysis was the shafts that carried the motion from the motors to the gears and components. All shafts will use a round solid profile of Grade 304 Stainless Steel. The shafts will be dimensioned using the maximum shear stress $\tau_{\max}$ which equated that at no point in the system's functioning can the generated shear stresses exceed the yield shear stress τ_{ced}. The maximum shear stress for a round profile at any given moment is given by Equation 3.31.

$$\tau_{\max} = \sqrt{\left(\frac{2F_{\text{A}}}{\pi d_{\text{S}}^{\,2}} + \frac{16M_{\text{F}}}{\pi d_{\text{S}}^{\,3}}\right)^2 + \left(\frac{16V}{3\pi d_{\text{S}}^{\,2}}\right)^2 + \left(\frac{16T}{\pi d_{\text{S}}^{\,3}}\right)^2} \tag{3.31}$$

where F_{A} is the axial force, d_{S} the shaft diameter, M_{F} the bending moment, V the transversal cutting force, and T the torsional moment.

This is a very conservative formulation of the maximum shear stress since it assumes all tension types will achieve maximum intensity at the same point of the body. The value of the required forces and moments can be obtained directly from the multibody simulation, using, once again, the same 10-minute motion. The expression to evaluate the maximum shear stress was calculated for each timestep of the simulation.

The yield shear stress for the considered material can be determined by the yield stress using the following notation:

$$\tau_{\text{yield}} = \frac{\sigma_{\text{yield}}}{2} \tag{3.32}$$

The chosen shaft diameters, corresponding stresses, and resulting safety factors are displayed in Table 3.11.

Table 3.11 – Shear Stresses and safety factors of the stainless-steel shafts of CHARMIE's arms

Joint	Shaft Material	Diameter [mm]	Max Shear Stress [MPa]	Shear Yield Stress [MPa]	Safety Factor
5	Grade 304 Steel	20	12.66	102.5	8.1
6	Grade 304 Steel	20	12.73	102.5	8.1
7	Grade 304 Steel	12	36.98	102.5	2.8
8	Grade 304 Steel	8	60.86	102.5	1.7
9	Grade 304 Steel	8	34.77	102.5	3.0
10	Grade 304 Steel	5	41.54	102.5	2.5
11	Grade 304 Steel	5	4.62	102.5	22.2

All previous results were implemented into CHARMIE's arm design, resulting in the design shown in Figure 3.8.

3.4. Prototype Construction

To confirm the validity of the proposed concepts, test the validity of the used methodologies, and confirm the viability of the construction of the robot, a physical prototype of CHARMIE has been constructed. This prototype uses off-the-shelf components, steel sheet metal which was laser cut and bent to create the desired profiles, steel standardized profiles which were laser cut into the desired final shape, and steel parts produced through metal lathing and hole drilling. Full replicability of the constructed prototype is guaranteed with the aid of the technical drawings of the non-standardized parts available in Appendix III. For a better understanding of the assembly steps of some of the most complex systems, Appendix IV provides a step-by-step assembly guide of relevant subsystems of CHARMIE.

The current prototype does not contemplate all the subsystems developed in this document. The three main mechanisms that were fully prototyped were: i) the locomotion and suspension system, ii) the squatting motion mechanism, and iii) the trunk flexion motion mechanism.

The first system to be prototyped was the suspension and locomotion system (see Figure 3.11). Regarding standardized components, this system employs 203mm double aluminum omni-directional wheels, UCFL 205 flange bearings, EMG49 DC Motors, and the compression springs detailed in Table 3.2. Metal lathing was used to produce the shafts transmitting the motor powers to the wheel and guiding the suspension springs. The remaining components were made from laser-cut steel profiles.

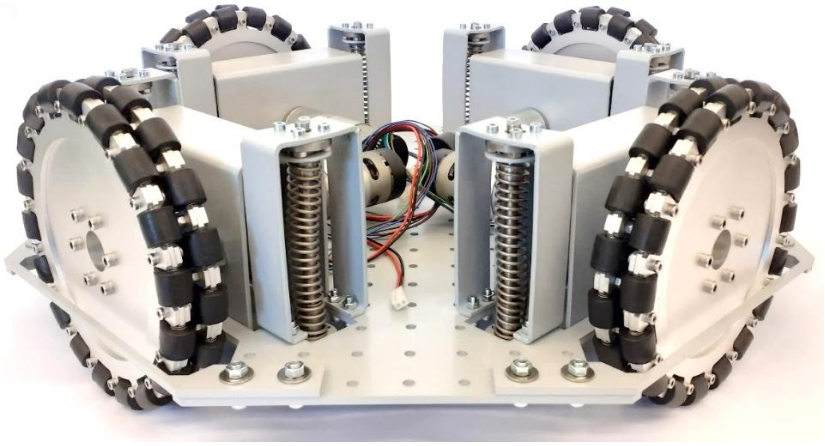


Figure 3.11 – Prototype of CHARMIE's suspension and locomotion systems

The squatting and trunk flexion motion mechanisms were prototyped simultaneously see Figure 3.12. Both mechanisms are actuated with 1500 N linear actuators with a stroke of 150mm. The revolute joints employ 12 mm diameter stainless steel rods and KFL12 or KP12 bearings. The two types of linear joints use 20 mm diameter stainless steel rods supported by SHF20 or SK20 shaft supports, with the motion being guaranteed by LMK20LUU and SC20UU linear bearings. The types of tension springs used are described in Table a1.1 in the annexes. Most used sheet metal was bent into a u-profile to increase the resistance to flexion stresses. Several additional electronic components are added into the system, which are supported by 3D printed parts using Fused Deposition Modeling (FDM) of Polylactic Acid (PLA) filament. Due to the iterative nature of product development, the specific electronics of CHARMIE, and their configuration, can still be subject to future change, so these components are merely auxiliary and not finalized.

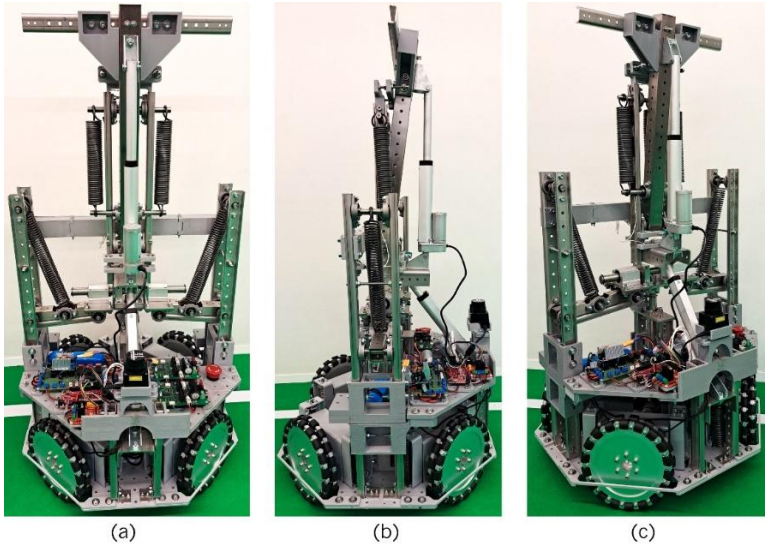


Figure 3.12 – Prototype of CHARMIE's squatting and trunk flexion motion mechanisms: a) front view, b) side view, and c) three-quarter-view.

The prototype with the squatting and trunk flexion motion mechanisms was used to test the efficiency of the static balancing solutions. The electric power consumed over time was compared between including or removing the static balancing springs (see Figure 3.13). Both solutions were tested for the same motions: the squatting actuator started compressed, then extended at a velocity of 3.9 mm/s for 31 s, it stopped for 1 second, and then compressed with the same velocity for the same duration; the trunk flexion actuator started compressed, then extended at a velocity of 3.8 mm/s for 26 seconds, it stopped for 1 second, and then compressed with the same velocity for the same duration.

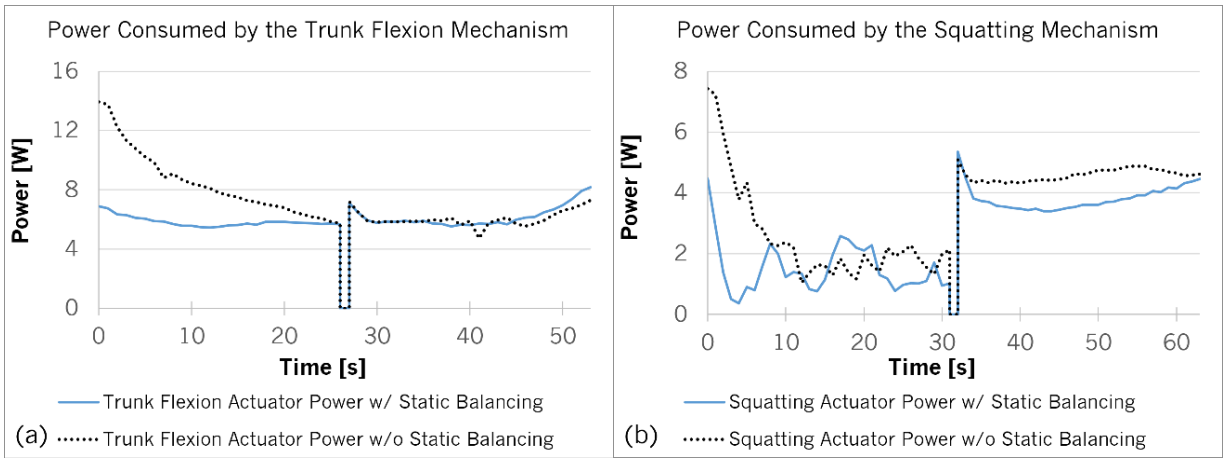


Figure 3.13 – Comparison of the power consumed by the actuators of the trunk flexion and squatting mechanism with or without static balancing springs

The energy consumption for the squatting mechanism was reduced by 25.5% and for the trunk flexion mechanism by 16.0%. These results are already satisfactory, but it is predicted that as the prototype further develops and approaches the configuration of the multibody simulation environment the results will improve. Improving the manufacturing quality of the prototype and using better-quality joint tolerances will also increase the prototype's performance.

Testing CHARMIE in work scenarios, where all systems are included and functional, was vital to confirm the system's validity. To this end, a temporary arm and head were added to CHARMIE, as well as all the required electronics for the prototype to be fully functional (see Figure 3.14).

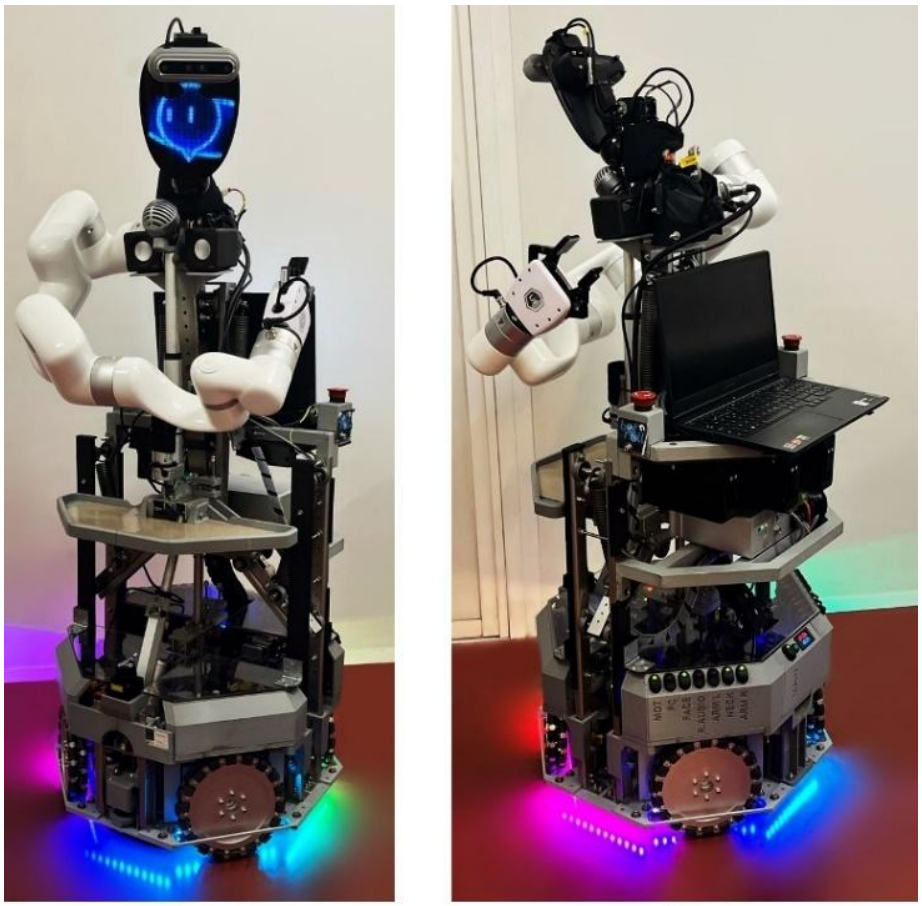


Figure 3.14 – Current functional prototype of the CHARMIE robot a) front view, b) rear view.

The current prototype is being tested and optimized for performing domestic tasks. One of the ways to analyze and benchmark the robot's performance is to participate in international scientific competitions, namely RoboCup@Home, specifically developed with the application-based evaluation of domestic service robots in mind (Wisspeintner et al., 2009). The current iteration of CHARMIE achieves strong performances in this competition, placing 1st in the 2024 national edition, and 5th out of 17 in the 2024 international edition. These results show that the robot is already able to autonomously perform a set of vital tasks in domestic environments.

3.5. Summary and Discussion

In this chapter, the main specifications for CHARMIE were defined, establishing it as a mobile manipulator robot with a human-inspired shape, wheeled omnidirectional locomotion, an articulated hip, and two 6-DOF manipulator arms for object interaction.

These requirements were employed to establish a preliminary design of CHARMIE using CAD software to produce 3D models. Basic principles from mechanism design and structural analysis were used to produce a safe, reliable, and easy-to-produce model whose manufacturing uses standardized components, metal lathing, and the laser cutting and bending of steel sheet metal.

The preliminary study of the robot identified two critical systems that required a more in-depth study, namely the static balancing of the hip mechanisms (to reduce actuator loads) and the structural analysis of the robot's arms. Both studies employed the developed dynamic multibody model simulator. Regarding the static balancing solution, a parametric optimization study was implemented which allowed the selection of ideal springs and geometries, resulting in a reduction of over 65% in all simulator actuator forces. The structural analysis of the arms allowed the selection of motors, gears, and shafts that are compatible with each other and can support the loads generated in the serial manipulator component of the robot.

A prototype that fully represents three of the developed systems: locomotion and suspension, squatting motion mechanism, and trunk flexion motion mechanism; was successfully manufactured and assembled. All designed features and calculated rigidities were validated by the physical model. This model also confirmed the efficacy of the static balancing solution, which reduced actuator energy consumption by approximately 20%. In future work, the proposed manipulator arms and the robot's head will also be prototyped. The current version of the prototype employs a temporary arm and head, which allow the robot to be fully functional. The functional prototype was able to autonomously perform domestic tasks, achieving successful results in international robotics competitions (RoboCup@Home) which are used as benchmarks for the performance of domestic service robots.

"*Move swift as the Wind and closely formed as the Wood.*
Attack like the Fire and be still as the Mountain."
Sun Tzu

4. MULTIBODY MODEL OF CHARMIE

The development of multibody models plays a pivotal role in the research of robotic systems. A well-implemented multibody model provides a solid computational basis for a set of key studies by assisting in the analysis of the robot's motion and trajectories, allowing the comparison between the efficiency of different control solutions, and supporting the design optimization of mechanical systems (Siciliano et al., 2009).

Multibody research on humanoid robots is mostly conducted on commercial (or open-source) applications due to the software's ability to tackle complex studies using a user-friendly interface without requiring in-depth studies (Ivaldi et al., 2014). However, describing the behavior of a system by deriving its equations of motion allows the construction of more focused, versatile, adaptive, and highly efficient models inside an integrated development environment. This latter method gives researchers full control over the simulation environment, allowing its optimization for each study at hand. These benefits are highly advantageous for the ongoing iterative development of a robotic system such as CHARMIE.

CHARMIE's multibody model was fully implemented in Python, resulting in a simulation environment that is both hardware-independent and can be implemented into the robot's embedded computer. The construction of this model follows four main steps: inputting the description of the multibody system into the computation simulation, solving the robot's forward kinematic analysis using a methodology centered around a recursive algorithm (Gonçalves et al., 2022c), solving the robot's inverse dynamics by using the recursive Newton-Euler algorithm presented in (Gonçalves et al., 2022a), and finally modeling additional formulations to allow the robot's model to interact with its surroundings.

4.1. Description of the Multibody System

CHARMIE is a complex human-inspired mobile manipulator robot (see Figure 4.1) with 45 bodies and 54 joints, assembled in a configuration that results in a total of 36 degrees of freedom (including the six degrees of freedom that allow the robot to navigate through space).

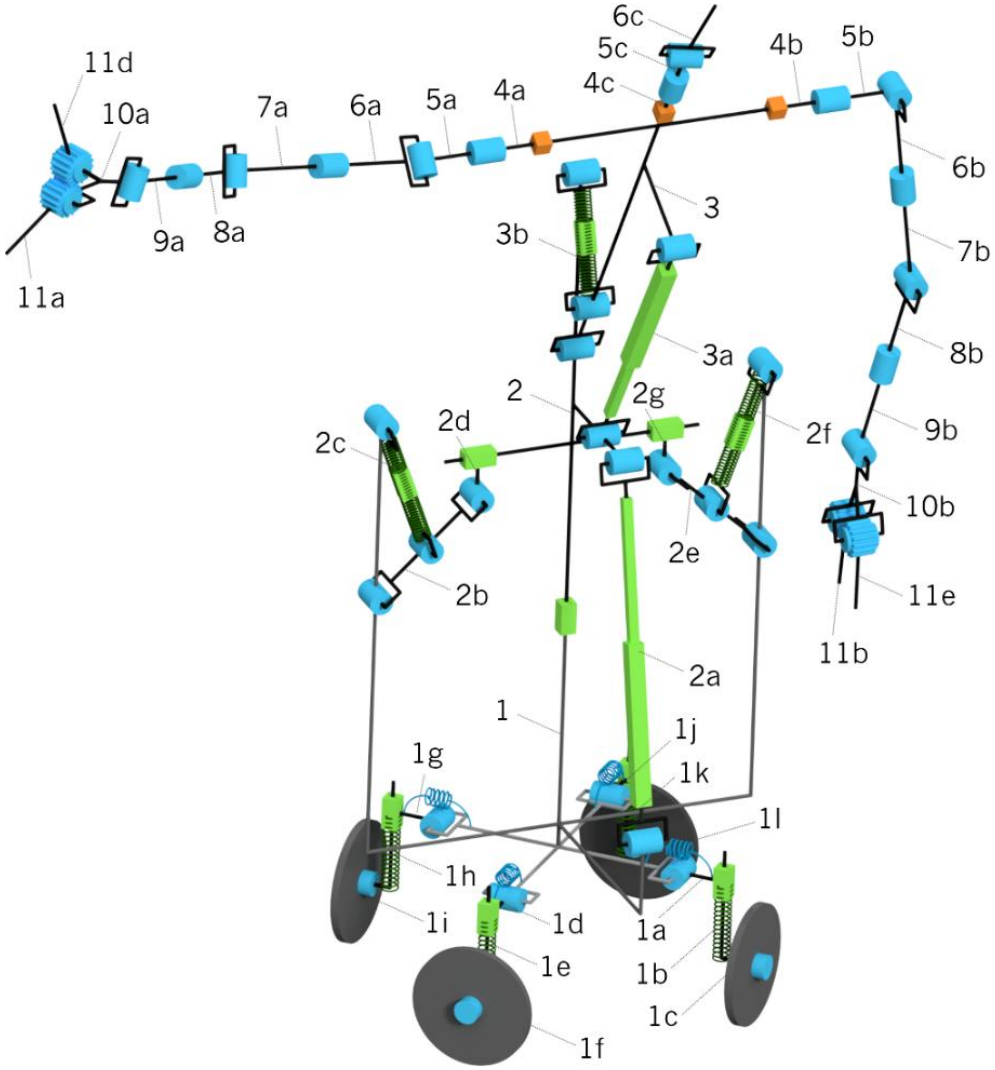


Figure 4.1 – Kinematic diagram showcasing the bodies and degrees of freedom of the CHARMIE robot

The robot's base is considered the main reference of the kinematic chain. As the kinematic chains progress from the base towards the end-effector (wheels, claws, or head), each following body $i + 1$ succeeds the previous one i. In turn, bodies $i - 1$, closer to the robot's base, are preceding link i.

The analyzed multibody system was represented as a kinematic tree (see Figure 4.2). The base hip and hip torso are connected by a closed and over-constrained mechanism. All other branches are

serial kinematic chains that progress either from the base to the wheel (for the locomotion and suspension systems), or from the robot's torso to its end-effectors (for both manipulation arms and the robot's head).

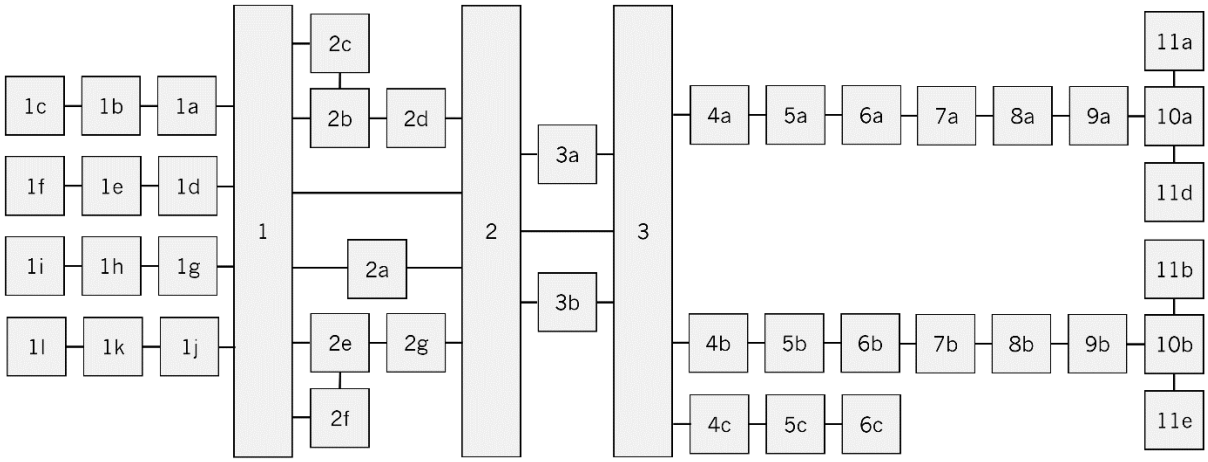


Figure 4.2 – Kinematic tree representing CHARMIE's multibody system

The multibody system was fully characterized by using a systematic notation to describe each body and joint based on the required inputs of the implemented computational algorithm.

4.1.1. Characteristics of the Bodies

The kinematic and dynamic analysis of CHARMIE required the definition of each of the 45 bodies' geometry, mass, and inertial properties.

The geometry of each body is expressed using the cartesian coordinates of points. The first step was defining a set of relevant coordinate axes in which these points will be expressed. When referring to the coordinates of a point, a superscript defines the coordinate axis that is being used with the following notation:

- $\mathbf{P}_i'$ The coordinates of point P of link i being expressed in the local coordinates of link i;
- $\mathbf{P}_i''$ The coordinates of point P of link i expressed in an auxiliary axis representing body i rotated around $\mathbf{O}_i'$ to its global orientation considering the current configuration of the robot;
- $\mathbf{P}_i$ The coordinates of point P of link i in the global coordinate axis;
- $\mathbf{P}_i'^j$ The coordinates of point P of link i expressed in the local coordinates of body j.

The shape of each body was established in local coordinates by using point clouds. Any number of points can be used based on the required level of detail. Only four points are required for running the computational algorithms, namely:

- Point O_i, the origin of the local coordinate system of body i;
- Point A_i, the connection point between body i and its main preceding link in the kinematic chain;
- Points B_i the connection points between body I and its succeeding links in the kinematic chain (a singular body can have more than one of these points);
- Point C_i, the center of mass of body i.

Point A_i guarantees the connection between body i and its preceding link $i-1$. When the joint preceding link i is revolute, point A_i is overlapped with O_i. If the joint preceding link i is prismatic, the position of A_i is determined with the expression:

$$\mathbf{A}_i' = \mathbf{O}_i' + d_i \mathbf{t}_i \tag{4.1}$$

where d_i is the distance between the origin of body i and its connection point to the preceding link in the kinematic chain, and $\mathbf{t}_i$ is the unit vector defining the orientation of the prismatic joint expressed in local coordinates (See Figure 4.3).

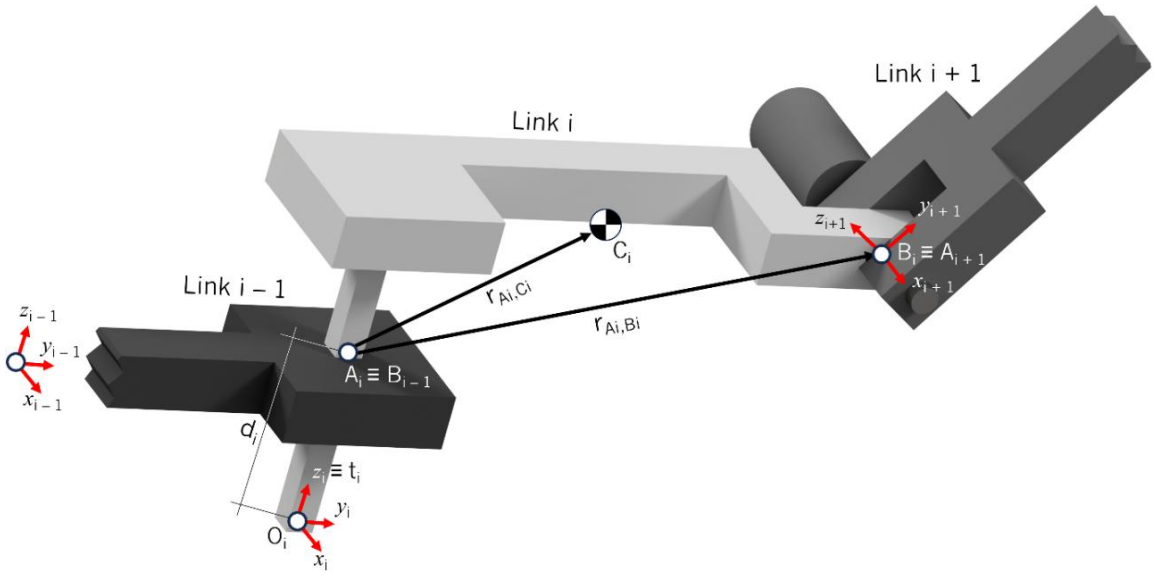


Figure 4.3 – Schematic representation of the notation used for the description of the body geometries.

In the current iteration of the simulator, the coordinates of a minimal number of points are manually computed. Figure 4.4.a showcases the identification of the key points for CHARMIE's torso in the CAD model, whose coordinates are used to build Table 4.1. In turn, these points are used to build

the computational model of Figure 4.4.b. This process was repeated for all bodies; the geometric properties of which are listed in Appendix IV.

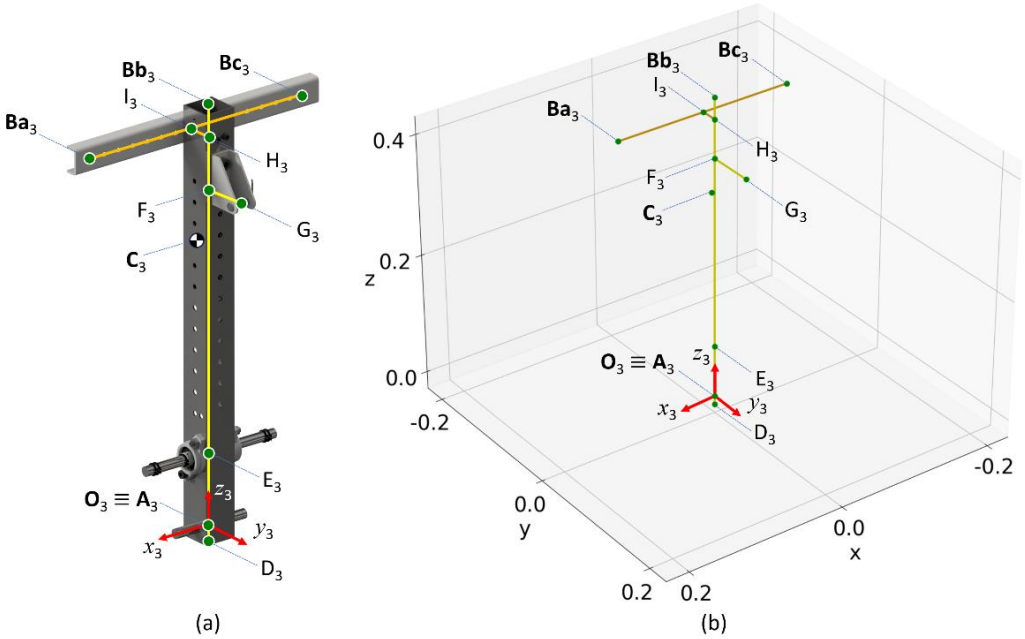


Figure 4.4 – Identification of the key points required to describe the geometry of the robot's torso in a) the CAD model, and b) the computational simulation (see Table 4.1 for the associated point coordinates).

Table 4.1 – Coordinates of the points used to characterize the robot's torso represented in Figure 4.4.

Point	Local coordinates [mm]		
	x	y	z
O_3	0.0	0.0	0.0
A_3	0.0	0.0	0.0
Ba_3	110.0	− 20.0	460.0
Bb_3	− 110.0	− 20.0	460.0
Bc_3	0.0	0.0	495.0
C_3	0.0	− 5.7	338.5
D_3	0.0	0.0	− 15.0
E_3	0.0	0.0	85.0
F_3	0.0	0.0	397.2
G_3	0.0	56.0	397.2
H_3	0.0	0.0	460.0
I_3	0.0	− 20.0	460.0

The vital information for characterizing the bodies is inputted into the computer simulation in the form of two vectors: $\mathbf{r}_{Ai,Bi}$ and $\mathbf{r}_{Ai,Ci}$. Vector $\mathbf{r}_{Ai,Bi}$ defines the relative position of two consecutive joints, while $\mathbf{r}_{Ai,Ci}$ expresses the position of the center of mass as illustrated in Figure 4.3. Due to the existence of

prismatic joints, these vectors can be obtained from the constant vectors $\mathbf{r}_{Oi,Bi}$ (see Table 4.4) and $\mathbf{r}_{Oi,Ci}$ (see Table 4.2) using the expressions:

$$\mathbf{r}_{Ai,Bi} = \mathbf{r}_{Oi,Bi} - d_i \mathbf{t}_i \tag{4.2}$$

$$\mathbf{r}_{Ai,Ci} = \mathbf{r}_{Oi,Ci} - d_i \mathbf{t}_i \tag{4.3}$$

Table 4.2 – Names and mass properties of CHARMIE's bodies.

Body i	**Body Name**	**Mass** $[\mathbf{kg}]$	$r_{Oi,Ci}$ $[\mathbf{mm}]$		
			x	y	z
1	Base	23.0	−1.5	26.3	165.8
1a \ 1d \ 1g \ 1j	Suspension system intermediary link	0.59	0.0	0.0	70.2
1b \ 1e \ 1h \ 1k	DC motor assembly	3.43	0.0	−13.7	52.3
1c \ 1f \ 1i \ 1l	Swedish omni wheel	1.56	0.0	50.6	0.0
2a	Lower linear actuator	1.12	0.0	−7.7	124.9
2b \ 2e	Left \ Right lower static balancing lever	0.50	1.0	0.0	81.8
2c \ 2f	Left \ Right lower static balancing spring	0.39	0.0	0.0	127.7
2d \ 2g	Left \ Right lower static balancing linear bearing	0.88	−23.9	0.0	0.0
2	Hip	5.77	0.0	18.0	348.5
3a	Upper linear actuator	1.12	0.0	7.7	131.0
3b	Upper static balancing spring	1.18	0.0	0.0	113.7
3	Torso	1.59	0.0	−5.7	338.5
4a \ 4b	Left \ Right scapula	2.68	∓28.0	0.0	56.0
5a \ 5b	Left \ Right shoulder	1.92	∓28.6	60.7	0.0
6a \ 6b	Left \ Right proximal arm	1.61	0.0	0.0	54.3
7a \ 7b	Left \ Right distal arm	2.20	∓6.0	0.0	115.4
8a \ 8b	Left \ Right proximal forearm	0.40	0.0	−0.8	54.5
9a \ 9b	Left \ Right distal forearm	0.49	∓8.6	−0.3	80.8
10a \ 10b	Left \ Right hand	0.37	∓1.1	5.5	29.9
11a \ 11b	Left \ Right thumb claw	0.01	∓34.7	0.0	26.4
11d \ 11e	Left \ Right opposite claw	0.01	±34.7	0.0	26.4
4c	Lower cervical spine	0.28	0.0	−1.9	5.6
5c	Neck	0.55	3.0	3.8	10.8
6c	Head	0.57	−14.6	18.7	65.7

The final stage was defining the mass and inertia properties of the bodies. Both characteristics were estimated from the CAD model of the robot. The mass of each part was defined, which in turn was used to calculate the total masses, center of mass positions (see Table 4.2), and inertia properties (see Table 4.3) of each body.

Table 4.3 – Inertial properties the bodies from the multibody model of CHARMIE.

Body i	Inertia $[\mathrm{kg.mm^2}]$					
	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
1	9.8×10^5	1.2×10^6	8.5×10^5	2.9×10^3	2.1×10^3	7.1×10^3
1a \ 1d \ 1g \ 1j	1.5×10^3	7.1×10^3	6.3×10^3	0.0	0.0	0.0
1b \ 1e \ 1h \ 1k	1.2×10^4	1.3×10^4	1.6×10^4	-0.2×10^0	-5.0×10^0	3.8×10^2
1c \ 1f \ 1i \ 1l	4.3×10^3	6.6×10^3	4.3×10^3	0.0	0.0	0.2×10^0
2a	8.3×10^3	8.0×10^3	4.8×10^2	-0.5×10^0	4.5×10^0	-7.2×10^2
2b \ 2e	1.1×10^3	1.0×10^3	1.7×10^2	0.0	-9.6×10^0	-1.0×10^0
2c \ 2f	1.6×10^3	1.6×10^3	9.8×10^1	0.0	0.0	0.0
2d \ 2g	1.2×10^3	5.1×10^2	1.3×10^3	0.0	0.0	0.0
2	1.9×10^5	2.0×10^5	2.5×10^4	1.7×10^0	3.2×10^1	3.2×10^3
3a	1.1×10^4	1.0×10^4	4.8×10^2	-0.5×10^0	-4.6×10^0	7.8×10^2
3b	3.2×10^3	7.9×10^3	5.0×10^3	0.0	-4.4×10^0	0.5×10^0
3	4.2×10^4	4.3×10^4	1.2×10^3	-0.4×10^0	-1.1×10^0	2.4×10^2
4a \ 4b	1.4×10^4	1.5×10^4	2.1×10^3	0.0	7.4×10^2	1.3×10^0
5a \ 5b	1.5×10^3	5.3×10^3	5.5×10^3	$\mp 4.2 \times 10^2$	0.0	-0.4×10^0
6a \ 6b	4.4×10^3	4.4×10^3	1.0×10^3	0.0	0.0	9.2×10^0
7a \ 7b	9.4×10^3	9.9×10^3	1.8×10^3	-1.5×10^0	-7.3×10^2	-7.2×10^0
8a \ 8b	8.2×10^2	8.3×10^2	9.5×10^1	0.0	-8.9×10^0	-1.8×10^1
9a \ 9b	1.2×10^3	1.5×10^3	3.4×10^2	-1.4×10^0	-1.7×10^2	-1.4×10^1
10a \ 10b	5.3×10^2	7.3×10^2	4.9×10^2	$\mp 5.6 \times 10^0$	7.8×10^0	$\pm 2.0 \times 10^1$
11a \ 11b	7.2×10^0	1.0×10^1	3.3×10^0	0.0	$\mp 3.3 \times 10^0$	0.0
11d \ 11e	7.2×10^0	1.0×10^1	3.3×10^0	0.0	$\pm 3.3 \times 10^0$	0.0
4c	1.9×10^2	1.8×10^2	3.1×10^2	0.0	0.0	2.3×10^0
5c	3.9×10^2	3.7×10^2	6.0×10^2	2.2×10^1	-2.9×10^1	3.1×10^1
6c	7.6×10^2	6.2×10^2	4.8×10^2	1.8×10^0	5.9×10^0	-4.6×10^1

With the geometrical parameters and physical properties described, all bodies are completely characterized for the multibody simulation of the CHARMIE robot.

4.1.2. Description of the Joints

The joints of CHARMIE are described concerning the two consecutive bodies they connect. The forward kinematic algorithm constructs each body and joint sequentially in the following order of operations: a) creation of body i, b) definition of the joints succeeding i concerning i, and c) definition of joints preceding body i+1. After step c) is complete, all information required to define a new body i is established. The joints are first modeled by defining their position in body i using vectors $\mathbf{r}_{Ai,Bi}$ obtained from Table 4.4 and Equation 4.3. When modeling each successive body $\mathrm{i}+1$, the information of the

preceding joints is used to determine the position and orientation of the body. The preceding joints are described by their current displacement\orientation and two variables: JT_i which defines the type of joint preceding body i as: P = prismatic, R = revolute, or F = fixed; and JO_i which defines the joint orientation defined by a common axis between both bodies.

The simulation of CHARMIE's control requires establishing the actuator's kinematic properties. The required properties are the minimum $\vartheta\mathrm{min}_\mathrm{k}$ and maximum $\vartheta\mathrm{max}_\mathrm{k}$ position or angle limits (for the joints that do not perform continuous rotations), as well as the maximum velocity $\dot{\vartheta}\mathrm{max}_\mathrm{k}$ and acceleration $\ddot{\vartheta}\mathrm{act}_\mathrm{k}$ which for each actuator. CHARMIE has a total of 22 actuators; Table 4.5 lists the relevant properties for all actuators, referring to each actuator by the body index whose motion they are controlling.

Table 4.4 – Properties defining the joints of CHARMIE. Bodies marked with * have variable values due to being bodies with variable lengths.

Body i	**Preceding Joint**		**Succeeding Joints**			
	$\boldsymbol{JT_i}$	$\boldsymbol{JO_i}$	**Succeeding Body**	$\boldsymbol{r_{\mathrm{Oi,Bi}}}$ **[mm]**		
				x	**y**	**z**
1	-	-	1a	−129.8	129.8	5.0
			1d	−129.8	−129.8	5.0
			1g	129.8	−129.8	5.0
			1j	129.8	129.8	5.0
			2a	0.0	238.0	204.0
			2b	245.0	0.0	341.0
			2c	245.0	0.0	631.0
			2e	−245.0	0.0	341.0
			2f	−245.0	0.0	631.0
			2	0.0	0.0	258.0
1a \ 1d \ 1g \ 1j	R	x	1b \ 1e \ 1h \ 1k	0.0	0.0	105.0
1b \ 1e \ 1h \ 1k	P	z	1c \ 1f \ 1i \ 1l	0.0	9.1	60.0
1c \ 1f \ 1i \ 1l	R	y	-	-	-	-
2a	R	x	2	0.0	1.5	*351.0
2b \ 2e	R	y	2d \ 2g	0.0	0.0	170.0
2c \ 2f	R	y	2b \ 2e	0.0	0.0	*255.4
2d \ 2g	R	y	2	−45.0	0.0	0.0
2	P	z	3a	0.0	105.3	381.2
			3b	0.0	0.0	708.2
			3	0.0	0.0	398.2
3a	R	x	3	0.0	−1.5	*398.2
3b	R	x	3	0.0	0.0	*227.4

Body i	Preceding Joint		Succeeding Joints			
	JT_{i}	JO_{i}	Succeeding Body	$r_{\mathrm{Oi,Bi}}$ [mm]		
				x	y	z
3	R	x	4a	110.0	−20.0	460.0
			4b	−110.0	−20.0	460.0
			4c	0.0	0.0	495.0
4a \ 4b	F	-	5a \ 5b	∓46.2	0.0	100.2
5a \ 5b	R	z	6a \ 6b	∓120.3	66.5	0.0
6a \ 6b	R	x	7a \ 7b	0.0	0.0	150.8
7a \ 7b	R	z	8a \ 8b	∓89.8	0.0	170.4
8a \ 8b	R	x	9a \ 9b	0.0	0.0	130.6
9a \ 9b	R	z	10a \ 10b	∓93.8	0.0	126.6
10a \ 10b	R	x	11a \ 11b	±26.3	5.9	68.1
			11d \ 11e	∓26.3	5.9	68.1
11a \ 11b	R	y	-	-	-	-
11d \ 11e	R	y	-	-	-	-
4c	F	-	5c	0.0	0.0	46.3
5c	R	z	6c	14.7	−5.3	18.3
6c	R	x	-	-	-	-

Table 4.5 – Joint actuator velocities, accelerations and position constraints for the CHARMIE robot

Actuated Body	$\vartheta \mathrm{min}_{\mathrm{k}}$	$\vartheta \mathrm{max}_{\mathrm{k}}$	$\dot{\vartheta} \mathrm{max}_{\mathrm{k}}$	$\ddot{\vartheta} \mathrm{act}_{\mathrm{k}}$
1c \ 1f \ 1i \ 1l	-	-	180 °/s	20 °/s^2
2	262 mm	383 mm	50 mm/s	50 mm/s^2
3	300 mm	398 mm	50 mm/s	50 mm/s^2
5c	− 75°	75 °	20 °/s	12.5 °/s^2
6c	− 50°	45 °	20°/s	12.5°/s^2
5a \ 5b	− 180°	180°	20°/s	12.5°/s^2
− 6a \ 6b	0°	90°	20°/s	12.5°/s^2
7a \ 7b	− 20°	120°	20°/s	12.5°/s^2
−8a \ 8b	0°	135°	20°/s	12.5°/s^2
9a \ 9b	− 180°	180°	20°/s	12.5°/s^2
− 10a \ 10b	− 85°	85°	20°/s	12.5°/s^2
11a \ 11b	0°	55°	20°/s	12.5°/s^2

The full description of the bodies and joints of CHARMIE allows the assemblage of the computational multibody model of the robot.

4.2. Forward Kinematics Analysis

The forward kinematics study aims to use the geometric characteristics of the bodies and joints to obtain the assembled model of the robot in a given configuration. Different configurations can be evaluated along iterations to analyze the robot's motion over time. The evolution of the position and orientations of joints is determined from the velocities and accelerations of the joints, which are defined by the robot's trajectory control.

The approach used for the forward kinematic analysis is centered around the original recursive algorithm described in (Gonçalves et al., 2022c). This methodology follows three main steps: the first preparatory step involves converting the output from the control system of the robot (actuator configurations) into the inputs of the recursive algorithm (joint configurations) for the indirectly actuated joints; the second step is defining and running the recursive algorithm, which employs cartesian coordinates and Euler angle rotations to model each body of CHARMIE sequentially for serial and branched kinematic chains; and finally the third step completes the model by studying the configuration of the bodies in closed and overconstrained loops.

4.2.1. Geometric Analysis of the Indirectly Actuated Joints

Charmie only has two indirectly actuated joints: the prismatic squatting motion between bodies 1 and 2, which is controlled by a prismatic actuator without a parallel alignment to the direction of motion; and the revolute trunk flexion motion between bodies 2 and 3, actuated by a prismatic actuator. The geometric parameters required to study both mechanisms are shown in Figure 4.5.

The full multibody model is yet to be assembled, therefore distances between points that are a part of two different bodies cannot be used. The only thing which is known is the local geometry of both bodies, and the connection between them defined by the linear actuator. When studying joint 2, the aim is to determine the linear displacement d_2 as a function of the position of the linear actuator ϑ_2. This can be determined in the local coordinates of body 2 by the expression:

$$d_2 = \mathrm{A}_2 z - \mathrm{O}_2 z \tag{4.4}$$

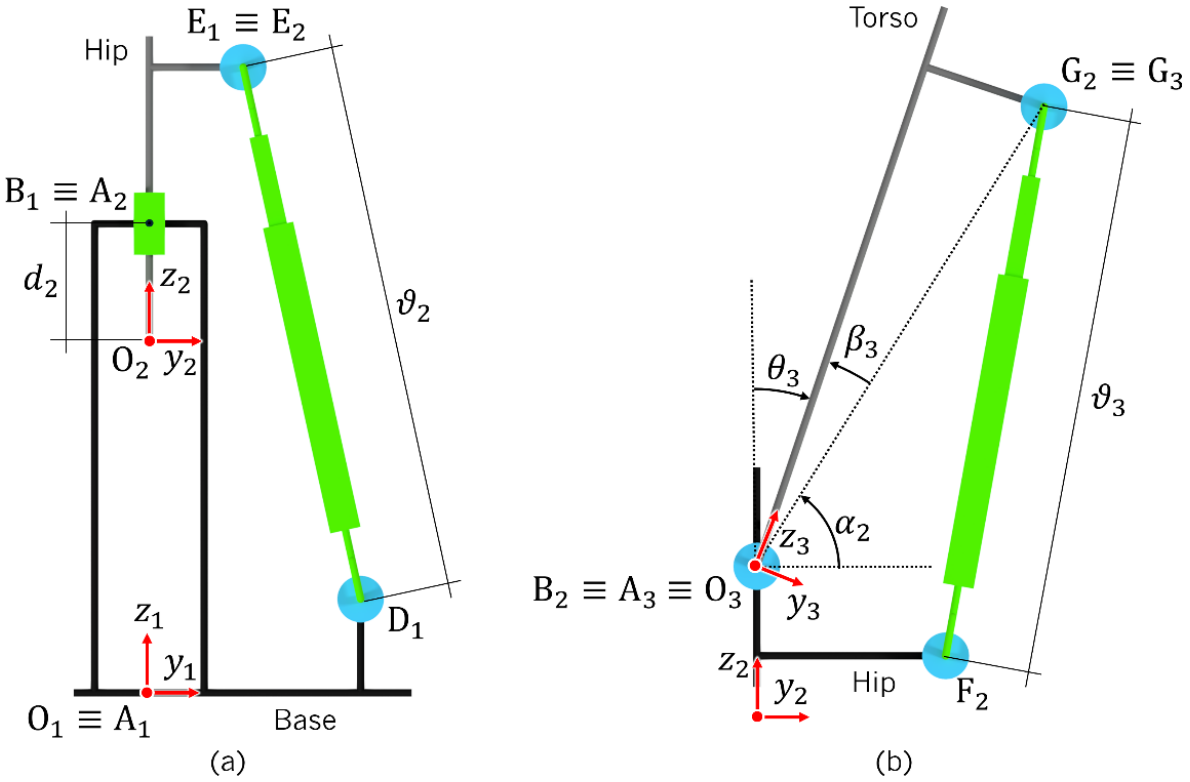


Figure 4.5 – Geometric parameters for the analysis of the indirectly actuated joints of CHARMIE, namely (a) the joint preceding body 2 and (b) the joint preceding body 3.

The coordinates of these two points can be expressed concerning body 1. Point A_2 directly overlaps with point B_1; the z-coordinate of point O_2 can be calculated regarding body 1 by starting from point D_1 and using two auxiliary measurements, the vertical displacement caused by the linear actuator (calculated using the Pythagorean theorem), and the constant vertical displacement between E_2 and O_2 defined by the geometry of body 2. This can be equated as follows:

$$d_2 = \mathrm{B}_1'z - \left[\mathrm{D}_1'z + \sqrt{\vartheta_2^{\,2} - \left(\mathrm{E}_1'y - \mathrm{D}_1'y\right)^2} - \left(\mathrm{E}_2'z - \mathrm{O}_2'z\right)\right] \tag{4.5}$$

This expression establishes the desired relationship between d_2 and ϑ_2 since all other variables are known constants.

When studying joint 3, the aim is to determine the angular orientation θ_3 as a function of the position of the linear actuator ϑ_3. The value of θ_3 can be determined using the auxiliary angles α_2 and β_3 using the expression:

$$\theta_3 = \alpha_2 + \beta_3 - 90^{\circ} \tag{4.6}$$

Angle β_3 is constant, characterized by the geometry of body 3 as:

$$\beta_3 = \arctan\left(\frac{\mathrm{G}_3'y - \mathrm{A}_3'y}{\mathrm{G}_3'z - \mathrm{A}_3'z}\right) \tag{4.7}$$

Angle α_2 is a function of the actuator length ϑ_2. This angle can be expressed regarding body 2 as:

$$\alpha_2 = \arctan\left(\frac{\mathrm{G}_2'z - \mathrm{B}_2'z}{\mathrm{G}_2'y - \mathrm{B}_2'y}\right) \tag{4.8}$$

The position of G_2 in the local coordinates of body 2 can be determined via the intersection of two circumferences: one with a center in F_2 and a radius equal to ϑ_3; and the other centered in B_2 whose radius is defined by the constant distance between B_2 and G_2. The intersection between these two circumferences is calculated using the auxiliary parameters e_2, l_2 and h_2:

$$e_2 = \sqrt{\left(\mathrm{F}_2'y - \mathrm{B}_2'y\right)^2 + \left(\mathrm{F}_2'z - \mathrm{B}_2'z\right)^2} \tag{4.9}$$

$$l_2 = \frac{\vartheta_3^{\,2} - \left[\left(\mathrm{G}_3'y - \mathrm{A}_3'y\right)^2 + \left(\mathrm{G}_3'z - \mathrm{A}_3'z\right)^2\right] + e_2^{\,2}}{2e_2^{\,2}} \tag{4.10}$$

$$h_2 = \sqrt{\vartheta_3^{\,2} - l_2^{\,2}} \tag{4.11}$$

The coordinates of G_2 can finally be obtained using the expressions:

$$\mathrm{G}_2'y = \frac{l_2}{e_2}\left(\mathrm{B}_2'y - \mathrm{F}_2'y\right) + \frac{h_2}{e_2}\left(\mathrm{B}_2'z - \mathrm{F}_2'z\right) + \mathrm{F}_2'y \tag{4.12}$$

$$\mathrm{G}_2'z = \frac{l_2}{e_2}\left(\mathrm{B}_2'z - \mathrm{F}_2'z\right) + \frac{h_2}{e_2}\left(\mathrm{B}_2'y - \mathrm{F}_2'y\right) + \mathrm{F}_2'z \tag{4.13}$$

Solving the combined Equations 4.6—4.13 established the desired relation between θ_3 and ϑ_3, completing the analysis of the indirectly actuated joints.

4.2.2. Recursive Algorithm for the Forward Kinematic Analysis

The recursive algorithm employs ZXZ intrinsic Euler angle rotations and cartesian coordinates to automatically model the bodies of serial and branched kinematic chains of CHARMIE (Gonçalves et al., 2022c). The methodology divides rotations and translations of bodies into two different operations to maximize computational efficiency. Each body is modeled following three main operations: i) creating the body in local coordinates utilizing input data, ii) rotating the body into its current orientation, and iii) moving the body into its current position.

The rotations between two consecutive bodies are defined using Euler angles in the following notation:

$$\mathrm{ZXZEuler}(Z_1, X_2, Z_3) = \begin{bmatrix} c_1c_3 - s_1c_2s_3 & -c_1s_3 - s_1c_2c_3 & s_1s_3 \\ s_1c_3 - c_1c_2s_3 & c_1c_2c_3 - s_1s_3 & -c_1s_2 \\ s_2s_3 & s_2c_3 & c_2 \end{bmatrix} \tag{4.14}$$

where c and s represent the cosine and sine functions respectively, and the indices 1, 2, and 3 refer to the three Euler angles.

The rotation matrix between two bodies $\mathrm{i}-1$ and i can then be computed based on the joint properties as follows:

$$\mathbf{R}_{\mathrm{i}}^{\mathrm{i}-1} = \begin{cases} \mathrm{ZXZEuler}\left(0, \vartheta_{\mathrm{i}}, 0\right) & \forall JT_{\mathrm{i}} = R \wedge JO_{\mathrm{i}} = x \\ \mathrm{ZXZEuler}\left(90^{\mathrm{o}}, \vartheta_{\mathrm{i}}, -90\right) & \forall JT_{\mathrm{i}} = R \wedge JO_{\mathrm{i}} = y \\ \mathrm{ZXZEuler}\left(0, 0, \vartheta_{\mathrm{i}}\right) & \forall JT_{\mathrm{i}} = R \wedge JO_{\mathrm{i}} = z \\ \mathrm{ZXZEuler}\left(0, 0, 0\right) & \forall JT_{\mathrm{i}} = P \end{cases} \tag{4.15}$$

This notation defines all rotations between consecutive bodies of CHARMIE except for the fixed joints, which are characterized by the expressions:

$$\mathbf{R}_{3\mathrm{a}}^{3} = \mathrm{ZXZEuler}\left(90^{\mathrm{o}}, -90^{\mathrm{o}}, 180^{\mathrm{o}}\right) \tag{4.16}$$

$$\mathbf{R}_{3\mathrm{b}}^{3} = \mathrm{ZXZEuler}\left(90^{\mathrm{o}}, 90^{\mathrm{o}}, 0\right) \tag{4.17}$$

$$\mathbf{R}_{3\mathrm{c}}^{3} = \mathrm{ZXZEuler}\left(0, 0, 0\right) \tag{4.18}$$

When running the recursive algorithm, the global orientation $\mathbf{R}_{\mathrm{i}}^{0}$ of the current body is determined by the multiplication of rotation matrices considering both the global orientation of the preceding body $\mathbf{R}_{\mathrm{i}-1}^{0}$, and the relative rotation between the two bodies $\mathbf{R}_{\mathrm{i}}^{\mathrm{i}-1}$ as expressed in Equation 4.19.

$$\mathbf{R}_i^0 = \mathbf{R}_{i-1}^0 \mathbf{R}_i^{i-1} \tag{4.19}$$

This rotation matrix is employed to rotate all P_i points of body i around its origin (Point O_i), so that body i attains its current orientation:

$$\mathbf{P}_i'' = \mathbf{R}_i^0 \mathbf{P}_i' \tag{4.20}$$

The final step is moving body i into its current position. Since body $\mathrm{i}-1$ was already assembled in the previous iteration, this condition can be guaranteed by moving point A_i to point $\mathrm{B_{i-1}}$, and then guaranteeing that all points of the body i follow this same path:

$$\mathbf{P}_i = \mathbf{P}_i'' + (\mathbf{B}_{i-1} - \mathbf{A}_i'') \tag{4.21}$$

By successively repeating these steps for all CHARMIE's bodies, the forward kinematics of the serial and branched kinematic chains are solved.

4.2.3. Kinematics of the Auxiliary Bodies of the Closed Loops

Most of the auxiliary bodies of CHARMIE, such as linear actuators or extension springs, have their ends coinciding with two of the bodies which are a part of the main serial kinematic chains. In these cases, all information characterizing the auxiliary bodies is fully generated by the recursive algorithm. When the length l_s of an auxiliary body, whose ends are points S_i and S_j, is required, the following notation can be used:

$$l_s = \sqrt{\left(S_i x - S_j x\right)^2 + \left(S_i y - S_j y\right)^2 + \left(S_i z - S_j z\right)^2} \tag{4.22}$$

Any desired orientation can be obtained using the law of cosines. First, a new point R_i to body i so that the segment from S_i to R_i is aligned with one of the local coordinate axes of body i. Points S_i to R_i and S_j now form a triangle whose sides have lengths of l_s, a_s, and b_s, which can be determined using Equation 4.22. The rotation α_s between the chosen local axes of body i, and the local axes of the auxiliary body can then be calculated using the expression:

$$\alpha_s = \frac{l_s^2 + b_s^2 - a_s^2}{2 l_s b_s} \tag{4.23}$$

where a_s is the side of the triangle opposite to the desired internal angle.

The only situation where Equations 4.22 and 4.23 are not sufficient to fully study a body is in the closed loop formed by bodies 2b-2c-2d and 2e-2f-2g. Since both mechanisms are symmetric, only bodies 2e-2f-2g will be studied (see Figure 4.6).

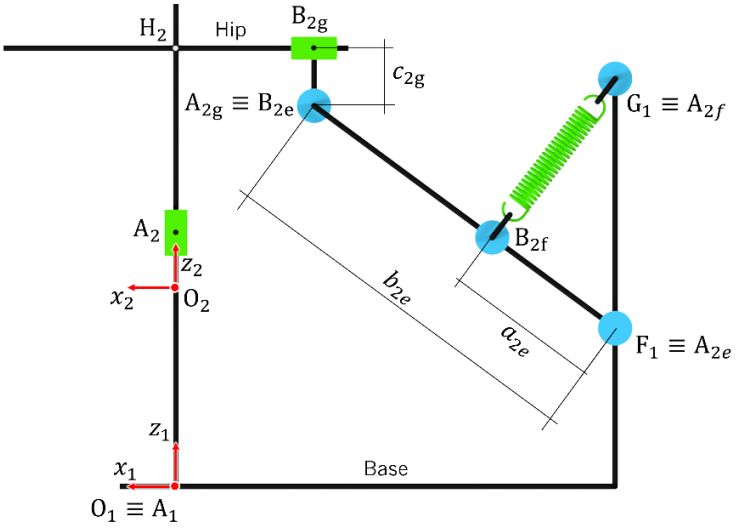


Figure 4.6 – Geometric parameters for the analysis of the indirectly actuated joints of CHARMIE, namely (a) the joint preceding body 2 and (b) the joint preceding body 3.

The goal of this study is to determine the unknown coordinates of points B_{2g}, B_{2e}, and B_{2f}. At this stage, bodies 1 and 2 are fully modeled in global coordinates, so conversions between references are possible. All points will be expressed regarding the robot's base (body 1). The z-coordinate of point B_{2g} can be directly determined from point H_2:

$$B_{2g}^{'1}z = H_2^{'1}z \tag{4.24}$$

The following pair of equations then fully characterize point B_{2e}:

$$B_{2e}^{'1}z = B_{2g}^{'1}z - c_{2g} \tag{4.25}$$

$$B_{2g}^{'1}x = F_1^{'}x - \sqrt{{b_{2e}}^2 - \left(B_{2e}^{'1}z - F_1^{'}z\right)^2} \tag{4.26}$$

The characterization of point B_{2g} can be finished using the following relation:

$$B_{2g}^{'1}x = B_{2e}^{'1}x \tag{4.27}$$

The last point, B_{2f}, can finally be defined using the triangle proportionality theorem:

$$B_{2f}^{'1}x = F_1^{'}x + \frac{a_{2e}}{b_{2e}}\left(B_{2e}^{'1}x - F_1^{'}x\right) \tag{4.28}$$

$$B_{2f}^{'1}z = F_1^{'}z + \frac{a_{2e}}{b_{2e}}\left(B_{2e}^{'1}z - F_1^{'}z\right) \tag{4.29}$$

These three points can then be transformed from the local coordinates of body 1 into the global coordinate reference by following the same rotation and translation that was applied to body 1 during its modeling in the recursive algorithm.

Concluding the analysis of the auxiliary bodies and closed loops, the forward kinematic analysis is fully concluded, and the position of all points for any given robot configuration can be determined. This was utilized to create the multibody model of CHARMIE in the simulation environment shown in Figure 4.7.

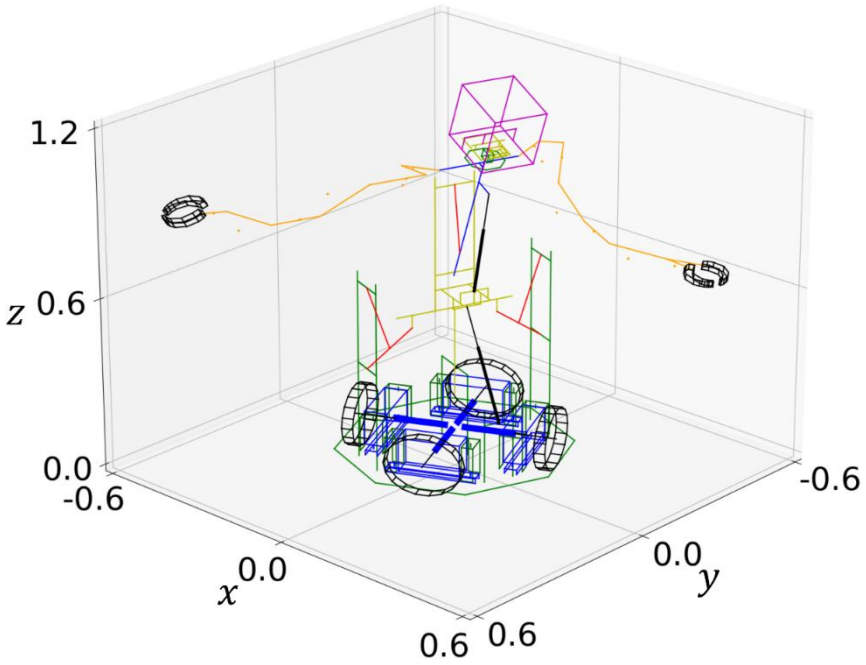


Figure 4.7 – Kinematic model of CHARMIE in the multibody simulation environment

4.3. Inverse Dynamics Analysis

The inverse dynamics analysis aims to determine the forces and torques applied by connected bodies for their known motion. Joint kinematic properties, such as positions, orientations, velocities, and accelerations are used alongside the mass and inertia properties of the bodies to compute the loads transmitted in the joints.

A novel Newton-Euler recursive algorithm based on the formulation of (Siciliano et al., 2009) is implemented to automatically solve the inverse dynamics of serial and branched kinematic chains (Gonçalves et al., 2022b). In the second stage, an additional formulation is derived to analyze the closed and overconstrained loops of CHARMIE. In this subsection, superscripts on all equations denote the coordinate reference a variable is expressed on.

4.3.1. Recursive Newton-Euler Algorithm

Recursive Newton-Euler algorithms are often divided into two stages. In the first stage, forward iterations progress from the robot's base to its end-effectors, determining the velocities and accelerations of key points (joints and center of mass). In the second stage, iterations progress from the robot's end-effectors to its base, determining the loads that preceding bodies must be applying on the current body to generate the current motion.

For this analysis, the unit vector $\mathbf{z}_i$ must first be defined based on the orientation of the joint preceding body i:

$$\mathbf{z}_i = \begin{cases} [1,0,0]^T & \forall JO_i = x \\ [0,1,0]^T & \forall JO_i = y \\ [0,0,1]^T & \forall JO_i = z \end{cases} \tag{4.30}$$

The forward iteration of the recursive algorithm first determines the angular velocity $\boldsymbol{\omega}_i^i$ of the current body:

$$\boldsymbol{\omega}_i^i = \begin{cases} \mathbf{R}_i^{i-1T}\boldsymbol{\omega}_{i-1}^{i-1} & \forall JT_i = P \\ \mathbf{R}_i^{i-1T}\left(\boldsymbol{\omega}_{i-1}^{i-1} + \dot{\theta}_i \mathbf{z}_i\right) & \forall JT_i = R \end{cases} \tag{4.31}$$

where $\mathbf{R}_i^{i-1T}$ is the rotation matrix that converts data expressed in the reference axis of the preceding body to the current body, $\boldsymbol{\omega}_{i-1}^{i-1}$ the angular velocity of the preceding body, and $\dot{\theta}_i$ the angular velocity of the joint preceding body i.

The angular acceleration $\dot{\boldsymbol{\omega}}_i^i$ of the current body is then determined as:

$$\dot{\boldsymbol{\omega}}_i^i = \begin{cases} \mathbf{R}_i^{i-1T}\dot{\boldsymbol{\omega}}_{i-1}^{i-1} & \forall JT_i = P \\ \mathbf{R}_i^{i-1T}\left(\dot{\boldsymbol{\omega}}_{i-1}^{i-1} + \ddot{\theta}_i \mathbf{z}_i + \dot{\theta}_i \dot{\boldsymbol{\omega}}_{i-1}^{i-1} \times \mathbf{z}_i\right) & \forall JT_i = R \end{cases} \tag{4.32}$$

where $\dot{\boldsymbol{\omega}}_{i-1}^{i-1}$ is the angular acceleration of the preceding body, and $\ddot{\theta}_i$ the angular acceleration of the joint preceding body i.

With angular velocities and accelerations determined, the linear acceleration of key points can be calculated. These accelerations include the Coriolis, Euler, and centrifugal effects. The $\ddot{\mathbf{p}}_{Bi}^i$ linear accelerations of points B_i connecting i to succeeding bodies are computed in Equation 4.33.

$$\ddot{\mathbf{p}}_{\mathrm{Bi}}^{\mathrm{i}}=\begin{cases}\mathbf{R}_{\mathrm{i}}^{\mathrm{i-1T}}\left(\ddot{\mathbf{p}}_{\mathrm{Bi-1}}^{\mathrm{i-1}}+\ddot{d}_{\mathrm{i}}\mathbf{z}_{\mathrm{i}}\right)+\dot{\boldsymbol{\omega}}_{\mathrm{i}}^{\mathrm{i}}\times\mathbf{r}_{\mathrm{Ai,Bi}}+\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\left(\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\mathbf{r}_{\mathrm{Ai,Bi}}\right)+2\dot{d}_{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\mathbf{R}_{\mathrm{i}}^{\mathrm{i-1T}}\mathbf{z}_{\mathrm{i}} & \forall JT_{\mathrm{i}}=P\\ \mathbf{R}_{\mathrm{i}}^{\mathrm{i-1T}}\ddot{\mathbf{p}}_{\mathrm{Bi-1}}^{\mathrm{i-1}}+\dot{\boldsymbol{\omega}}_{\mathrm{i}}^{\mathrm{i}}\times\mathbf{r}_{\mathrm{Ai,Bi}}+\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\left(\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\mathbf{r}_{\mathrm{Ai,Bi}}\right) & \forall JT_{\mathrm{i}}=R\end{cases} \tag{4.33}$$

where $\ddot{\mathbf{p}}_{\mathrm{Bi-1}}^{\mathrm{i-1}}$ is the linear acceleration of point A_i in the reference to the preceding body, $\dot{d}_{\mathrm{i}}$ and $\ddot{d}_{\mathrm{i}}$ the linear velocity and acceleration of the joint preceding link i, and $\mathbf{r}_{\mathrm{Ai,Bi}}^{\mathrm{i}}$ is the vector defining the position of B_i.

The final step of the forward iterations is to determine the linear acceleration $\ddot{\mathbf{p}}_{\mathrm{Ci}}^{\mathrm{i}}$ of the center of mass using the expression:

$$\ddot{\mathbf{p}}_{\mathrm{Ci}}^{\mathrm{i}}=\ddot{\mathbf{p}}_{\mathrm{Bi}}^{\mathrm{i}}+\dot{\boldsymbol{\omega}}_{\mathrm{i}}^{\mathrm{i}}\left(\mathbf{r}_{\mathrm{Ai,Ci}}-\mathbf{r}_{\mathrm{Ai,Bi}}\right)+\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\left[\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\times\left(\mathbf{r}_{\mathrm{Ai,Ci}}-\mathbf{r}_{\mathrm{Ai,Bi}}\right)\right] \tag{4.34}$$

where $\mathbf{r}_{\mathrm{Ai,Ci}}^{\mathrm{i}}$ is the vector defining the position of C_i.

The backward iterations start by calculating the sum of the $\mathbf{f}_{\mathrm{j}}^{\mathrm{i}}$ forces applied by all the j preceding bodies on the current body i. If a single body precedes body i, this sum characterizes the forces on the joint between i and $\mathrm{i}-1$:

$$\sum_{\mathrm{j=1}}^{J}\mathbf{f}_{\mathrm{j}}^{\mathrm{i}}=m_{\mathrm{i}}\ddot{\mathbf{p}}_{\mathrm{Ci}}^{\mathrm{i}}+\sum_{\mathrm{k=1}}^{K}\left(\mathbf{R}_{\mathrm{k}}^{\mathrm{i}}\mathbf{f}_{\mathrm{k}}^{\mathrm{k}}\right) \tag{4.35}$$

where j is the index for each preceding link, J the total number of preceding links, k is the index of each succeeding link, K the total number of succeeding links, m_{i} the mass of link i, and $\mathbf{f}_{\mathrm{k}}^{\mathrm{k}}$ the force applied by the succeeding link k on the current link i.

The backward iterations finish by determining the sum of the $\boldsymbol{\mu}_{\mathrm{j}}^{\mathrm{i}}$ torques applied by the j preceding bodies on the current body using the following formulation:

$$\sum_{\mathrm{j=1}}^{J}\boldsymbol{\mu}_{\mathrm{j}}^{\mathrm{i}}=\bar{\mathbf{I}}_{\mathrm{i}}^{\mathrm{i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}}^{\mathrm{i}}+\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\left(\bar{\mathbf{I}}_{\mathrm{i}}^{\mathrm{i}}\boldsymbol{\omega}_{\mathrm{i}}^{\mathrm{i}}\right)+\sum_{\mathrm{j=1}}^{J}\left(-\mathbf{f}_{\mathrm{j}}^{\mathrm{i}}\times\mathbf{r}_{\mathrm{Aj,Ci}}^{\mathrm{i}}\right)+\sum_{\mathrm{k=1}}^{K}\left[\mathbf{R}_{\mathrm{k}}^{\mathrm{i}}\mathbf{f}_{\mathrm{k}}^{\mathrm{k}}\left(\mathbf{r}_{\mathrm{Ai,Ci}}^{\mathrm{i}}-\mathbf{r}_{\mathrm{Ai,Bk}}^{\mathrm{i}}\right)+\mathbf{R}_{\mathrm{k}}^{\mathrm{i}}\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{k}}\right] \tag{4.36}$$

where $\bar{\mathbf{I}}_{\mathrm{i}}^{\mathrm{i}}$ is the inertia matrix of body i, $\mathbf{r}_{\mathrm{Aj,Ci}}^{\mathrm{i}}$ the vector characterizing the position of the preceding joint between body i and j, $\mathbf{r}_{\mathrm{Ai,Bk}}^{\mathrm{i}}$ the vector characterizing the position of the succeeding joint between body i and k, and $\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{k}}$ the torque applied by the succeeding body k on the current body i.

These equations allow the characterization of all forces and torques in the CHARMIE robot except for bodies 2 and 3. These bodies contain more than one preceding link, so additional formulation is required to determine the distribution of loads amongst them.

4.3.2. Analysis of the Closed and Overconstrained Loops

The recursive algorithm automatically solves the inverse dynamics of chained and branched links. However, additional formulations are required to solve overconstrained loops. These loops are present on Charmie's torso and hip mechanisms. The free-body diagrams of these mechanisms are shown in Figure 4.8.

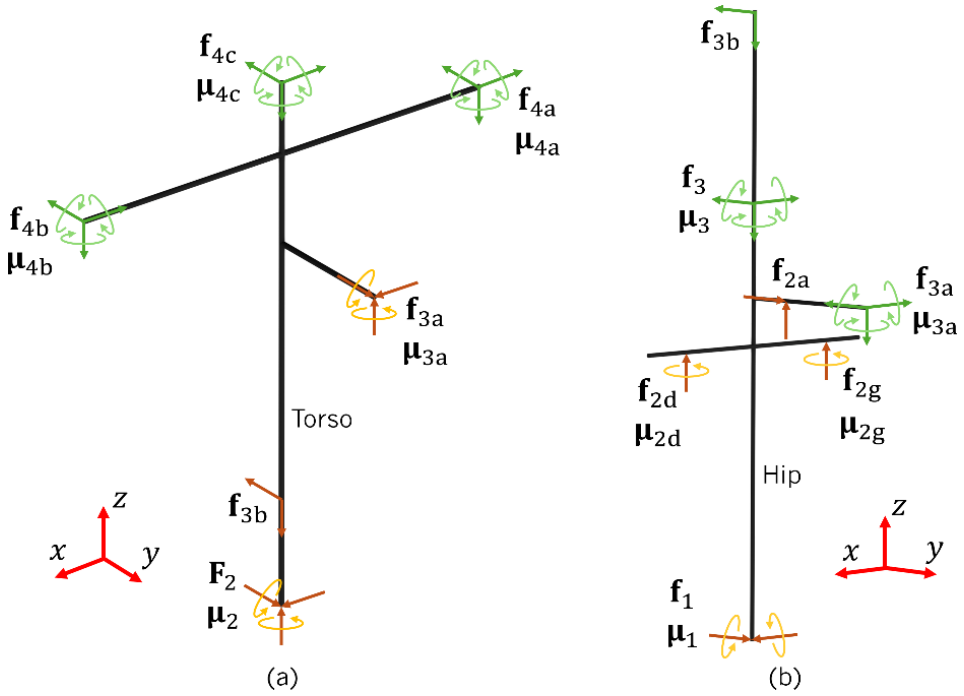


Figure 4.8 – Free body diagram of CHARMIE's torso and hip. Green hues represent forces determined by the recursive algorithm, while orange hues represent calculated forces in the analysis of this subsection.

The backward iterations progress from end-effector to base, so the torso must be studied first.

The forces applied from the tension springs F_{Spring} only depend on their geometry and orientation, which can be determined by pairing Equations 4.22 and 4.23 with the following expression:

$$F_{\text{Spring}} = k_{\text{Spring}} \left(L_0 - L_{\text{Spring}} \right) + F_{0\text{Spring}} \tag{4.37}$$

where k_{Spring} is the spring constant, L_0 the spring's free length, L_0 the current spring length and $F_{0\text{Spring}}$ the force originating from the spring's pre-tension during manufacturing.

The focus of this approach is maximizing the computational efficiency. This was achieved by implementing a set of simplifications to allow the direct calculation of force distributions. Three main considerations were implemented.

- Since the joints from the linear actuator and hip are both revolute around the *x*-axis, their forces must guarantee the *x*-torque equilibrium;
- The force applied by the linear actuator in the *yz* plane must be tangential to the orientation of the actuator;
- The joints from the linear actuator and hip have similar construction and tolerances, therefore each will apply half of the force reaction on the *x*-axis.

These considerations allow solving all forces applied on the robot's torso with the following expression:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & \mathbf{r}^3_{\mathrm{A3a,C3}}\hat{\mathbf{z}} & -\mathbf{r}^3_{\mathrm{A3a,C3}}\hat{\mathbf{y}} & 0 & \mathbf{r}^3_{\mathrm{A2,C3}}\hat{\mathbf{z}} & -\mathbf{r}^3_{\mathrm{A2,C3}}\hat{\mathbf{y}} \\ 0 & \cos(\phi_{3a}) & -\sin(\phi_{3a}) & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f}^3_{3a}\hat{\mathbf{x}} \\ \mathbf{f}^3_{3a}\hat{\mathbf{y}} \\ \mathbf{f}^3_{3a}\hat{\mathbf{z}} \\ \mathbf{f}^3_{2}\hat{\mathbf{x}} \\ \mathbf{f}^3_{2}\hat{\mathbf{y}} \\ \mathbf{f}^3_{2}\hat{\mathbf{z}} \end{bmatrix} \tag{4.38}$$

where $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{z}}$ are the unit vectors related to the three coordinate axes, $\mathbf{r}^3_{\mathrm{A3a,C3}}$ and $\mathbf{r}^3_{\mathrm{A2,C3}}$ are the vectors defining the positions of the preceding joints in relation to the center of mass of the torso, Φ_{3a} is the angle defining the rotation of the linear actuator in relation to the torso around the *x*-axis. The left side of the equation system is defined by the variables:

$$a_1 = \sum_{j=1}^{2} \mathbf{f}^3_{\mathrm{j}}\hat{\mathbf{x}} \tag{4.39}$$

$$a_2 = \sum_{j=1}^{2} \mathbf{f}^3_{\mathrm{j}}\hat{\mathbf{y}} \tag{4.40}$$

$$a_3 = \sum_{j=1}^{2} \mathbf{f}^3_{\mathrm{j}}\hat{\mathbf{z}} \tag{4.41}$$

$$a_4 = \bar{\mathbf{I}}^3_3\dot{\boldsymbol{\omega}}^3_3\hat{\mathbf{x}} + \boldsymbol{\omega}^3_3\left(\bar{\mathbf{I}}^3_3\boldsymbol{\omega}^3_3\right)\hat{\mathbf{x}} + \left(-\mathbf{f}^3_{3b}\times\mathbf{r}^3_{\mathrm{A3b,C3}}\right)\hat{\mathbf{x}} + \sum_{k=1}^{3}\left[\mathbf{R}^5_k\mathbf{f}^k_k\left(\mathbf{r}^3_{\mathrm{A3,C3}} - \mathbf{r}^3_{\mathrm{A3,Bk}}\right) + \mathbf{R}^3_k\boldsymbol{\mu}^k_k\right]\hat{\mathbf{x}} \tag{4.42}$$

The variables a_1, a_2, and a_3 represent the sums of unknown forces along the x, y and z axes, while a_4 represents the sum of torques around the *x*-axis without considering the forces from the linear actuator and the hip.

For the determination of unknown torques, only one consideration is required:

- The similar construction of the joints from the linear actuator and hip results in both applying half of the reaction torques around the *y* and *z* axes.

The unknown torques of the robot's torso can be determined with the expression:

$$\begin{bmatrix} \boldsymbol{\mu}_{3a}^{3}\hat{\mathbf{y}}+\boldsymbol{\mu}_{2}^{3}\hat{\mathbf{y}} \\ \boldsymbol{\mu}_{3a}^{3}\hat{\mathbf{z}}+\boldsymbol{\mu}_{2}^{3}\hat{\mathbf{z}} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu}_{3a}^{3}\hat{\mathbf{y}} \\ \boldsymbol{\mu}_{3a}^{3}\hat{\mathbf{z}} \\ \boldsymbol{\mu}_{2}^{3}\hat{\mathbf{y}} \\ \boldsymbol{\mu}_{2}^{3}\hat{\mathbf{y}} \end{bmatrix} \tag{4.43}$$

where the left side of the equation contains the sum of unknown torques which was determined by Equation 4.36.

The hip of CHARMIE has four sets of unknown forces and torques applied by the linear actuator, the robot's base, and the two symmetrical halves of the static balancing mechanism.

The forces applied by the static balancing mechanism will only depend on the geometrical configuration of the spring (see Figure 4.9).

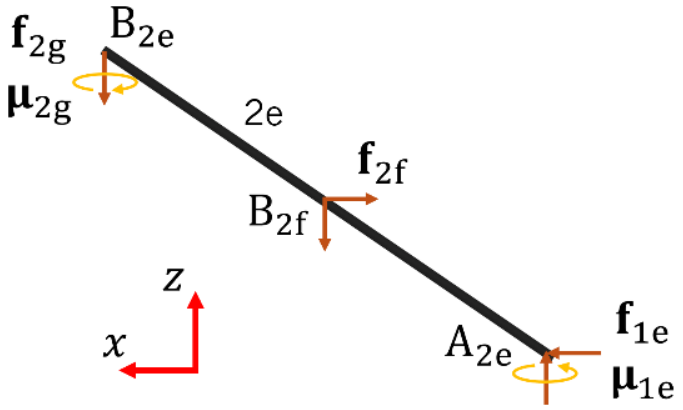


Figure 4.9 – Free body diagram of the static balancing mechanism bar for the squatting motion

The spring force $\mathbf{f}_{2f}$ is determined from the spring geometry and orientation. For simplification, this bar was considered as massless. The unknown forces can then be determined from the force and torque balance:

$$\begin{bmatrix} \mathbf{f}_{2f}^{2}\hat{\mathbf{x}} \\ \mathbf{f}_{2f}^{2}\hat{\mathbf{z}} \\ (\mathbf{f}_{2f}^{2}\hat{\mathbf{z}})(\mathbf{r}_{A2e,B2e}^{2}\hat{\mathbf{x}})-(\mathbf{f}_{2f}^{2}\hat{\mathbf{x}})(\mathbf{r}_{A2e,B2e}^{2}\hat{\mathbf{z}}) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \\ -\mathbf{r}_{A2e,B2f}^{2}\hat{\mathbf{x}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f}_{2g}^{2}\hat{\mathbf{z}} \\ \mathbf{f}_{1e}^{2}\hat{\mathbf{x}} \\ \mathbf{f}_{1e}^{2}\hat{\mathbf{z}} \end{bmatrix} \tag{4.44}$$

The remaining forces applied on the hip required the following considerations:

- The force applied by the linear actuator in the *yz* plane must be tangential to the orientation of the actuator;
- The joint between the base and the hip has much tighter tolerances than the static balancing mechanism, so it was estimated that this joint applies all undetermined reaction forces in the *x* and *y* axes.

With these simplifications, all the unknown forces can be determined with the expression:

$$\begin{bmatrix} \mathbf{f}_1^2\hat{\mathbf{x}}+\mathbf{f}_{2a}^2\hat{\mathbf{x}} \\ \mathbf{f}_1^2\hat{\mathbf{y}}+\mathbf{f}_{2a}^2\hat{\mathbf{y}} \\ \mathbf{f}_1^2\hat{\mathbf{z}}+\mathbf{f}_{2a}^2\hat{\mathbf{z}}+\mathbf{f}_{2d}^2\hat{\mathbf{z}}+\mathbf{f}_{2g}^2\hat{\mathbf{z}} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ \cos(\phi_{2a}) & -\sin(\phi_{2a}) & 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{f}_{2a}^2\hat{\mathbf{y}} \\ \mathbf{f}_{2a}^2\hat{\mathbf{z}} \\ \mathbf{f}_1^2\hat{\mathbf{x}} \\ \mathbf{f}_1^2\hat{\mathbf{y}} \end{bmatrix} \tag{4.45}$$

where Φ_{2a} is the angle defining the rotation of the linear actuator concerning the hip around the *x*-axis and the sum of forces on the left side of the equation system can be determined with Equation 4.36.

The final step for the inverse dynamic analysis is studying the distribution of torques in the hip. The following simplifications were considered:

- The joint between the base and the hip has much tighter tolerances than the static balancing mechanism, so it was estimated that this joint applies all undetermined reaction torques in the *x* and *y* axes.
- The joint between the base and the hip cannot apply torque around the *z*-axis, and the static balancing mechanism joints have tighter tolerance than the linear actuator, so each will apply half of the torque reaction in the *z*-axis.

This results in the following system of equations:

$$\begin{bmatrix} \boldsymbol{\mu}_1^2\hat{\mathbf{x}} \\ \boldsymbol{\mu}_1^2\hat{\mathbf{y}} \\ \boldsymbol{\mu}_{2d}^2\hat{\mathbf{z}}+\boldsymbol{\mu}_{2g}^2\hat{\mathbf{z}} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu}_1^2\hat{\mathbf{x}} \\ \boldsymbol{\mu}_1^2\hat{\mathbf{y}} \\ \boldsymbol{\mu}_{2d}^2\hat{\mathbf{z}} \\ \boldsymbol{\mu}_{2g}^2\hat{\mathbf{z}} \end{bmatrix} \tag{4.46}$$

where the left side of the equation is defined by using Equation 4.36.

The implemented dynamic analysis was validated by comparing the results obtained from it with those generated by commercial software. The same motion was implemented in both simulations as

described in (Gonçalves et al., 2022a). The compared values were for the force exerted by the floor plane on the robot's base, which requires the full execution of both recursive algorithms. The results from Figure 4.10 validate the present methodology, presenting only slight variations on the reaction torques due to a change in the application point of some of the generated forces between the two considered models.

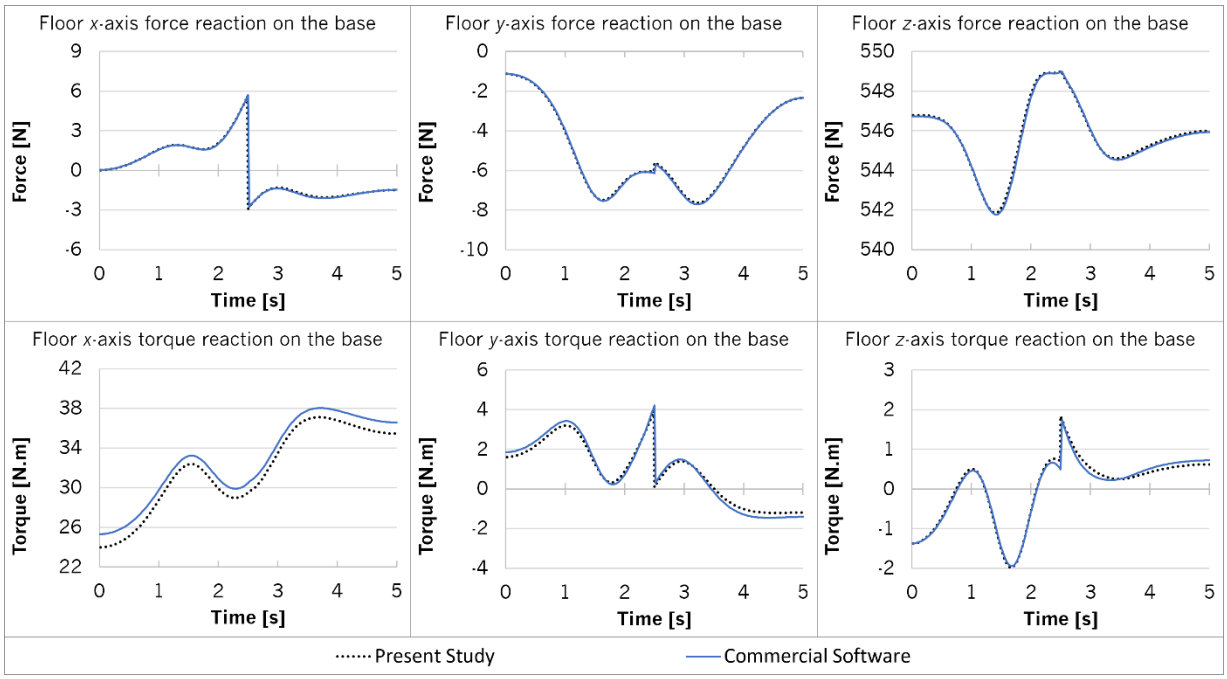


Figure 4.10 – Forces and torques exerted by the floor on CHARMIE's base on the simulation of a 5-second determined using the method described in this thesis, as well as from a commercial software for comparison.

4.4. Robot-Environment Interaction

With the multibody model of CHARMIE created, the simulator can be completed by modeling how the robot interacts with the surrounding environment. There are two relevant types of environmental interaction: locomotion analysis, which aims at employing forward dynamics to determine the motion of CHARMIE around the workspace; and object interaction analysis, which employs kinematic data to study the collision with obstacles and the interaction of the manipulation system with objects of manipulation.

4.4.1. Locomotion Analysis

The locomotion analysis mixes inverse dynamics with forward dynamics to analyze how the robot moves along the workspace. This process follows four steps: 1) determine the forces and torques from the suspension system, 2) determine the forces and torques from the motors of the locomotive system

and reaction of the floor, 3) analyze the sum of forces and torques implemented in the robot's base, 4) determine the acceleration of the robot's bases, and use them to update the velocities and positions of the robot.

The suspension system is passive. An independent suspension system is employed; therefore, each system was modeled separately (see Figure 4.11).

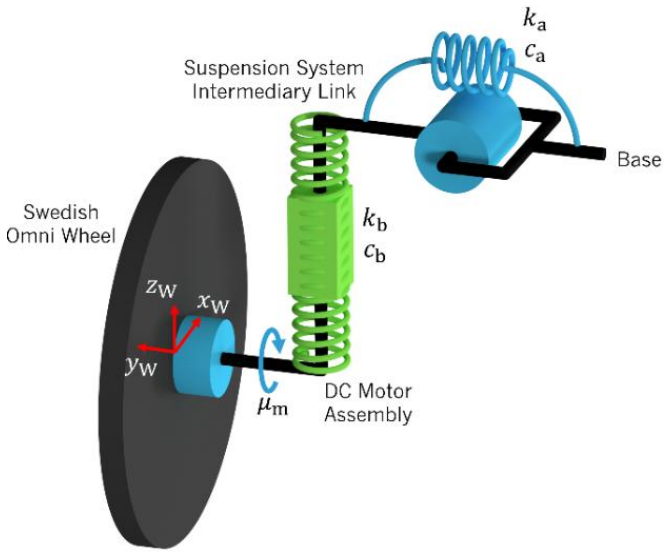


Figure 4.11 – Schematic representation of CHARMIE's independent suspension system.

There are two components for each independent suspension. The linear suspension k_b is due to the installed compression spring, while k_a refers to the deformation of the robot's base, which allows for slight adjustments from the robot's base. A damping effect was added to both suspension elements to better mimic the real behavior of the robot. The force and torque exerted by each part of the suspension system can be determined by the equations:

$$F_{\text{susp}} = \left(l_0 - l\right)k_{\text{b}} + \dot{l}c_{\text{b}} \tag{4.47}$$

$$\mu_{\text{susp}} = \left(\theta_0 - \theta\right)k_{\text{a}} + \dot{\theta}c_{\text{a}} \tag{4.48}$$

where the variables k and c refer to the suspensions' spring constant and damping coefficients, and l and θ to the position and orientation of the suspension.

Swedish omni wheels were used for the robot's locomotion, which only applied active forces in their direction of rotation. The floor plane will apply 2 types of forces on each of the robot's wheels: the vertical reaction force that supports the robot's mass, whose distribution between each of the wheels is determined by the suspension system; and the locomotion force generated from the motor's torque. The $\mathbf{f}_{\text{w}}^{\text{w}}$ force generated by the wheel floor interaction is then determined by Equation 4.49.

$$\mathbf{f}_{\mathrm{w}}^{\mathrm{w}} = F_{\mathrm{susp}}\hat{\mathbf{z}} + \mu_{\mathrm{m}}\frac{d_{\mathrm{wheel}}}{2}\hat{\mathbf{x}} \tag{4.49}$$

where F_{susp} is the force of the associated suspension, μ_m the torque from the motor, d_{wheel} the wheel diameter.

The motion of the robot's base was simplified by grouping the mass and inertia from the base, suspension system, wheels, and motor into a single body 1t with the properties described in Table 4.6 and Table 4.7.

Table 4.6 – Mass properties of CHARMIE's base alongside the suspension and locomotion systems.

Body i	**Body Name**	**Mass** $[kg]$	$r_{\mathrm{Oi,Ci}}$ $[mm]$		
			x	y	z
1t	Base + Suspension & Locomotion Systems	45.24	−0.8	13.3	110.7

Table 4.7 – Inertial properties of CHARMIE's base alongside the suspension and locomotion systems.

Body i	**Inertia** $[kg.mm^2]$					
	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
1t	1.6×10^6	1.9×10^6	1.8×10^6	3.4×10^3	4.0×10^3	-2.6×10^5

The sum of forces implemented on the robot's base considers the forces from the wheels, those which are applied by the upper body, and the effect of gravity on the robot's base, which are expressed as:

$$\mathbf{f}_{\mathrm{lt}}^{\mathrm{lt}} = \sum_{\mathrm{w}=1}^{4}\mathbf{R}_{\mathrm{w}}^{\mathrm{lt}}\mathbf{f}_{\mathrm{w}}^{\mathrm{w}} + \sum_{\mathrm{k}=1}^{6}\mathbf{R}_{\mathrm{k}}^{\mathrm{lt}}\mathbf{f}_{\mathrm{k}}^{\mathrm{k}} - gm_{\mathrm{lt}}\hat{\mathbf{z}} \tag{4.50}$$

The sum of torques considers three effects, the torques generated from the wheel forces, the forces and torques applied by the upper body, and the torques generated by the revolute suspension.

$$\boldsymbol{\mu}_{\mathrm{lt}}^{\mathrm{lt}} = \sum_{\mathrm{w}=1}^{4}\mathbf{R}_{\mathrm{w}}^{\mathrm{lt}}\mathbf{f}_{\mathrm{w}}^{\mathrm{w}} \times \mathbf{r}_{\mathrm{Aw,Clt}}^{\mathrm{lt}} + \sum_{\mathrm{k}=1}^{6}\mathbf{R}_{\mathrm{k}}^{\mathrm{lt}}\mathbf{f}_{\mathrm{k}}^{\mathrm{k}} \times \left(\mathbf{r}_{\mathrm{Alt,Clt}}^{\mathrm{lt}} - \mathbf{r}_{\mathrm{Alt,Bk}}^{\mathrm{lt}}\right) + \sum_{\mathrm{k}=1}^{6}\mathbf{R}_{\mathrm{k}}^{\mathrm{lt}}\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{k}} + \sum_{\mathrm{j}=1}^{4}\mathbf{R}_{\mathrm{j}}^{\mathrm{lt}}\boldsymbol{\mu}_{\mathrm{j}}^{\mathrm{j}} \tag{4.51}$$

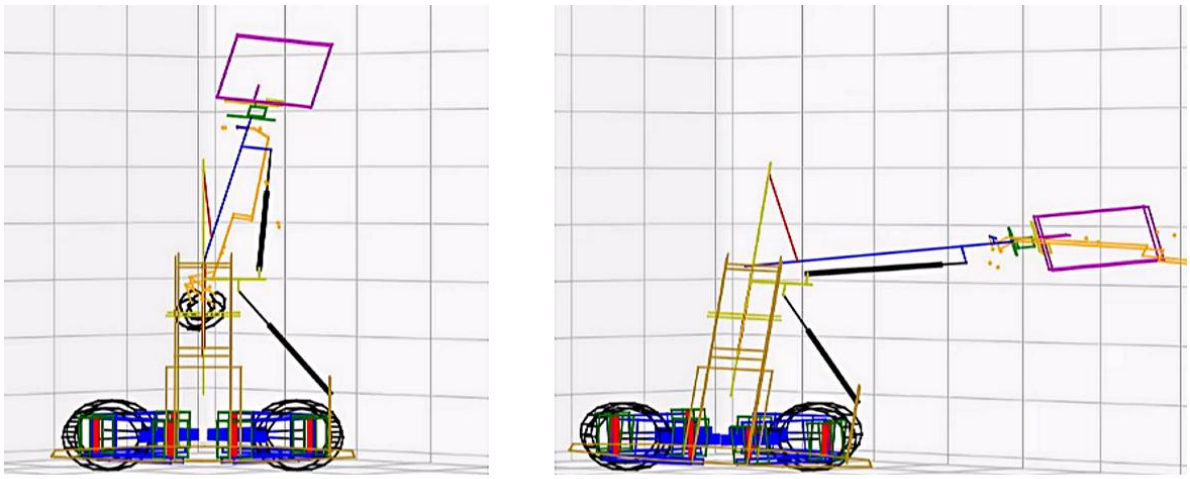
The accelerations of the robot's base are calculated using the algebraic equations of motion to solve the forward dynamics of a body expressed as:

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_{[3\times3]} \\ \mathbf{0}_{[3\times3]} & \bar{\mathbf{I}} \end{bmatrix}\begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \boldsymbol{\mu} \end{bmatrix} - \begin{bmatrix} \mathbf{0}_{[3\times1]} \\ \boldsymbol{\omega}\times\bar{\mathbf{I}}\boldsymbol{\omega} \end{bmatrix} \tag{4.52}$$

where $\mathbf{I}_3$ is a [3x3] identity matrix.

This determines all linear and angular acceleration of the robot's base, which in turn can be used alongside the Euler integration method to determine the evolution of velocities and positions along the simulation. These values are then used to update the position and velocities of the suspension and wheels, determining the input parameters required for the analysis of the inverse and forward dynamics of the next timestep.

The results from the methodology used were tested in the simulation. To test the suspension system's response to the upper body motion, the spring constant of the suspension system was lowered, and then the upper body was flexed. Figure 4.12 shows the right side of the suspension compressing (on side b) as desired due to the shifting position of the robot's center of mass, which results in the rotation of the robot's base.



(a) (b)

Figure 4.12 – Response of CHARMIE's suspension system to the motion of the upper body. The suspension's spring constant was lower for visual effect.

The simulation of CHARMIE's locomotion was tested by using two scenarios whose results could be determined analytically. From a dynamic analysis, the acceleration of a simplified omnidirectional locomotion system can be determined using Equation 4.53.

$$\begin{bmatrix} \ddot{p}_{\mathrm{x}} \\ \ddot{p}_{\mathrm{y}} \end{bmatrix} = \frac{2}{d_{\mathrm{wheel}} m_{\mathrm{total}}} \begin{bmatrix} \sin(45^{\mathrm{o}} - \theta_{\mathrm{z}}) & \cos(45^{\mathrm{o}} - \theta_{\mathrm{z}}) \\ \sin(-45^{\mathrm{o}} - \theta_{\mathrm{z}}) & \cos(-45^{\mathrm{o}} - \theta_{\mathrm{z}}) \\ \sin(-135^{\mathrm{o}} - \theta_{\mathrm{z}}) & \cos(-135^{\mathrm{o}} - \theta_{\mathrm{z}}) \\ \sin(135^{\mathrm{o}} - \theta_{\mathrm{z}}) & \cos(135^{\mathrm{o}} - \theta_{\mathrm{z}}) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \mu_{\mathrm{m1}} \\ \mu_{\mathrm{m2}} \\ \mu_{\mathrm{m3}} \\ \mu_{\mathrm{m4}} \end{bmatrix} \tag{4.53}$$

where $\ddot{p}_x$ and $\ddot{p}_y$ are the robot's lateral and forward acceleration respectively considering its local axis, θ_z is the orientation of the robot concerning the global axis, d_{wheel} the diameter of the omni wheels,

m_{total} the total mass of the robot and μ_{m1}, μ_{m2}, μ_{m3}, $\mu_{\text{m4}}$the accelerations of the four DC motors expressed in motor torques.

In the first test, Figure 4.13(a), motors 1 and 2 were activated with a torque of 2.54 N.m, while motors 3 and 4 with a torque of -2.54 N.m. By implementing Equation 4.53, it is possible to determine that this results in a linear acceleration of 1 m/s² in the positive y-axis, which in turn produces a 0.5 m displacement of the robot after 1 s of motion. The simulation followed this behavior exactly as intended. In the second test, Figure 4.13(b), only motors 1 and 3 were activated with a torque of 3.59 N.m and -3.59 N.m respectively for a diagonal acceleration of 1 m/s². In this second study, there was a slight deviation from the ideal trajectory. This is an intended effect since the robot's center of mass is not centered on its base, the activation of only two wheels causes the base to rotate, varying the direction in which the robot is accelerating regarding the global axis reference. This further highlights the need for robust control solutions in robotic locomotion systems.

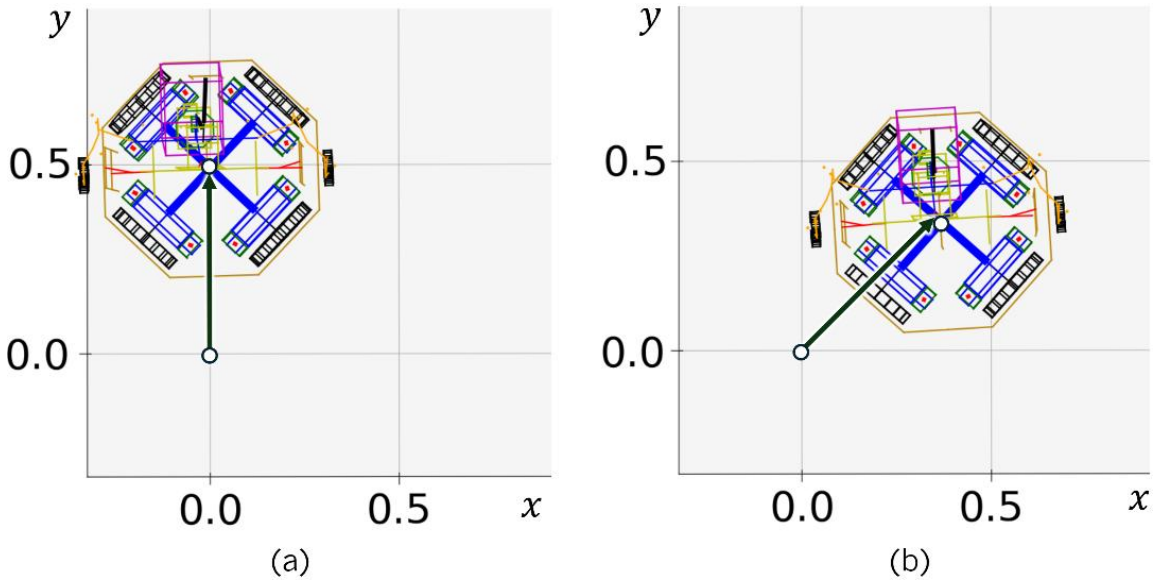


Figure 4.13 – Forward (a) and diagonal (b) motion of CHARMIE's base simulating the dynamics of the locomotion system. The arrow represents the ideal intended motion, while the white circles represent the starting and finishing positions of the robot.

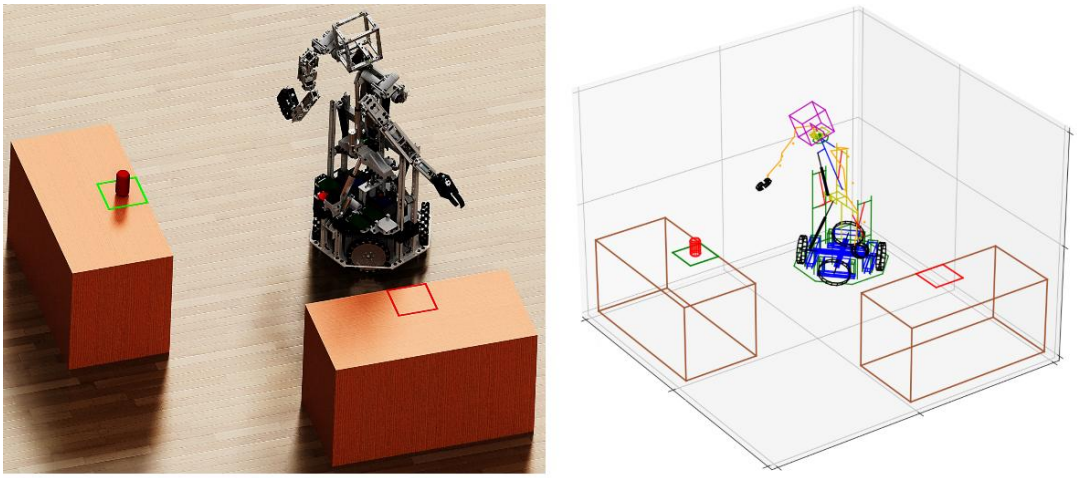
The results from the dynamic simulation of the locomotion and suspension systems were validated for the current study. For further studies on locomotion control, more effects should be considered, such as the drag forces, the inertia of the robot's wheel, and the internal inertia of the DC motors, amongst others. However, the in-depth analysis of the locomotion system greatly increases the computational resources of the simulation and does not produce any immediately relevant results for the trajectory control.

In the current iteration of the simulator for the neural network study, the analysis of the suspension and locomotion system was not considered. The suspension was considered as a rigid body, while the locomotion was simplified to its end results, a translation along the x and y global axes, and a rotation around the z-axis. In a future study, the desired output motion of the robot should be converted into the required motor commands to allow the direct implementation of the neural network.

4.4.2. Object Interaction

The object interaction analysis tackles two situations, the detection of collisions between bodies, and the robot's manipulation of items; in the simulation study, the manipulated object will only be a standard 33 cl beverage metal can.

In the simulation, the robot will never collide with itself due to the established joint constraints. The task established for the robot will be to transport a metal can from one counter to another (see Figure 4.14). The three possible obstacles of collision are the two counters and the can itself.



(a) (b)

Figure 4.14 – (a) 3D CAD model and (b) virtual simulation of CHARMIE in its work environment

Collisions are detected by determining if a certain point of the robot enters the boundaries defined by the object's limits (a parallelepiped for the counters, and a cylinder for the can). To deal with the lack of representation of some geometrical details in the computational model, a threshold range T_l is associated with the bodies, if a point plus this threshold enters the boundaries of the object, then there is a collision. For the counter, collisions C_1 for point P are defined as:

$$\left(x_{\min} - T_l < \mathrm{P}_i x < x_{\max} + T_l\right) \wedge \left(y_{\min} - T_l < \mathrm{P}_i y < y_{\max} + T_l\right) \wedge \left(\mathrm{P}_i z < z_{\max} + T_l\right) \Rightarrow C_1 = 1 \quad (4.54)$$

where $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$, and $z_{\max}$ define the limits of the counter.

For the can, it was considered that non-manipulated cans will always be in the upright position and lying on a surface. Therefore, unwanted collisions C_{Can} regarding a can with a center point $\mathrm{C_c}$ and radius r_{c} can be determined from the expression:

$$\left(\sqrt{\left(\mathrm{P_i}x-\mathrm{C_c}x\right)^2\left(\mathrm{P_i}y-\mathrm{C_c}y\right)^2}<r_{\mathrm{c}}+T_{\mathrm{l}}\right)\wedge\left(\mathrm{P_i}z<z_{\mathrm{max\,C}}+T_{\mathrm{l}}\right)\Rightarrow C_{\mathrm{can}}=1 \tag{4.55}$$

where z_{maxC} is the point of the can with the highest z-coordinate value.

The verification of the collision between all points of the robot with all obstacles would require computational resources, and reduce the simulator's effectiveness, especially as simulations get more complex. This limitation can be mitigated by introducing level-of-detail representations. Level-of-detail is a common strategy in computer graphics, where meshes of complex geometric environments are progressively refined as needed, the goal is utilizing the "worst possible" representation which fulfills all requirements to significantly reduce computational loads (Hoppe, 1997).

For CHARMIE, the robot was defined as a single point for the coarsest level of detail. Once this point gets close enough to an object so that a collision is possible, CHAMIE is split into five points representing five major systems, base, torso, head, left arm, and right arm. Once one of these systems is close enough to an obstacle where a collision could be possible, this point can then be split into further points or achieve the level of detail of a full model. At most timesteps of the simulation, CHARMIE will only be close to one obstacle, and use only some of its parts to interact, so this greatly reduces the computational effort of collision detection. The levels of detail used in CHARMIE are shown in Figure 4.15.

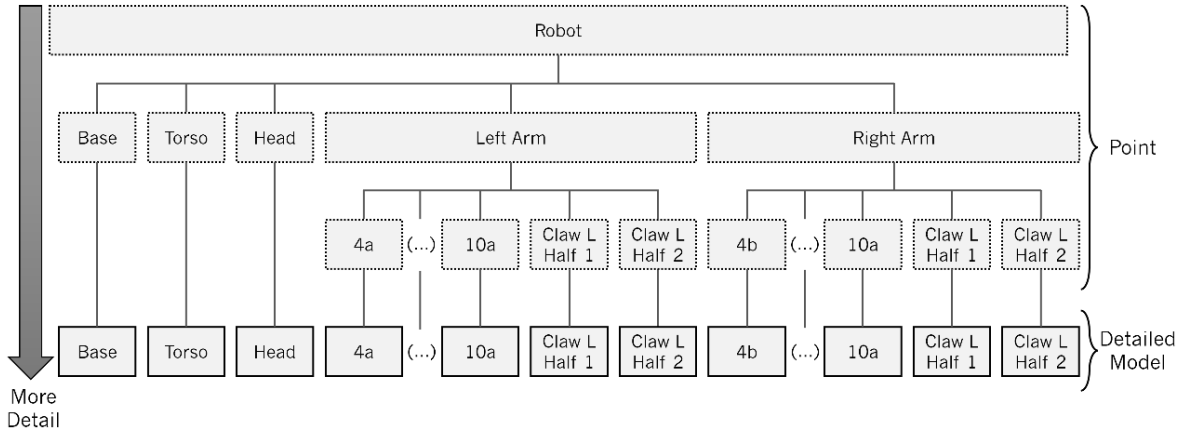


Figure 4.15 – Progression of the level of detail of CHARMIE's model for collision detection

A visual representation of CHARMIE's increasing level of detail is shown in Figure 4.16. The robot starts as a single point, and as it approaches an obstacle, its model gets progressively more refined until collision is detected.

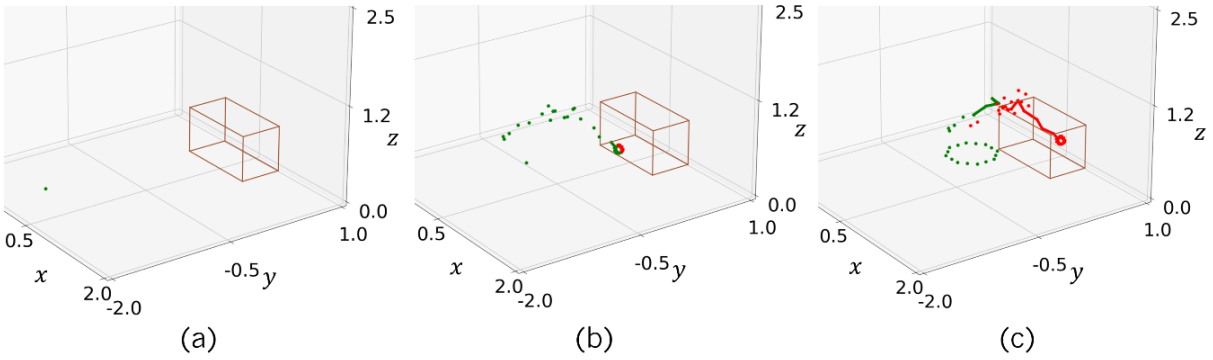


Figure 4.16 – Increase in CHARMIE's level of detail in the computer simulation as it approaches an obstacle. Bodies colored in red are colliding with the obstacle.

The second type of object interaction is the intentional manipulation of the metal can. The goal is transporting the metal can from its starting point to a target area in a second counter.

The most versatile option is to model the forward dynamics of the can, and its interaction with the claw and other objects in the simulation environment. This option increases the computational resources required for the simulation. Since only a single task will be modeled, the behavior of the can was modeled as a finite state machine (see Figure 4.17).

When one of the claws reaches the position of the can, it can close. If the claw closes in a valid position, the can is grabbed: its movement starts accompanying the trajectory of the claw. The can is treated as a part of the robot, and its collision with the counters is studied as any component of the robot. At this point, the robot must transport the can to its target position. When the target position is reached, after the claw is opened, the can is dropped and is back at a resting state.

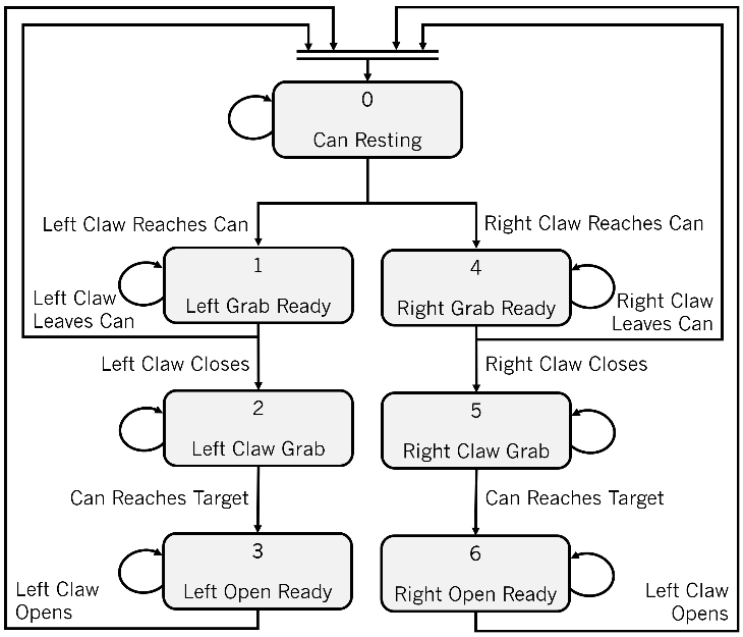


Figure 4.17 – Finite state machine describing the state of the metal can in the simulation environment

4.5. Summary and Discussion

In this chapter, the multibody model of CHARMIE was fully developed and implemented into a simulation environment in a Python IDE. The model fully describes CHARMIE's kinematics and dynamics, allows its navigation along the workspace, and detects object collision and interaction.

The first step for this analysis was the complete characterization of CHARMIE's bodies and joints in a coherent notation. The physical properties of each body were obtained from a detailed CAD model.

The forward kinematic analysis, alongside the inverse dynamic analysis, modeled the full behavior of the robot itself. Recursive algorithms that can tackle branched kinematic chains were used for both studies, while closed and overconstrained loops required further formulations. The kinematic component of the study performs a key role in the neural network trajectory study of CHARMIE, while the dynamic analysis provided the forces and torques required to optimize the robot's mechanisms and geometries. The results from these studies were fully validated by comparing them with commercial software.

The final step of studying the multibody model was defining its interactions with the surrounding environment. It is possible to implement complex forward dynamics analysis for the navigation of CHARMIE, but for the neural network solution, locomotion will be simplified into a "sliding" motion on top of the floor plane. CHARMIE can interact with three objects, two counters, and a can; the code can detect collisions with these objects with high computational efficiency, as well as model the behavior of the metal can being manipulated as a finite state machine.

The result of this methodology is a highly versatile, adaptable, and computationally efficient multibody model. This model played a key role in optimizing CHARMIE's mechanisms and provided the groundwork for starting the reinforcement learning trajectory control development.

"Tell me and I forget, teach me and I may remember, involve me and I learn."
Benjamin Franklin

5. REINFORCEMENT LEARNING TRAJECTORY CONTROL

The development of neural network control solutions, particularly those involving Reinforcement Learning and Deep Learning, is supporting significant progress in the motion control of robotic systems. These advances are even more significant in unstructured environments, which require adapting to new tasks and constraints (Liu et al., 2020).

The learning process of neural networks requires a period of experimentation for the control system to adapt to the required task. A way to remove risks associated with real-world exploration — such as potential damage to the robot or its environment— is to use model-based approaches (Kober et al., 2013). These methods implement the neural network in a simulated computer environment, allowing a risk-free exploration of the robot along thousands of work iterations to learn how to optimize its behavior. Current reinforcement learning approaches for robotics can be categorized into three main groups: Classical Machine Learning, Optimal Value Reinforcement Learning (such as Q-Learning), and Policy gradient Reinforcement learning (Zhou et al., 2022). Of all available methods, Proximal Policy Optimization (PPO) generates good results with the fastest convergence speed, becoming preferable for robotics optimization problems.

The trajectory control of CHARMIE will employ a PPO neural network to control the acceleration of its joints. This will be achieved by first defining the environment the neural network will operate. Thus, the neural network architecture can be defined, which in turn will be used for training the agent to solve CHARMIE's trajectory control. After proof of concept is obtained, and the neural network can control the robot's trajectory without any user input, the results are analyzed to tune the hyperparameters, allowing for faster training that produces more accurate results.

5.1. Proximal Policy Optimization Neural Network

Proximal Policy Optimization (PPO) is a subset of Policy Gradient Reinforcement Learning Neural Networks. A policy in the context of Reinforcement Learning refers to a strategy that determines actions that should be taken based on the current perceived state of the system. Policies can either be deterministic, where the same action is taken for each state, or stochastic, where actions are chosen based on a probability distribution. Stochastic policies are particularly useful in environments where exploration and handling of uncertainty are important. The overarching objective of any RL framework is to learn an optimal policy — one that yields the highest expected cumulative reward over time through continuous interaction with the environment.

Policy Gradient algorithms seek to maximize the reward obtained by the neural network in the environment by following the gradient of the expected reward in regard to defined policy parameters (Heess et al., 2015). This optimization is done iteratively, adjusting the neural network parameters in the direction that most increases the expected reward. Policy gradient algorithms employ actor-critic architectures, the actor (policy) learns via feedback received from the critic (value-function), see Figure 5.1. The policy and value functions are commonly modeled by two separate neural networks. The value function of the critic is learned from the rewards provided by the reward function, and it seeks to estimate the cumulative future rewards that an agent can obtain (Arulkumaran et al., 2017). The goal of the training will be to minimize the Temporal Difference (TD) error, which measures the difference between the predicted outcome from the actor-critic compared to the actual outcome.

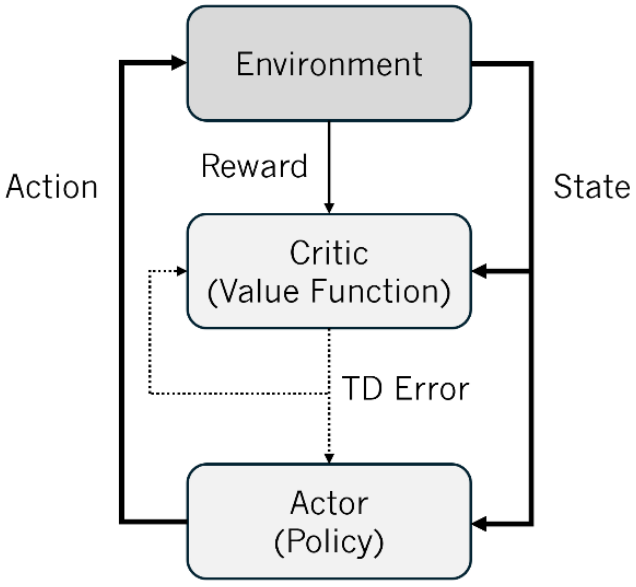


Figure 5.1 – Actor-critic method, adapted from (Arulkumaran et al., 2017)

PPO agents are simple to implement, yet they are considered some of the most robust and stable reinforcement learning algorithms. First presented by (Schulman et al., 2017), PPOs have proved their capabilities in robotics applications by converging with fast and consistent performance (Koch et al., 2018; Zhou et al., 2022), outperforming other options such as the Trust Region Policy Optimization (TRPO) and Deep Deterministic Policy Gradient approaches. PPO algorithms can maintain stable learning without computing second-order derivatives or complex trust regions, which boosts their computational efficiency.

The main distinguishing feature of PPO is the use of clipped surrogate objective functions (Arulkumaran et al., 2017). The goal is to take a conservative approach to policy updates by implementing a pessimistic bound. A hyperparameter ε is introduced to define an interval, the probability ratio between the new and old policy is penalized from leaving this interval. This approach avoids the downsides and losses of performance associated with destructive large updates without the need for high mathematical complexity.

The PPO algorithm learns from data generated by the current policy, which encourages consistent updates, but requires more samples compared to off-policy methods. In PPO the critic minimizes the mean squared error between predicted and actual returns, while the actor seeks to improve the clipped surrogate objective.

Despite its popularity, PPO has its limitations. The clipping mechanism may promote premature convergence or underfitting, particularly in sparse-reward environments such as the control of CHARMIE's motion in a continuous space environment. Furthermore, like in other neural network solutions, the choice of hyperparameters, such as the clipping range ε and learning rate, will always pose a challenge for the implementation of real-world scenarios without empirical tuning. Without adequate care, PPOs can be subjected to learning degradation or catastrophic forgetting along the training process. The PPO's success will also depend on the design of the simulation environment.

In CHARMIE, the implementation of the neural network in the computational simulation was facilitated since the simulator was exclusively computed in Python. The PPO algorithm was imported from the Stable Baselines3 open-source library. The neural network model could then be directly updated and altered in terms of architecture and hyperparameters and trained on environments created using the Gymnasium library.

5.2. Environment Definition

The first stage in training a neural network is defining its work environment; the environment can be defined independently of the NN type that will be used. Defining an environment includes four steps: i) creating the work environment; ii) determining the state space and action space describing the current state and the system's evolution; iii) determining the observation space (inputs of the neural network); and iv) establishing the reward function to evaluate the performance of the neural network. The environment for the neural network simulation was implemented in the Gymnasium open-source Python library.

The work environment of the robot is centered around the multibody simulation described in section 4.4.2. The first approach towards neural network control was to include every single actuator of the robot. However, preliminary tests showed that vast computational resources would be required to tackle problems with such high amounts of input and output variables. If the neural network is divided into two problems, one which only tackles locomotion, and the other which only tackles upper body motion and object manipulation, each problem becomes much more manageable even with limited resources.

For proof of concept of the validity of neural network control in CHARMIE's trajectory, a neural network (and consequently, work environment) was developed that focuses on the motion of one of the robot's arms, its torso motion, and base rotation. Object manipulation without locomotion can be obtained in the layout described in Figure 5.2. The goal is for the robot to grab the can placed in the center of the green area and move it toward the area marked red on a second counter.

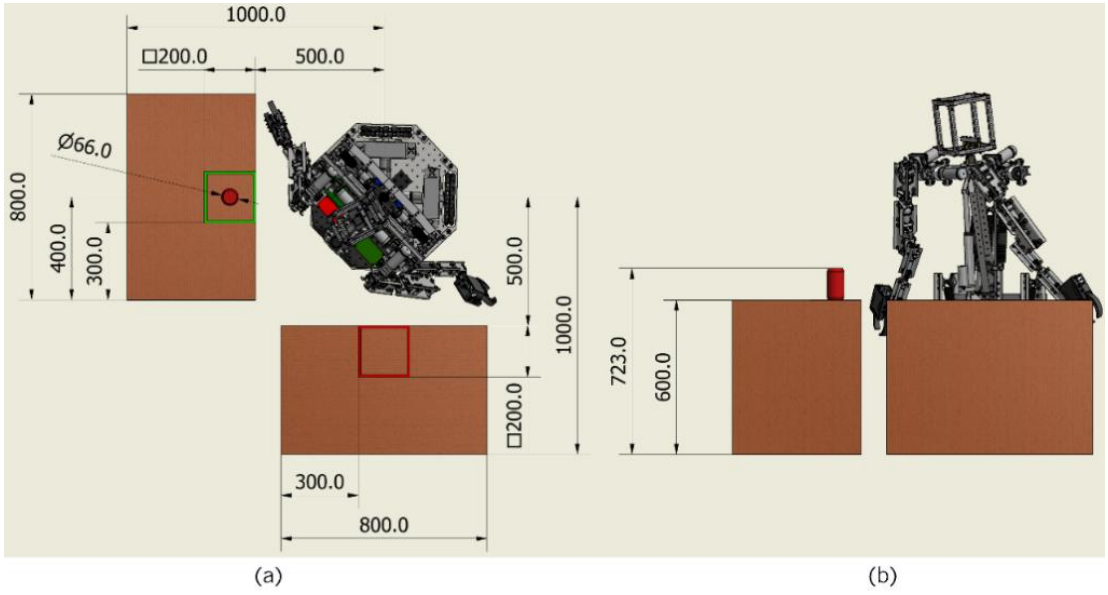


Figure 5.2 – Work environment for the robot for the neural network metal can manipulation task

5.2.1. State Space and Action Space

For the neural network to operate, the status of the system must evolve along steps. The result of each step is a new state, which is a function of the state at the previous step and the action which was taken:

$$s_{n+1} = f\left(s_n, a_n\right) \tag{5.1}$$

where s_n represents the state space of the system, and a_n is the action space.

The state space must contain information that fully characterizes the robot, and its surrounding environments without any additional input. This was achieved by using a total of 50 variables (see Table 5.1): 21 to define the position of joints; 21 to define the velocities of joints; two to define the status of the can interaction (Figure 4.17) — two variables to store both the current status and the most advanced reached status for reward attribution; one variable to report collision detection; one variable to define the state in which the can was grabbed by the claw; three variables to describe the position of the can when not grabbed; and one variable to store the elapsed time.

Table 5.1 – Variables describing the status of the robot simulation environment

i	**Description**	*i*	**Description**
0	Base x Position	(...)	(...)
1	Base y Position	41	Joint 18 Velocity
2	Base Yaw Angle	42	Elapsed Time
3	Joint 1 Position	43	World Can Status
(...)	(...)	44	Max World Can Status
20	Joint 18 Position	45	Collison Status
21	Base x Velocity	46	Claw Grab Position
22	Base y Velocity	47	x Position Can
23	Base Yaw Angular Velocity	48	y Position Can
24	Joint 1 Velocity	49	z Position Can

The PPO neural network returns a value of either −1 or 1 for each neuron of the output layer. These outputs are responsible for controlling the acceleration of each of the joints, increasing it in the positive or negative directions. Preliminary tests of the neural network's performance showed that it was vital to have a command for each joint which corresponds to "decelerate and maintain position". The inclusion of this third possibility requires the use of two neurons to control each joint following the logic of Table 5.2.

Table 5.2 – Logic table for the control of the acceleration of each joint using two neurons.

Output	Neuron 1	Neuron 2
-1	$+1$	-1
0	$+1$	$+1$
0	-1	-1
$+1$	-1	$+1$

The control of every single joint of CHARMIE would require 42 neurons on the output layer for 21 accelerations. However, since the studied problem will be focused on object manipulation, the model was simplified to have 18 neurons controlling nine accelerations: rotation of the robot's base; contraction and extension of the two linear actuators controlling the squatting and trunk flexion motions, and one for each of the six joints in the robot's arms. The opening and closing of the claw will not be controlled by the neural network, but rather by hard codding which instructs the claw to open or close when all required conditions for adequate can manipulation are achieved.

5.2.2. Observation Space

The observation space refers to the inputs of the neural network; it is the simulation data that the neural network can access to assess its status. A total of 26 variables were used to describe the observation space (see Table 5.3).

Table 5.3 – Variables describing the observation space of the neural network

i	Description	*i*	Description
0	Base Yaw Angle	13	Right Arm Joint 1 Velocity
1	Squatting Actuator Position	14	Right Arm Joint 2 Velocity
2	Trunk Flexion Actuator Position	15	Right Arm Joint 3 Velocity
3	Right Arm Joint 1 Position	16	Right Arm Joint 4 Velocity
4	Right Arm Joint 2 Position	17	Right Arm Joint 5 Velocity
5	Right Arm Joint 3 Position	18	Right Arm Joint 6 Velocity
6	Right Arm Joint 4 Position	19	Right Claw Velocity
7	Right Arm Joint 5 Position	20	World Can Status
8	Right Arm Joint 6 Position	21	Right Claw x Goal Distance
9	Right Claw Position	22	Right Claw y Goal Distance
10	Base Yaw Angular Velocity	23	Right Claw z Goal Distance
11	Squatting Actuator Velocity	24	Right Claw Pitch Angle
12	Trunk Flexion Actuator Velocity	25	Right Claw Roll Angle

Twenty of these variables inform the neural network about the current state of the robot. Another variable communicates which task the robot is in (searching for the can, grabbing the can, transporting the can, or unloading the can). The final five variables inform the neural network of its status regarding the performance of the task, it is the way to communicate different objectives to the neural network, and the neural network should learn by itself that its goal is the minimization of the modulus value of these variables. Goal positions are communicated via the three cartesian coordinate distances of the claw to its target (this target switches mid-simulation from the can to be grabbed, to the unloading point of the can). For adequate can manipulation, the neural network is also informed of the roll and pitch angles of the claw.

To enhance the performance of the neural network model, feature scaling employing min-max normalization was applied to the observation space into a range of $[-1, 1]$ as follows:

$$x' = 2\frac{x - \min(x)}{\max(x) - \min(x)} - 1 \tag{5.2}$$

5.2.3. Reward Function

The goal of the reward function is to guide the neural network toward the desired behaviors. It is a mathematical equation whose output value should increase as the system takes actions with positive outcomes. Modeling a good reward function is paramount for the performance of neural network systems. When reward functions do not have appropriate modeling, the system can tend towards undesired behaviors that are mathematically efficient.

In a complex system such as CHARMIE, where the desired state is a very reduced subset of all possible states, it is very hard for the neural network to discover a solution just by rewarding its completion. The steps towards completion must be rewarded as well.

The first approach in modeling the reward function was to reward any action that brought the robot's state closer to the desired state. This originated two unintended consequences depending on the weights given to the rewards: i) Sometimes, the system would oscillate back and forth to generate more steps where positive actions were taken, and ii) due to how the arm motion and joints are configurated, it is sometimes required for the system to take non-optimal actions to reach the desired state, this approach incentivized convergence towards local maximums.

A second concept that had to be discarded at first was the punishment of collisions. A collision can be a cause of catastrophic failure in real application scenarios. However, targets of collisions, such as the counters the can is resting at, are close to the goal of interaction. To avoid the punishments, the neural network learned to always stay clear of these obstacles, therefore never approaching the intended positions. After testing, the best approach was to use the sum of two main rewards:

- Utilize a set of discrete rewards to guide the claw to its desired position. This reward is attributed to all timesteps and rewards bringing the claw closer to the desired position. This reward can go from 5 to 35, corresponding to the sum of 5 sub-rewards. Each sub-reward will evaluate one of the cartesian coordinates or one of the angles of the robot claw (see Table 5.4.).
- Reward the completion of a task (the value of the Max World Can Status increasing) with a reward that has a high value. This ensures the neural network registers a good status and does not unlearn it. The value of these rewards decreases over time to incentivize fast completion of tasks (see Equation 5.3).

Table 5.4 – Reward attribution for CHARMIE's motion based on the position of the manipulation claw

Reward	***x* distance [m]**	***y* distance [m]**	***z* position [m]**	**Pitch [º]**	**Roll [º]**
1	[0, 0.01[	[0, 0.01[	[0, 0.61[	[0, 1[	[0, 1[
2	[0.01, 0.02[	[0.01, 0.02[	[1.00, +∞ [	[1, 2[	[1, 2[
3	[0.02, 0.05[	[0.02, 0.05[	[0.80, 1.00[	[2, 5[	[2, 5[
4	[0.05, 0.10[	[0.05, 0.10[	[0.72, 0.80[	[5, 10[	[5, 10[
5	[0.10, 0.20[	[0.10, 0.20[	[0.70, 0.72[∨ [0.61, 0.65[	[10, 20[	[10, 20[
6	[0.20, 0.50[	[0.20, 0.50[	[0.69, 0.70[	[20, 30[	[20, 30[
7	[0.50, + ∞ [	[0.50, +∞ [	[0.65, 0.69[	[30, +∞ [	[30, +∞ [

$$\mathrm{Rwd}_2 = \frac{100000}{1+\frac{t}{100}} \tag{5.3}$$

These rewards were adequate for leading CHARMIE to learn the intended behaviors. Further optimization of the neural network can still be achieved via the fine-tuning of hyperparameters, leading to both faster and more stable learning.

5.3. Agent Training

With the Environment for training created, and the neural network type created, it is possible to start the training of the agent of the neural network. The training of each agent under determined conditions is an automated process, the NN architecture is defined, the hyperparameters are established, the environment conditions are set, and then the neural network is trained along several iterations. The role of the researcher is to monitor the training performance and implement a methodology to improve this performance to achieve stable and fast training.

In a final real-case application scenario, the neural network must reach a generalizable behavior, so that it can complete the defined objectives (grabbing a can and transporting it to a target position) regardless of its start state. However, at this state of research, for a faster convergence of results which results in a faster detection of errors and optimization, and to achieve a proof of concept before requiring further computational resources, the goal will be for the neural network to solve the trajectory of the manipulation for a fixed starting position.

From early testing, the system achieved better performance, and explored more actively, when joints were initialized in their midpoint configuration. When joints were initialized near their limits, the system could get locked into an end-position of an actuator due to high immediate reward and not explore any further behaviors. Considering these limitations, the joints were initialized in the positions listed in Table 5.5. For simplicity, the other non-controlled joints were placed in a configuration that would completely avoid collisions (left arm raised)

Table 5.5 – Start configuration of the CHARMIE robot for the neural-network agent training

Joint	Configuration	Joint	Configuration
Base Yaw Angle	0.0^o	Right Arm Joint 3 Position	50^o
Squatting Actuator Position	0.373 m	Right Arm Joint 4 Position	67.5^o
Trunk Flexion Actuator Position	0.382 m	Right Arm Joint 5 Position	0^o
Right Arm Joint 1 Position	0^o	Right Arm Joint 6 Position	0^o
Right Arm Joint 2 Position	45.0^o	Right Claw Position	55.0^o

With a defined starting condition, the agent training was initialized. The maximum length for each simulation episode was established at 200 s (2000 timesteps) of elapsed simulator time. This allows the robot to explore during enough time in each iteration, so it becomes possible to reach a target position, while it also limits the length for each simulation to increase computational efficiency and promote the discarding of inadequate policies. To allow a proper evaluation of each agent training condition, models

were trained for a total of 4 million timesteps, which corresponds to 2000 episodes from start to finish if the final goal is never reached (completing the desired task causes the simulation to finish earlier, allowing for more training episodes within the same period).

Determining the architecture of the neural network (number of neurons and their layout) is not an exact science. By default, on the used library, neural networks employ a Rectified Linear Unit (ReLU) activation function with fully connected (dense) layers.

The tunning of the architecture will mainly refer to the number of neurons. A higher number of neurons does not necessarily mean better performance, the goal is to find a good balance; few neurons result in underfitting and the inability to learn how to solve the problem, while excessive neurons increase the risk of overfitting and slow down training. It is also important to balance the width (number of neurons per layer) and depth (number of layers) when structuring the hidden layers of a neural network.

Several factors affect the number of neurons to be used: the work environment is complex (a continuous control environment with several interconnected actuators), and both the observation space and action space have high dimensions (26 and 18 variables respectively). These considerations incentivize the use of both deep (2 to 9 layers) and wide (128 to 512 neurons) neural networks. Since PPO algorithms are being considered, the same depth and width of layers will be used for both the actor and the critic.

The study of the NN started with an architecture of two layers with 128 neurons each. First, the depth was varied until the best learning behavior was attained. Figure 5.3 shows that a depth of 6 hidden layers outperformed all other options in terms of both learning speed, maximum reward obtained at the end of learning, and learning stability (In the following figures, average rewards are shown instead of immediate rewards to avoid spikes that make data harder to interpret). Then, the width for this depth was changed, however, the starting 128 neurons had a better performance than any tested alternative (see Figure 5.4). The ideal configuration was set at six layers of 128 neurons each (For future studies, with different learning rates or number of episodes, the optimal architecture can differ).

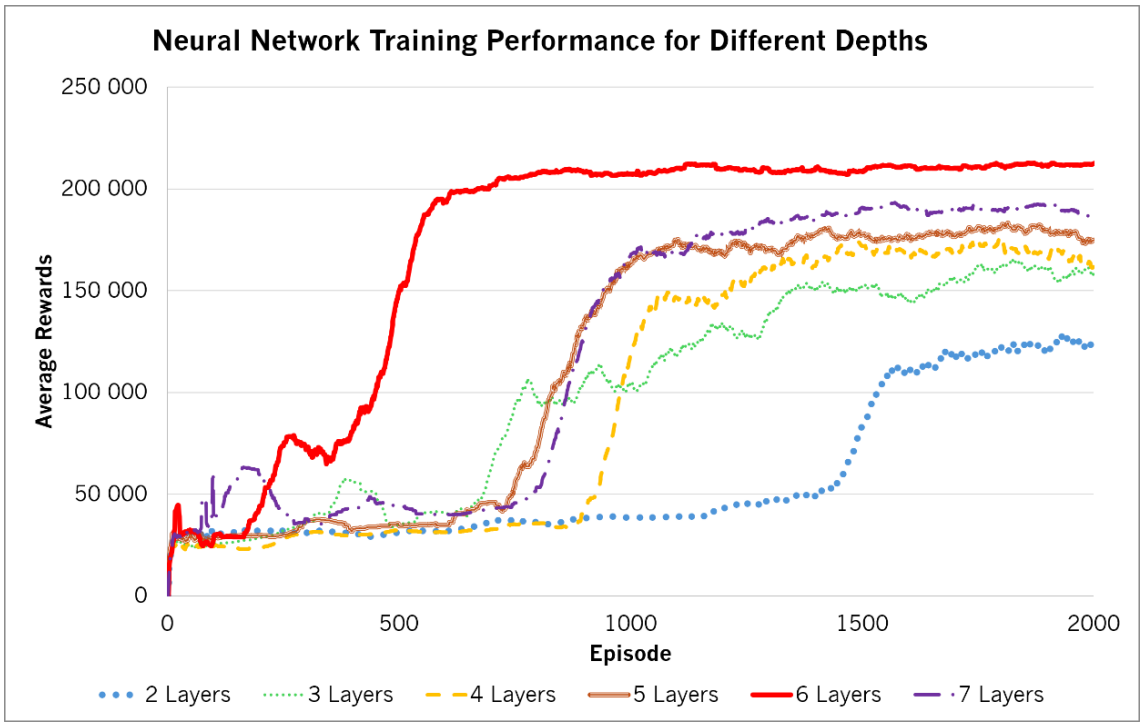


Figure 5.3 – Average rewards for different neural network depths for a constant width of 128 neurons

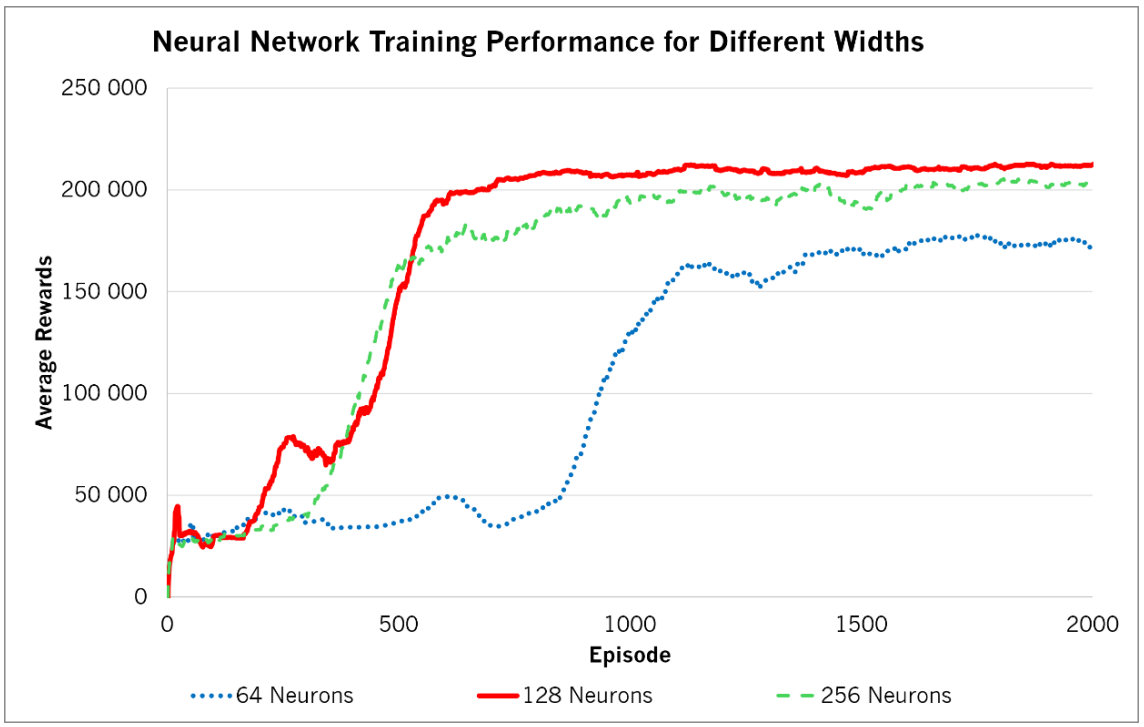


Figure 5.4 – Average rewards for different neural network widths for a constant depth of six layers

The current NN algorithm was able to learn and achieve satisfactory results, but two main problems were identified in the learning that had to be addressed: the learning degradation of the PPO algorithm, and the catastrophic forgetting issue.

5.3.1. Learning Degradation

When training the NN for the trajectory control, during the first episodes the reward kept increasing over time, as desired. However, after a certain point, the rewards started decreasing more and more, and the NN was unlearning previously learned tasks. This is a sign of learning degradation.

Learning degradation can be the result of many phenomena, such as improper hyperparameter selection, overfitting, bad reward structure, or excessive incentives toward exploration. In the context of PPO algorithms, a constant learning rate can often become a cause of overfitting or instability. This effect can be mitigated by implementing a learning rate decay.

In the first training tests, the learning rate was set at a constant value of 0.001. The effects of learning degradation were very evident under these conditions. The NN's performance was substantially increased by decreasing this learning rate to a constant value of 0.0001. For further optimization, a linear decay was implemented which starts from 0.0001 and decreases to 0.0 over time. The performance increased once again, and the learning degradation issue was fully solved (see Figure 5.5).

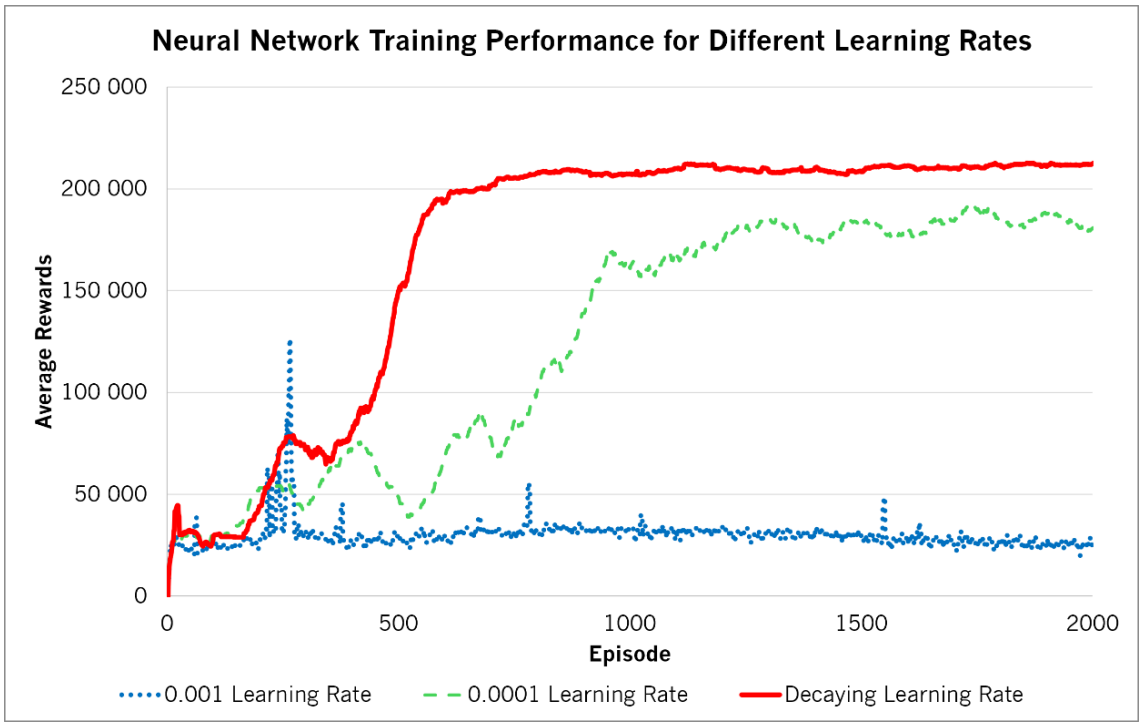


Figure 5.5 – Difference in training results for different learning rates

5.3.2. Catastrophic Forgetting

Catastrophic forgetting, in the context of reinforcement learning, refers to the process of losing previously learned knowledge when faced with learning a new task. This is especially problematic when a system is forced to learn sequential tasks; the system is either incapable of learning the second task due to the different rewards or forgets how to execute the first task upon learning the second one. This is a key challenge in reinforcement learning.

In CHARMIE's control, catastrophic forgetting was a clear issue. The main overall task is divided into two stages, going into the can position and grabbing it, and then going into a target position to ungrab it. Once the target positions change, so does the reward structure.

The strategy to deal with catastrophic forgetting in CHARMIE's case was to pre-train the neural network for a more generalized work scenario. As an alternative to beginning training in a fixed start position considering the whole manipulation tasks, ten different start scenarios were established. Each of these scenarios corresponds to different positions and stages of the manipulation task, in some of them the claw is unopened and close to the start position, while in others the claw is closed, the can is grabbed, and the next step is for the robot to ungrab the can in the target position. By pre-training for generalized cases, the neural network avoids overfitting to a particular task. Table 5.6 lists the ten different starting conditions considered. In the pre-training, each episode will start at one of those starting conditions chosen at random.

Table 5.6 – Starting conditions for the ten scenarios used to train CHARMIE for generalized behaviors

Joint	**Scenario**				
	1	**2**	**3**	**4**	**5**
Base Yaw Angle	0.0^o	20.0^o	35.0^o	60.0^o	-20.0^o
Squatting Actuator Position	0.373 m	0.373 m	0.34 m	0.37 m	0.34 m
Trunk Flexion Actuator Position	0.382 m	0.382 m	0.34 m	0.37 m	0.34 m
Right Arm Joint 1 Position	0.0^o	10.0^o	0.0^o	11.0^o	20.0^o
Right Arm Joint 2 Position	45.0^o	35.0^o	49.0^o	25.0^o	55.0^o
Right Arm Joint 3 Position	50.0^o	20.0^o	46.0^o	40.0^o	55.0^o
Right Arm Joint 4 Position	67.5^o	47.5^o	33.5^o	57.5^o	60.5^o
Right Arm Joint 5 Position	0.0^o	10.0^o	15.0^o	5.0^o	25.0^o
Right Arm Joint 6 Position	0.0^o	10.0^o	15.0^o	5.0^o	25.0^o
Right Claw Position	55.0^o	55.0^o	55.0^o	55.0^o	55.0^o
World Can Status	0	0	0	0	0

Joint	Scenario				
	6	7	8	9	10
Base Yaw Angle	50.0º	130.0º	60.0º	150.0º	152.0º
Squatting Actuator Position	0.373 m	0.37 m	0.37 m	0.38 m	0.38 m
Trunk Flexion Actuator Position	0.382 m	0.385 m	0.37 m	0.38 m	0.385 m
Right Arm Joint 1 Position	0.0º	1.0º	11.0º	20.0º	0.0º
Right Arm Joint 2 Position	45.0º	45.0º	25.0º	45.0º	45.0º
Right Arm Joint 3 Position	50.0º	50.0º	40.0º	50.0º	50.0º
Right Arm Joint 4 Position	67.5º	64.5º	57.5º	67.5º	67.5º
Right Arm Joint 5 Position	0.0º	22.0º	85.0º	0.0º	0.0º
Right Arm Joint 6 Position	45.0º	32.0º	85.0º	0.0º	0.0º
Right Claw Position	0.0º	0.0º	0.0º	0.0º	0.0º
World Can Status	5	5	5	5	5

The learning of the training scenario had high reward fluctuation depending on the considered starting scenario. However, as illustrated in Figure 5.6, the neural network completed tasks in all scenarios, and its performance was still improving over time.

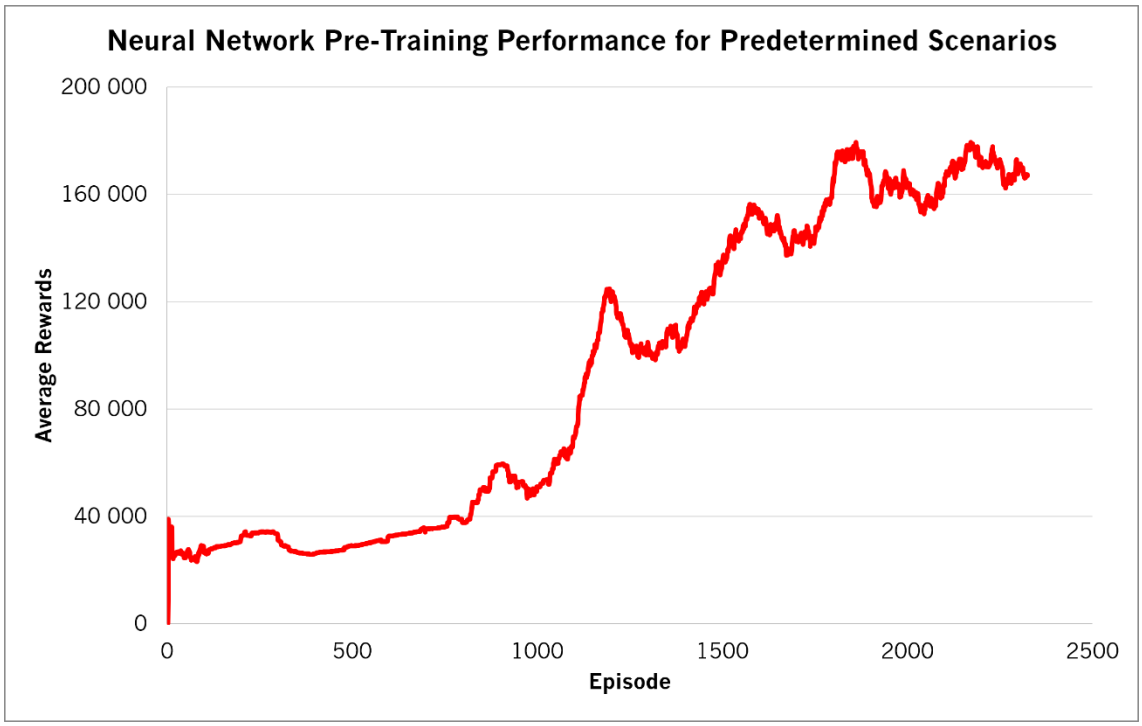


Figure 5.6 – Rewards over time for the pre-training of the neural network

The pre-trained neural network was then loaded and implemented for training in the singular application scenario with fixed starting conditions. The performance of the Neural Network increased (see Figure 5.7). In some episodes, the neural network was able to fully complete the task, thus achieving

desirable performance. However, the learning rate was not stable, and there was not a substantial increase in rewards over time, which should be addressed in hyperparameter tuning.

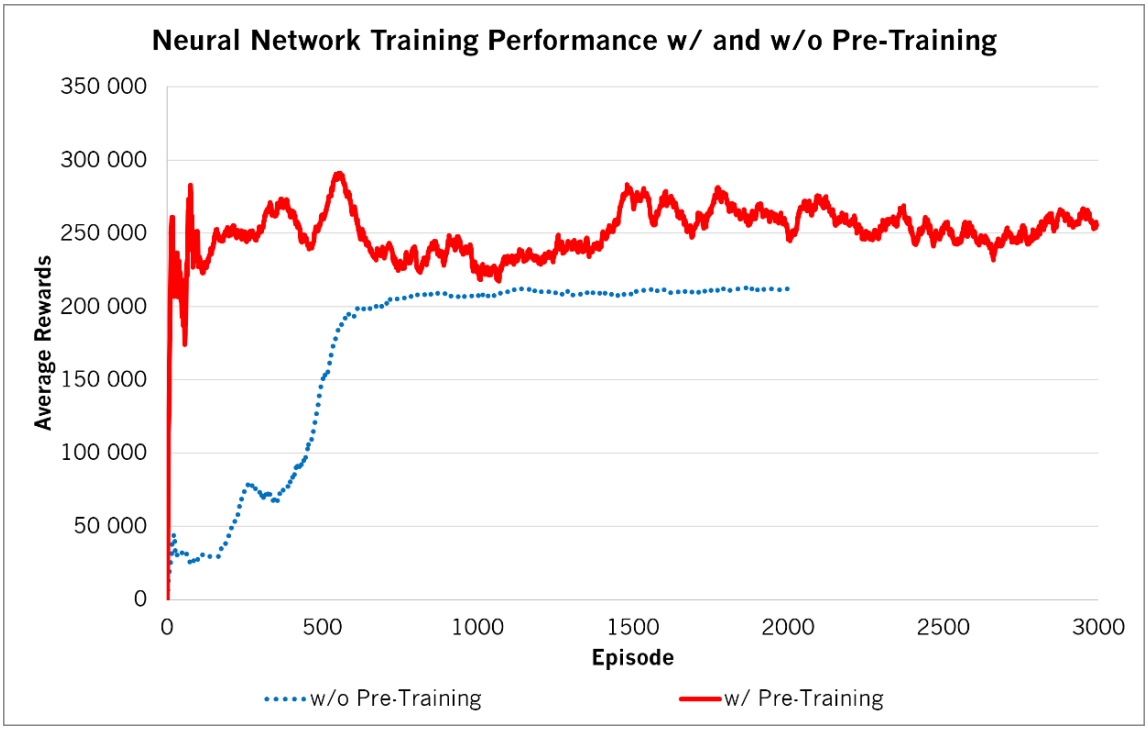


Figure 5.7 – Comparison of results of the training for the application scenario with and without pretraining.

5.4. Result Analysis and Performance Improvement

The final test for the implemented PPO control algorithm was to achieve the most optimal training performance with the most efficient, accurate, and fast resulting trajectory. The hyperparameters used for these final tests are all listed in Table 5.7.

Table 5.7 – Hyperparameters and architecture for the training of the PPO neural network

Parameter	Value	Parameter	Value
Actor Depth	6	Generalized Advantage Estimator Lambda	0.95
Actor Width	128	Clip Range	0.0
Critic Depth	6	Clip Range Value Function	None
Critic Width	128	Normalize Advantage	True
Learning Rate	0.0001 to 0.0	Value Function Coefficient	0.5
Number of Steps	2048	Maximum Gradient Clipping	0.5
Batch Size	64	Generalized State-Dependent Exploration	False
Number of Epochs	10	Frequency to Sample a New Noise Matrix	−1
Gamma	0.99	Seed	9600
Entropy Coefficient	0.01	-	-

The used simulator does not model the dynamics of the interaction with the metal can. Therefore, a tolerance was established which defines the robotic claw as ready to interact with the metal can. When this tolerance is looser, the training performance is much faster and more stable, but when the tolerance is too loose, the realistic accuracy of the scenario may be hindered. The advantages of both loose and tight tolerances were harnessed by setting up simulation environments with different degrees of tolerance (see Table 5.8).

Table 5.8 – Tolerances for different scenarios for the training of CHARMIE's trajectory control.

Parameter	**Scenario 1**	**Scenario 2**	**Scenario 3**	**Scenario 4**
Angular Tolerance	25º	10º	10º	10º
Claw Displacement Tolerance [m]	0.11	0.06	0.03	0.03
Can Height Grab Tolerance [m]	0.07	0.04	0.04	0.04
Can Height Placement Ungrab Tolerance [m]	0.10	0.05	0.05	0.05
Collision Punishment	False	False	False	True

The first scenario is easy to train but does not accurately reflect reality (allowing the claw to grab the can as far as 11 cm away), while the tolerances of the final scenario are an accurate reflection of reality, and the small tolerances reflect adjustments and displacements that would happen with dynamic interactions.

The best-performing neural network was obtained by executing a four million timestep pre-training on the scenario with the loosest tolerances (Figure 5.8).

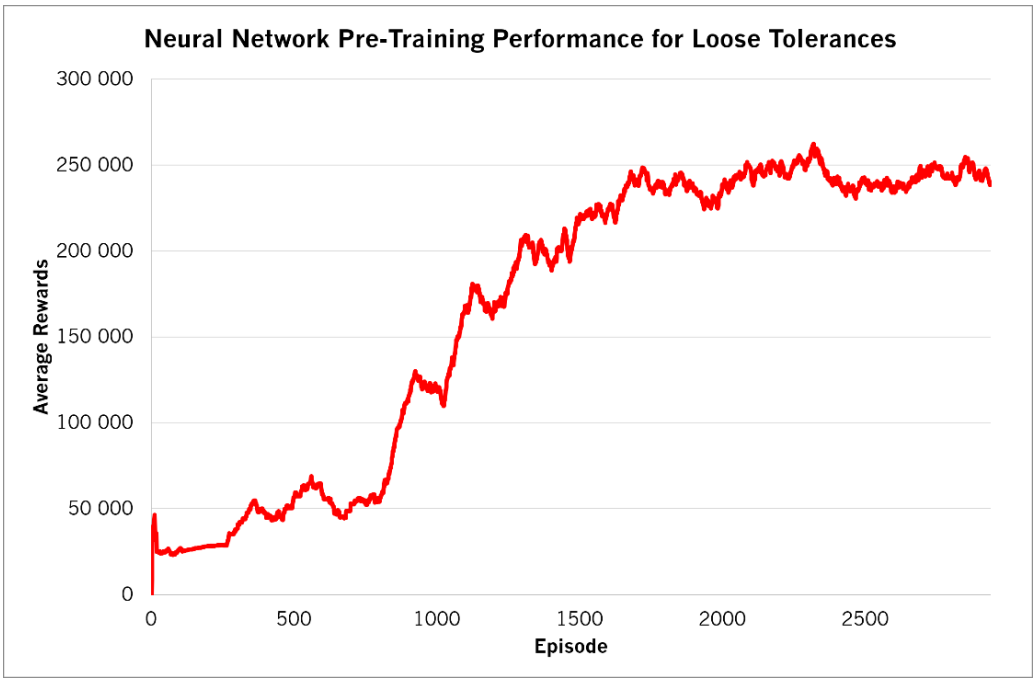


Figure 5.8 – Training results for the pre-training of the neural network in a loose tolerance scenario.

After the initial pre-training, the trained network was loaded and trained for one million timesteps for each scenario of growing tighter tolerances. The results are shown in Figure 5.9.

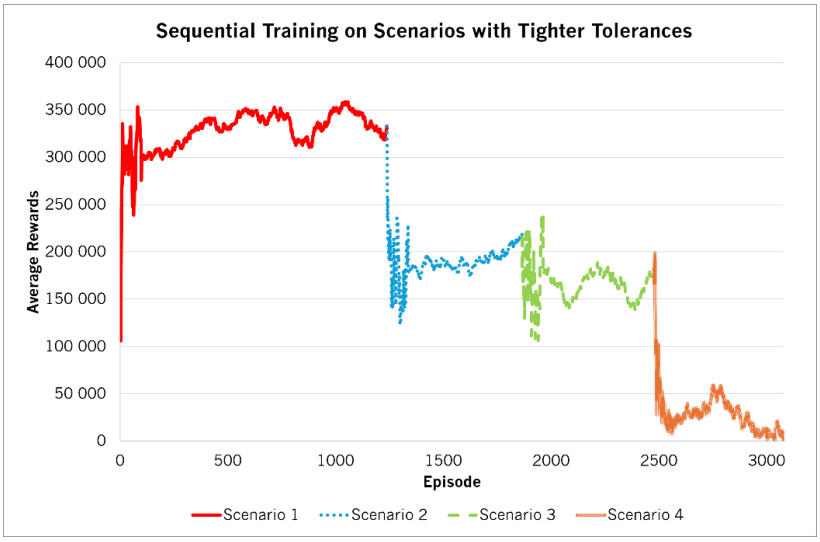


Figure 5.9 – Training results for the neural network in scenarios with sequentially tighter tolerances.

The training performance decreased as the scenarios became more complex. The first scenario, which corresponds to the conditions of the pre-training, had a constantly good performance. The tightening of the tolerances decreased the performance, and finally, punishing collisions greatly increased the instability in learning. However, the presented rewards report the average performance, the consistency decreased over time, but the peak performance did not decrease as much while the outcomes became more and more realistic (see Figure 5.10).

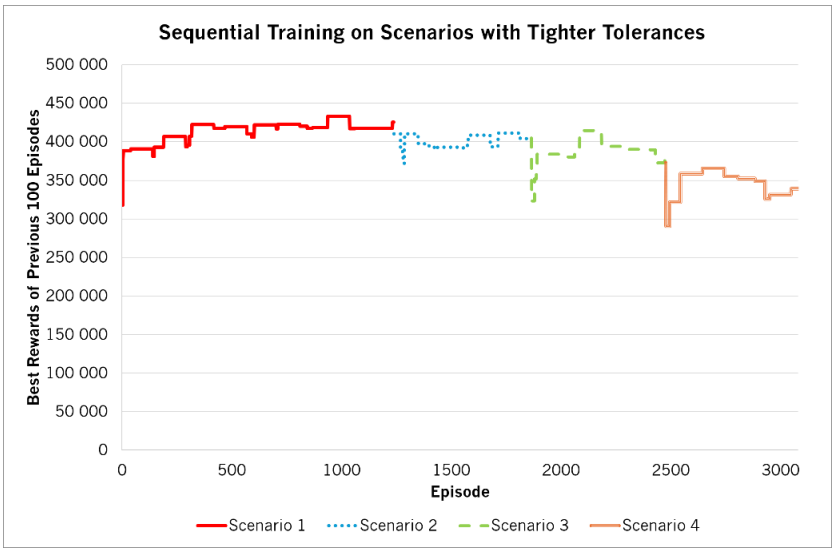


Figure 5.10 – Maximum rewards for the neural network in scenarios with sequentially tighter tolerances.

The conclusion of the training of the neural network allows the inspection of its performance. In this final training, tighter tolerances were implemented, and collisions were punished, therefore, the neural network learned how to manipulate the metal can in a realistic scenario while avoiding collisions. The entire task was completed in 32 seconds. Figure 5.11 highlights the learned behavior, showcasing how the joints of the arm moved over time with the aim of simultaneously fulfilling all angular and positional goals.

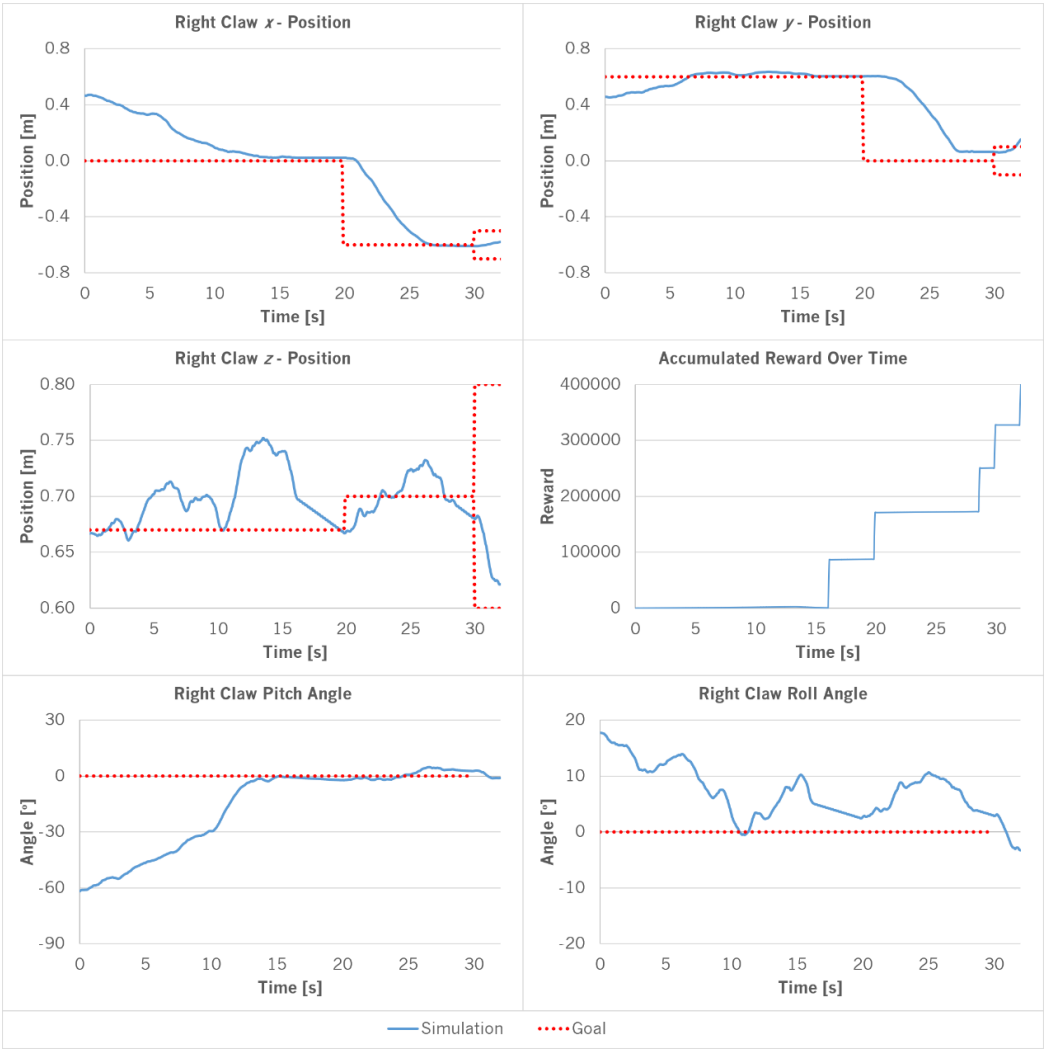


Figure 5.11 – Evolution of the right claw position and orientation over time when controlled by the NN

Figure 5.12 displays the motion over time of the trajectory shown in Figure 5.11. The first fifteen seconds correspond to preparing motions, the NN learned the required angle and height to approach the metal can for manipulation and evolved from the starting state into this configuration. These preparations are executed at a safe distance from the can to avoid collisions. After setup is complete, the claw approaches the can and starts grabbing it at the 17.5 s time frame, completing the motion after 20 s. Since the robot is now in a good configuration to both grab as well as to drop the can, it then rotates at

an accelerated speed for the following 7.5 seconds to reach the second counter. Once there, the claw opens to drop the can successfully at the target destination. After 30 seconds of motion, the task is complete, and the robot slightly retreats the claw to back away into a safer position. Thus the simulation is concluded in a total of 32 seconds efficiently and logically without any human intervention.

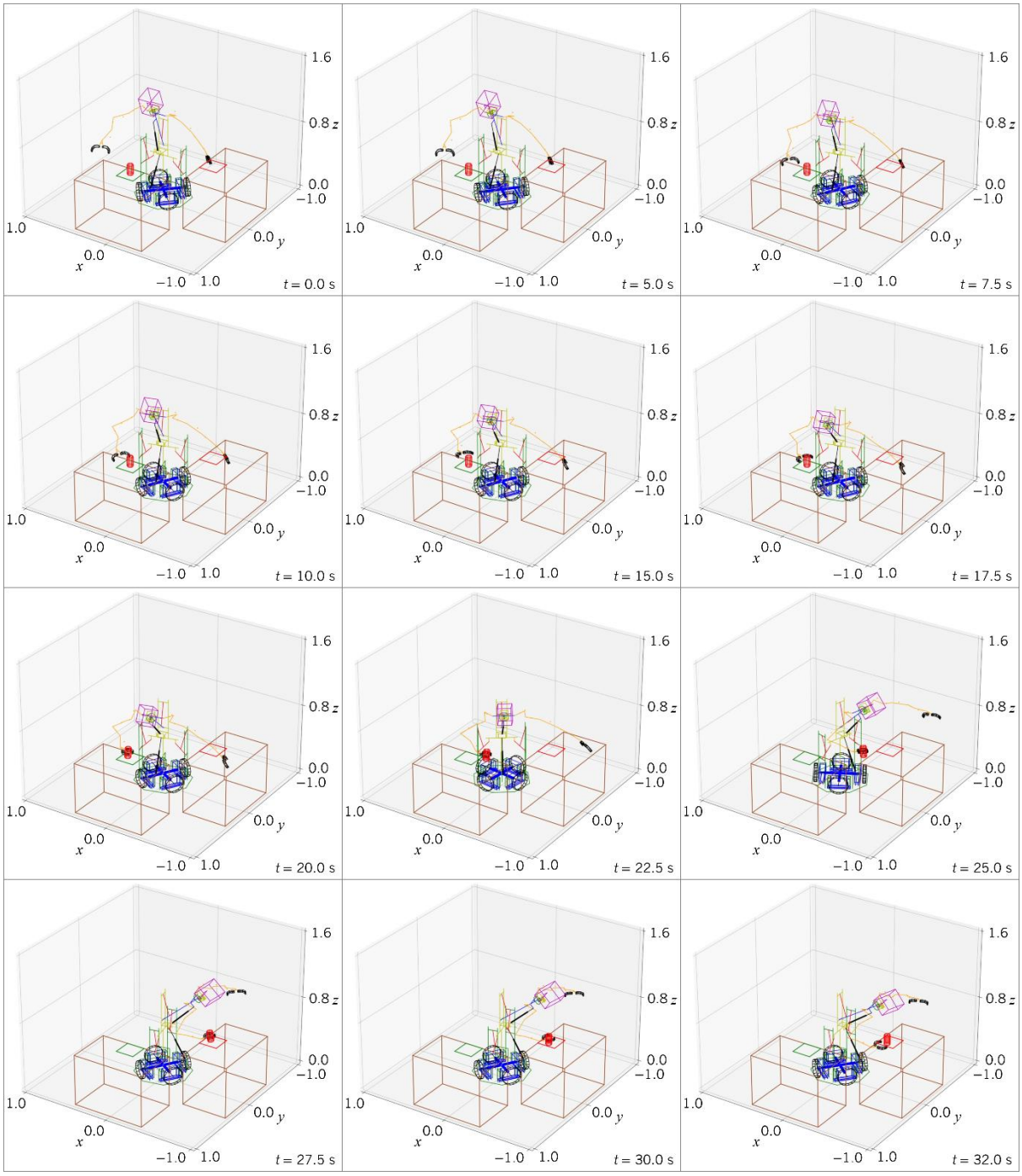


Figure 5.12 – Motion over time of the robot controlled by the Neural Network in the simulation environment

5.5. Summary and Discussion

In this chapter, a neural network control solution was implemented that was able to control the trajectories of CHAMIE's manipulation motions. The studied problem was the movement of a metal can from one counter to the other, which required the control of the robot's yaw orientation, actuation of the hip and torso mechanisms, as well as the motion of the robot's right arm. The system was able to perform the task in the simulation environment without the need for any human intervention.

The study of the neural network was initialized by first comprehending the proximal policy optimization neural network and its implementation into a Python IDE for it to be compatible with the developed simulator. The Stable Baselines3 and Gymnasium libraries were pivotal in this integration.

One of the more vital steps in training a neural network agent is the definition of the environment. The simulation used to evaluate how states evolve depending on the performed action is the simulator developed in this thesis. The space state, action space, and observation space were determined so that the NN could perceive the robot and the environment's state along the different timesteps while acting accordingly by controlling the robot's joints via their acceleration. Finally, a reward function was established that rewards both the robot's state shifting towards the completion of an action, as well as a bigger reward that incentivizes completing tasks as fast as possible.

Different architectures were tested for the NN to learn the required problem as fast as possible while generating high-performing trajectories according to the reward function. This study concluded that for the current problem, six hidden layers with 128 neurons each are ideal for both the actor and the critic neural network. Two key issues were initially identified in the training, namely learning degradation and catastrophic forgetting. Learning degradation was solved by adding a linear decay to the learning rate of the NN. Catastrophic forgetting was addressed by performing a pre-training of the neural network under different simulation conditions where the starting point of the simulation task was randomized.

Finally, the neural network was optimized and tested for performing the object transportation task. The robot displayed fast intuitive motion, approaching each required target directly, completing the task in 32 seconds.

"You certainly usually find something, if you look,
but it is not always quite the something you were after."
J. R. R. Tolkien

6. CONCLUSIONS AND FUTURE DEVELOPMENTS

This work succeeded in creating a mobile anthropomorphic manipulator robot for daily tasks and in achieving its control. To conclude the document, this section goes over the concluding remarks of the project, how they are interconnected, and then suggests future work to continue the robot's development.

6.1. Concluding Remarks

A novel manipulator robot has been developed in this work. Several key issues within the multidisciplinary field of robotics were addressed, such as the morphology of the robot, the design and optimization of its key mechanisms, the study of the system's kinematics and dynamics in a virtual simulation, and the autonomous trajectory control of the joints employing reinforcement learning.

The most relevant literature for the project was revised in Chapter 2. The literature review was used to establish the current trends in the design of robotic systems. This highlighted the interest in developing robots with object manipulation capabilities for domestic applications. Furthermore, it established guidelines for the design of CHARMIE based on the feedback of other applied robots. These guidelines include features like presenting a human-like morphology and adopting wheeled locomotion. Moreover, the robot's project was further refined by employing mechanism design concepts from the literature. The adopted four-wheel locomotion solution required the implementation of an independent suspension system; the simplest solution was to implement a prismatic joint into each of the wheels with a spring limiting its motion. A fundamental aspect of mobile robots is their limited energy supply; a common solution found in the literature to reduce energy consumption by minimizing actuator loads is to use static balancing systems.

A second aspect of the literature analysis was to determine the best theoretical approaches to study the robot. For the computational analysis of the robot's bodies and mechanisms, a multibody

dynamic simulator was employed that solves the forward kinematics and inverse dynamics. This simulator was based on published recursive algorithms, employing Euler angles and rotation matrices to model the robot's 3D kinematics. The inverse dynamics was solved using Newton-Euler equations. Finally, a brief review of NN control solutions justified their current relevance in robotics for tackling unstructured work environments and identified PPO as the ideal algorithm for the trajectory control of CHARMIE.

The next step was the development of the robot itself. In Chapter 3, CHARMIE was conceptualized as a human-inspired robot with six main systems: a base with omnidirectional locomotion and an independent suspension system, an articulated hip to allow the robot's squatting and trunk flexion motions, the robot's structural torso, the two manipulator arm with 6-DOF each and a claw-like end-effector, and finally a 2-DOF robot head to include sensors and communication features. A detailed CAD model was created for each system considering both its manufacturing and assembly, which in turn was the basis for the technical drawings for productions. A more in-depth analysis was conducted for two of the sub-systems. The first was the parametric optimization of the static balancing mechanisms for the squatting and trunk flexion motions. With the aid of the simulation environment, theoretical actuator loads were reduced by approximately 75% with the aid of tension springs. The simulator was also employed to study the forces and torques for the adequate selection of actuators and gears to implement in the 6-DOF serial arms.

The project of CHARMIE as a robotic system culminated in the construction of a physical prototype. The prototype employed off-the-shelf components, as well as steel metal parts produced in a lathe, and sheet metal that was laser cut and bent for added resistance. Three of the robot's main systems were fully prototyped: the base, the hip, and the torso. This allowed the verification of the performance of the robot's locomotion, suspension, squatting, and trunk flexing motions, thus validating specific mechanisms such as the static balancing employing tension springs. Technical drawings of all the developed parts are documented to allow the full replicability of the proposed robotic system.

One of the pillars in the study of the robot was the multibody simulator presented in Chapter 4. The first step towards implementing the robot in a computational environment was to gather the geometry, mass, and inertia properties of all bodies with the assistance of detailed CAD models. The multibody models were computed by employing recursive algorithms. For the forward kinematic analysis, each body is modeled by an algorithm based on the cartesian coordinates of the points of joints and the Euler angles expressed in rotation matrices defining the rotations between bodies. This algorithm can be applied to serial and branched kinematic chains, but closed and over-constrained loops, as well as indirectly

actuated joints, required further geometrical analysis for integration within the kinematic model. The inverse dynamics employed recursive Newton-Euler algorithms from the literature to model the forces and torques between each pair of bodies based on the known kinematics. The algorithm determines forces applied between pairs of bodies, so when two consecutive bodies are connected by more than a single joint, additional mathematical formulations were devised to determine the distribution of forces and torques within each joint. The used notation, and its implementation, were completely validated by comparing the results obtained with those generated by a commercial software.

The devised multibody simulator is customizable, compatible with other programs, and implementable in different hardware. To incorporate further environment interaction and study of the robot's navigation, the detailed dynamic model of the robot's locomotion and suspension was devised. Analyzing the locomotion accurately requires smaller timesteps, increasing the needed computational loads, but this is a powerful tool for future studies. The study of CHARMIE's manipulative capabilities required analyzing collision between bodies, as well as interaction with targets of manipulation. The collision analysis was carried out based on the kinematic study of bodies that can intersect. Computational requirements for this study were heavily reduced by implementing a level of detail based approach, where bodies get modeled with more detail as they get closer to situations where impact is possible. In terms of object manipulation, it was important to consider the goals of the simulator: to train neural networks for the trajectory control of CHARMIE. Computational efficiency was paramount, so a kinematic-based approach, disregarding dynamic effects, was utilized. Object manipulation was modeled as a finite state machine, where an adequate interaction with a target of manipulation following set conditions (such as correct claw position and orientation) allows the robot to grab the object and transport it. Opening the claw, thus releasing the object, finishes the manipulation action.

Chapter 5 addresses the use of a PPO reinforcement learning algorithm for the trajectory control of CHARMIE. The goal was to validate the feasibility of using neural networks for the control of the robot's actuators, so a simple task was devised. The aim is for the robot to autonomously learn how to transport a metal can from one counter to another as quickly and smoothly as possible.

The first step for reinforcement learning was to establish the environment for the neural network to train and perform on. This process was obtained by integrating the multibody dynamics simulator into the Gymnasium open-source Python library. The observation space, which defines the inputs of the neural network, was set as the joint positions, velocities, information regarding the goal, and the state of the simulation. For the action space, which defines the output layer, two neurons were used to control the

acceleration of the yaw rotation and each actuator in the robot's hip, torso, and right arm. Finally, the reward function was established to incentivize the robot's claw to approach targets of manipulation and to massively benefit the completion of tasks.

A step-by-step empirical approach was used to define the neural network's architecture and optimize its learning velocity and stability. After analyzing the performance of different architectures, a neural network with six hidden layers of 128 neural networks each provided optimal results. For the control system to be able to complete the transportation task, two main issues were addressed. The first was learning degradation, which was solved by linearly reducing the learning rate of the NN over time. The second problem was catastrophic forgetting, where the neural network was incapable of solving sequential operations, which was corrected by pre-training the network in an adapted environment where start conditions were randomized.

Optimal performance was achieved by training the neural network in different scenarios with increasingly difficult work conditions. A very forgiving scenario with loose tolerances was established for both the pre-training and the first training iterations of the network, where the generated motions were not representative of realistic behaviors, but this scenario made the tasks much easier to learn. The task was made subsequently harder until they reflected real working conditions, where proper joint placements were required and collisions had to be avoided. These considerations allowed the network to generate optimal trajectories. The motion of the robot on the simulator was fluid, logical, and natural. First, the claw was positioned to grab the metal can, then it grabbed it and transported it into a second counter, successfully unloading it at the target destination without any need for human intervention in the control. The task was concluded in 32 seconds without any object collisions.

To sum up, this work provided substantial contributions to the development of a new mobile manipulator robot to perform domestic tasks. The results are grouped into three main pillars: a functional and tested prototype with all information required to allow its replicability; a versatile multibody simulator for developing robotic systems and testing their performance; and a neural network-based solution for the control of the robot's trajectories.

6.2. Future Work

The development of any robotic system is an ever-changing iterative process, therefore, suggestions for key future work are available at any stage of the project. The future aims of the development are threefold: to obtain a finished serviceable product, to further test, evaluate, describe, and generalize the proposed computational tools, and to perform more optimizations on the neural network control.

Considering the physical prototype of the robot, the next step would be to fully prototype all systems of CHARMIE and subsequently develop an outer cover to increase the robot's acceptability. By completing and implementing a full version of the robot, research of a 2.0 version could begin based on the discovered limitations and nuances during the robot's operation in real work environments.

The proposed iterative approach for modeling the robot's forward kinematics and inverse dynamics can incorporate open and branched kinematic chains. Automating the process of tackling closed kinematic chains using systematic notations would expand the applicability of this approach. Furthermore, operation counting based on the input variables (number of bodies, number, and type of joints, ...) should be studied to quantify the computational efficiency of the different modules of the proposed simulator to both identify efficiency bottlenecks and to compare this methodology with other alternatives in the literature. Implementing the iterative methodology to different robotics projects would be beneficial to further identify the strengths and weaknesses of this approach. Since the proposed code is modular, future modules can be developed to enhance object interaction and the robot's locomotion. The possibility of modeling systems with higher accuracy, such as including internal inertia effects from the motors to determine the variation in their inner torques, can be considered for more targeted studies.

Regarding the Neural Network control solution, automated hyperparameter optimization can be implemented to further increase the system's performance. Moreover, different testing scenarios, with more varied starting conditions, could be used to increase the generalizability of the system. The main limiting factor in these tests is the computational time required for the training of neural networks. The evolution of the simulation to include better object interaction physics would facilitate the execution of the manipulation task. Considering that the applicability of the neural network is now confirmed, the next step would be its upgrade into controlling all the motions of CHARMIE, including its locomotion and both limbs. The final performance test for the neural network would be its integration into the robot's embedded computer and the execution of actual tasks in a real work environment.

"*It was amazing how many books one could fit into a room,*
assuming one didn't want to move around very much."
Brandon Sanderson

REFERENCES

Agravante, D. J., Pages, J., & Chaumette, F. (2013). Visual servoing for the REEM humanoid robot's upper body. *2013 IEEE International Conference on Robotics and Automation*, *May*, 5253–5258. https://doi.org/10.1109/ICRA.2013.6631328

Aloulou, A., & Boubaker, O. (2012). A Relevant Reduction Method for Dynamic Modeling of a Seven-linked Humanoid Robot in the Three-dimensional Space. *Procedia Engineering*, *41*(Iris), 1277–1284. https://doi.org/10.1016/j.proeng.2012.07.311

Angeles, J., Ma, O., & Rojas, A. (1989). An algorithm for the inverse dynamics of n-axis general manipulators using Kane's equations. *Computers & Mathematics with Applications*, *17*(12), 1545–1561. https://doi.org/10.1016/0898-1221(89)90054-0

Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, *34*(6), 26–38. https://doi.org/10.1109/MSP.2017.2743240

Arvanitou, E.-M., Ampatzoglou, A., Chatzigeorgiou, A., & Carver, J. C. (2021). Software engineering practices for scientific software development: A systematic mapping study. *Journal of Systems and Software*, *172*, 110848. https://doi.org/10.1016/j.jss.2020.110848

Asano, Y., Kozuki, T., Ookubo, S., Kawamura, M., Nakashima, S., Katayama, T., Yanokura, I., Hirose, T., Kawaharazuka, K., Makino, S., Kakiuchi, Y., Okada, K., & Inaba, M. (2016). Human mimetic musculoskeletal humanoid Kengoro toward real world physically interactive actions. *IEEE-RAS International Conference on Humanoid Robots*, 876–883. https://doi.org/10.1109/HUMANOIDS.2016.7803376

Asfour, T., Azad, P., Vahrenkamp, N., Regenstein, K., Bierbaum, A., Welke, K., Schröder, J., & Dillmann, R. (2008). Toward humanoid manipulation in human-centred environments. *Robotics and Autonomous Systems*, *56*(1), 54–65. https://doi.org/10.1016/j.robot.2007.09.013

Asfour, T., & Dillmann, R. (2003). Human-like Motion of a Humanoid Robot Arm Based on a Closed-Form Solution of the Inverse Kinematics Problem. *IEEE International Conference on Intelligent Robots and Systems*, *2*, 1407–1412. https://www.scopus.com/inward/record.uri?eid=2-s2.0-0348040439&partnerID=40&md5=c7291285650e3fbfcf8d9765a6defbcc

Asfour, T., Regenstein, K., Azad, P., Schröder, J., Bierbaum, A., Vahrenkamp, N., & Dillmann, R. (2006). ARMAR-III: An integrated humanoid platform for sensory-motor control. *Proceedings of the 2006 6th IEEE-RAS International Conference on Humanoid Robots, HUMANOIDS*, *June 2014*, 169–175. https://doi.org/10.1109/ICHR.2006.321380

Aye, M. M. M. (2011). Analysis of Euler angles in a simple two-axis gimbals set. *World Academy of Science, Engineering and Technology*, *81*(9), 389–394. https://doi.org/10.5281/zenodo.1330465

Bajones, M., Fischinger, D., Weiss, A., Wolf, D., Vincze, M., de la Puente, P., Körtner, T., Weninger, M., Papoutsakis, K., Michel, D., Qammaz, A., Panteleris, P., Foukarakis, M., Adami, I., Ioannidi, D., Leonidis, A., Antona, M., Argyros, A., Mayer, P., ... Frennert, S. (2018). Hobbit: Providing Fall Detection and Prevention for the Elderly in the Real World. *Journal of Robotics*, *2018*, 1–20.

https://doi.org/10.1155/2018/1754657

Bajrami, A., Palpacelli, M.-C., Carbonari, L., & Costa, D. (2024). Posture Optimization of the TIAGo Highly-Redundant Robot for Grasping Operation. *Robotics*, *13*(4), 56. https://doi.org/10.3390/robotics13040056

Bang, S. H., Gonzalez, C., Ahn, J., Paine, N., & Sentis, L. (2023). Control and evaluation of a humanoid robot with rolling contact joints on its lower body. *Frontiers in Robotics and AI*, *10*(October), 1–21. https://doi.org/10.3389/frobt.2023.1164660

Baradat, C., Arakelian, V., Briot, S., & Guegan, S. (2008). Design and Prototyping of a New Balancing Mechanism for Spatial Parallel Manipulators. *Journal of Mechanical Design*, *130*(7), 072305. https://doi.org/10.1115/1.2901057

Beer, J. M., Smarr, C.-A., Chen, T. L., Prakash, A., Mitzner, T. L., Kemp, C. C., & Rogers, W. A. (2012). The Domesticated Robot: Design Guidelines for Assisting Older Adults to Age in Place. *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction*, 335–342. https://doi.org/10.1145/2157689.2157806

Blad, T. W. A., van Ostayen, R. A. J., Herder, J. L., & Tolou, N. (2022). A Statically Balanced Compliant Ortho-Planar Mechanism for Low-Frequency Energy Harvesting. *Journal of Mechanical Design*, *144*(7). https://doi.org/10.1115/1.4053280

Bogue, R. (2017). Domestic robots: Has their time finally come? *Industrial Robot*, *44*(2), 129–136. https://doi.org/10.1108/IR-01-2017-0018

Boisclair, J., Richard, P.-L., Laliberte, T., & Gosselin, C. (2017). Gravity Compensation of Robotic Manipulators Using Cylindrical Halbach Arrays. *IEEE/ASME Transactions on Mechatronics*, *22*(1), 457–464. https://doi.org/10.1109/TMECH.2016.2614386

Bom, J., Bakx, P., Schut, F., & van Doorslaer, E. (2018). The Impact of Informal Caregiving for Older Adults on the Health of Various Types of Caregivers: A Systematic Review. *The Gerontologist*, *59*(5), e629–e642. https://doi.org/10.1093/geront/gny137

Bruzzone, L., & Quaglia, G. (2012). Review article: locomotion systems for ground mobile robots in unstructured environments. *Mechanical Sciences*, *3*(2), 49–62. https://doi.org/10.5194/ms-3-49-2012

Budynas, R. G., & Nisbett, J. K. (2011). *Shigley's Mechanical Engineering Design, 9th Edition*. McGraw-Hill. https://doi.org/10.1007/s13398-014-0173-7.2

Canfield, S. L., Hill, T. W., & Zuccaro, S. G. (2019). Prediction and Experimental Validation of Power Consumption of Skid-Steer Mobile Robots in Manufacturing Environments. *Journal of Intelligent & Robotic Systems*, *94*(3–4), 825–839. https://doi.org/10.1007/s10846-018-0779-7

Chadaj, K., Malczyk, P., & Frączek, J. (2017). A parallel Hamiltonian formulation for forward dynamics of closed-loop multibody systems. *Multibody System Dynamics*, *39*(1–2), 51–77. https://doi.org/10.1007/s11044-016-9531-x

Chadaj, Krzysztof, Malczyk, P., & Fraczek, J. (2017). A Parallel Recursive Hamiltonian Algorithm for Forward Dynamics of Serial Kinematic Chains. *IEEE Transactions on Robotics*, *33*(3), 647–660. https://doi.org/10.1109/TRO.2017.2654507

Chai, R., Niu, H., Carrasco, J., Arvin, F., Yin, H., & Lennox, B. (2024). Design and Experimental Validation of Deep Reinforcement Learning-Based Fast Trajectory Planning and Control for Mobile Robot in Unknown Environment. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(4), 5778–5792. https://doi.org/10.1109/TNNLS.2022.3209154

Chebotareva, E., Safin, R., Hsia, K. H., Carballo, A., & Magid, E. (2020). Person-Following Algorithm Based on Laser Range Finder and Monocular Camera Data Fusion for a Wheeled Autonomous Mobile Robot. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12336 LNAI*(October), 21–33. https://doi.org/10.1007/978-3-030-60337-3_3

Chen, G., & Zhang, S. (2011). Fully-compliant statically-balanced mechanisms without prestressing assembly: concepts and case studies. *Mechanical Sciences*, *2*(2), 169–174.

https://doi.org/10.5194/ms-2-169-2011

Chen, Q., Zhu, S., & Zhang, X. (2015). Improved Inverse Kinematics Algorithm Using Screw Theory for a Six-DOF Robot Manipulator. *International Journal of Advanced Robotic Systems*, *12*, 1. https://doi.org/10.5772/60834

Chen, Y.-L., Cai, Y.-R., & Cheng, M.-Y. (2023). Vision-Based Robotic Object Grasping—A Deep Reinforcement Learning Approach. *Machines*, *11*(2), 275. https://doi.org/10.3390/machines11020275

Chen, Y.-Y., Lin, Y.-H., Kung, C.-C., Chung, M.-H., & Yen, I.-H. (2019). Design and Implementation of Cloud Analytics-Assisted Smart Power Meters Considering Advanced Artificial Intelligence as Edge Analytics in Demand-Side Management for Smart Homes. *Sensors*, *19*(9), 2047. https://doi.org/10.3390/s19092047

Cheng, H., & Ji, G. (2016). Design and implementation of a low cost 3D printed humanoid robotic platform. *2016 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, *855*, 86–91. https://doi.org/10.1109/CYBER.2016.7574801

Corke, P. I. (2007). A Simple and Systematic Approach to Assigning Denavit–Hartenberg Parameters. *IEEE Transactions on Robotics*, *23*(3), 590–594. https://doi.org/10.1109/TRO.2007.896765

Corral, E., García, M. J. G., Castejon, C., Meneses, J., & Gismeros, R. (2020). Dynamic Modeling of the Dissipative Contact and Friction Forces of a Passive Biped-Walking Robot. *Applied Sciences*, *10*(7), 2342. https://doi.org/10.3390/app10072342

Deits, R., & Tedrake, R. (2014). Footstep planning on uneven terrain with mixed-integer convex optimization. *2014 IEEE-RAS International Conference on Humanoid Robots*, *2015-Febru*(November), 279–286. https://doi.org/10.1109/HUMANOIDS.2014.7041373

Diaz-Boladeras, M., Paillacho, D., Angulo, C., Torres, O., Gonzalez-Dieguez, J., & Albo-Canals, J. (2015). Evaluating Group-Robot Interaction in Crowded Public Spaces: A Week-Long Exploratory Study in the Wild with a Humanoid Robot Guiding Visitors Through a Science Museum. *International Journal of Humanoid Robotics*, *12*(4). https://doi.org/10.1142/S021984361550022X

Ding, F., & Liu, C. (2018). Applying coordinate fixed Denavit–Hartenberg method to solve the workspace of drilling robot arm. *International Journal of Advanced Robotic Systems*, *15*(4), 172988141879328. https://doi.org/10.1177/1729881418793283

DiSalvo, C. F., Gemperle, F., Forlizzi, J., & Kiesler, S. (2002). All robots are not created equal: The design and perception of humanoid robot heads. *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, *April 2015*, 321–326. https://doi.org/10.1145/778712.778756

Dong, S., Yuan, Z., & Zhang, F. (2019). A Simplified Method for Dynamic Equation of Robot in Generalized Coordinate System. *Journal of Physics: Conference Series*, *1345*(4), 042077. https://doi.org/10.1088/1742-6596/1345/4/042077

Duffy, B. R. (2003). Anthropomorphism and the social robot. *Robotics and Autonomous Systems*, *42*(3–4), 177–190. https://doi.org/10.1016/S0921-8890(02)00374-3

Dunning, A. G., Tolou, N., & Herder, J. L. (2011). Review Article: Inventory of platforms towards the design of a statically balanced six degrees of freedom compliant precision stage. *Mechanical Sciences*, *2*(2), 157–168. https://doi.org/10.5194/ms-2-157-2011

Emam, S. A. (2020). Generalized Lagrange's equations for systems with general constraints and distributed parameters. *Multibody System Dynamics*, *49*(1), 95–117. https://doi.org/10.1007/s11044-019-09706-z

Englsberger, J., Werner, A., Ott, C., Henze, B., Roa, M. A., Garofalo, G., Burger, R., Beyer, A., Eiberger, O., Schmid, K., & Albu-Schäffer, A. (2015). Overview of the torque-controlled humanoid robot TORO. *IEEE-RAS International Conference on Humanoid Robots*, *2015-Febru*, 916–923. https://doi.org/10.1109/HUMANOIDS.2014.7041473

Featherstone, R. (1999). A Divide-and-Conquer Articulated-Body Algorithm for Parallel O(log(n)) Calculation of Rigid-Body Dynamics. Part 1: Basic Algorithm. *The International Journal of Robotics*

Research, *18*(9), 867–875. https://doi.org/10.1177/02783649922066619

Featherstone, R. (2008). *Rigid Body Dynamics Algorithms*. Springer US. https://doi.org/10.1007/978-1-4899-7560-7

Fink, J. (2012). Anthropomorphism and Human Likeness in the Design of Robots and Human-Robot Interaction. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 7621 LNAI* (Issue October, pp. 199–208). https://doi.org/10.1007/978-3-642-34103-8_20

Fiorio, L., Scalzo, A., Natale, L., Metta, G., & Parmiggiani, A. (2017). A parallel kinematic mechanism for the torso of a humanoid robot: Design, construction and validation. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, *2017-Septe*, 681–688. https://doi.org/10.1109/IROS.2017.8202225

Fomin, A., Antonov, A., Glazunov, V., & Carbone, G. (2021). Dimensional (Parametric) Synthesis of the Hexapod-Type Parallel Mechanism with Reconfigurable Design. *Machines*, *9*(6), 117. https://doi.org/10.3390/machines9060117

French, M. J., & Widden, M. B. (2000). The spring-and-lever balancing mechanism, George Carwardine and the Anglepoise lamp. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, *214*(3), 501–508. https://doi.org/10.1243/0954406001523137

Fuchs, M., Borst, C., Giordano, P. R., Baumann, A., Kraemer, E., Langwald, J., Gruber, R., Seitz, N., Plank, G., Kunze, K., Burger, R., Schmidt, F., Wimboeck, T., & Hirzinger, G. (2009). Rollin' Justin - Design considerations and realization of a mobile platform for a humanoid upper body. *2009 IEEE International Conference on Robotics and Automation*, *May*, 4131–4137. https://doi.org/10.1109/ROBOT.2009.5152464

Fujita, Y., Uenishi, K., Ummadisingu, A., Nagarajan, P., Masuda, S., & Castro, M. Y. (2020). Distributed Reinforcement Learning of Targeted Grasping with Active Vision for Mobile Manipulators. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9712–9719. https://doi.org/10.1109/IROS45743.2020.9341605

Fulmer, T., Reuben, D. B., Auerbach, J., Fick, D. M., Galambos, C., & Johnson, K. S. (2021). Actualizing Better Health And Health Care For Older Adults. *Health Affairs*, *40*(2), 219–225. https://doi.org/10.1377/hlthaff.2020.01470

Ghaedrahmati, R., Raoofian, A., Kamali E., A., & Taghvaeipour, A. (2019). An enhanced inverse dynamic and joint force analysis of multibody systems using constraint matrices. *Multibody System Dynamics*, *46*(4), 329–353. https://doi.org/10.1007/s11044-019-09674-4

Giacaglia, G. E. O., & de Queiroz Lamas, W. (2012). Notes on Newton–Euler formulation of robotic manipulators. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-Body Dynamics*, *226*(1), 61–71. https://doi.org/10.1177/1464419311428034

Goldman, R. (2011). Understanding quaternions. *Graphical Models*, *73*(2), 21–49. https://doi.org/10.1016/j.gmod.2010.10.004

Gonçalves, F., Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022a). *Dynamic Modeling of a Human-Inspired Robot Based on a Newton-Euler Approach* (A. Kecskeméthy & V. Parenti-Castelli (eds.); pp. 79–90). Springer International Publishing. https://doi.org/10.1007/978-3-031-06409-8_8

Gonçalves, F., Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022b). Multibody model of the collaborative human-inspired robot charmie. *Atas Da DSM2022*.

Gonçalves, F., Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022c). A Recursive Algorithm for the Forward Kinematic Analysis of Robotic Systems Using Euler Angles. *Robotics*, *11*(1), 15. https://doi.org/10.3390/robotics11010015

Gosselin, C. (2008). Gravity Compensation, Static Balancing and Dynamic Balancing of Parallel Mechanisms. In L. Wang & J. Xi (Eds.), *Smart Devices and Machines for Advanced Manufacturing* (pp. 27–48). Springer London. https://doi.org/10.1007/978-1-84800-147-3_2

Graf, B., Hans, M., & Schraft, R. D. (2004). Care-O-bot II—Development of a Next Generation Robotic Home Assistant. *Autonomous Robots*, *16*(2), 193–205. https://doi.org/10.1023/B:AURO.0000016865.35796.e9

Grama, L., & Rusu, C. (2018). Adding audio capabilities to TIAGo service robot. *2018 International Symposium on Electronics and Telecommunications (ISETC)*, 1–4. https://doi.org/10.1109/ISETC.2018.8583897

Grotzinger, J. P., Crisp, J., Vasavada, A. R., Anderson, R. C., Baker, C. J., Barry, R., Blake, D. F., Conrad, P., Edgett, K. S., Ferdowski, B., Gellert, R., Gilbert, J. B., Golombek, M., Gómez-Elvira, J., Hassler, D. M., Jandura, L., Litvak, M., Mahaffy, P., Maki, J., ... Wiens, R. C. (2012). Mars Science Laboratory Mission and Science Investigation. *Space Science Reviews*, *170*(1–4), 5–56. https://doi.org/10.1007/s11214-012-9892-2

Hamilton, W. R. (2010). Elements of Quaternions. In *Cambridge Library Collection - Mathematics*. Cambridge University Press. https://doi.org/DOI: 10.1017/CBO9780511707162

Hand, L. N., & Finch, J. D. (1998). *Analytical Mechanics*. Cambridge University Press. https://doi.org/10.1017/CBO9780511801662

Hartenburg, R., & Denavit, J. (1964). Kinematic Synthesis of Linkages. In *New York: McGraw-Hill*.

Heess, N., Wayne, G., Silver, D., Lillicrap, T., Tassa, Y., & Erez, T. (2015). Learning Continuous Control Policies by Stochastic Value Gradients. *Advances in Neural Information Processing Systems*, *2015-Janua*, 2944–2952. http://arxiv.org/abs/1510.09142

Hemingway, E. G., & O'Reilly, O. M. (2018). Perspectives on Euler angle singularities, gimbal lock, and the orthogonality of applied forces and applied moments. *Multibody System Dynamics*, *44*(1), 31–56. https://doi.org/10.1007/s11044-018-9620-0

Hijikata, M., Miyagusuku, R., & Ozaki, K. (2023). Omni Wheel Arrangement Evaluation Method Using Velocity Moments. *Applied Sciences*, *13*(3), 1584. https://doi.org/10.3390/app13031584

Hoetmer, K., Woo, G., Kim, C., & Herder, J. (2010). Negative Stiffness Building Blocks for Statically Balanced Compliant Mechanisms: Design and Testing. *Journal of Mechanisms and Robotics*, *2*(4). https://doi.org/10.1115/1.4002247

Hollerbach, J. M. (1980). A Recursive Lagrangian Formulation of Maniputator Dynamics and a Comparative Study of Dynamics Formulation Complexity. *IEEE Transactions on Systems, Man, and Cybernetics*, *10*(11), 730–736. https://doi.org/10.1109/TSMC.1980.4308393

Hoppe, H. (1997). View-dependent refinement of progressive meshes. *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997*, 189–198. https://doi.org/10.1145/258734.258843

Huang, Z., Li, Q., & Ding, H. (2013). Basics of Screw Theory. In *Mechanisms and Machine Science* (Vol. 6, Issue January, pp. 1–16). https://doi.org/10.1007/978-94-007-4201-7_1

Hussain, Z., & Azlan, N. Z. (2016). KANE's method for dynamic modeling. *2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, *May*, 174–179. https://doi.org/10.1109/I2CACIS.2016.7885310

Hutter, M., Gehring, C., Lauber, A., Gunther, F., Bellicoso, C. D., Tsounis, V., Fankhauser, P., Diethelm, R., Bachmann, S., Bloesch, M., Kolvenbach, H., Bjelonic, M., Isler, L., & Meyer, K. (2017). ANYmal - toward legged robots for harsh environments. *Advanced Robotics*, *31*(17), 918–931. https://doi.org/10.1080/01691864.2017.1378591

Huynh, H. N., Assadi, H., Rivière-Lorphèvre, E., Verlinden, O., & Ahmadi, K. (2020). Modelling the dynamics of industrial robots for milling operations. *Robotics and Computer-Integrated Manufacturing*, *61*(March 2019), 101852. https://doi.org/10.1016/j.rcim.2019.101852

Indri, M., Lachello, L., Lazzero, I., Sibona, F., & Trapani, S. (2019). Smart sensors applications for a new paradigm of a production line. *Sensors (Switzerland)*, *19*(3), 1–36. https://doi.org/10.3390/s19030650

Iriondo, A., Lazkano, E., Susperregi, L., Urain, J., Fernandez, A., & Molina, J. (2019). Pick and Place

Operations in Logistics Using a Mobile Manipulator Controlled with Deep Reinforcement Learning. *Applied Sciences*, *9*(2), 348. https://doi.org/10.3390/app9020348

Ivaldi, S., Peters, J., Padois, V., & Nori, F. (2014). Tools for simulating humanoid robot dynamics: A survey based on user feedback. *2014 IEEE-RAS International Conference on Humanoid Robots*, *2015-Febru*, 842–849. https://doi.org/10.1109/HUMANOIDS.2014.7041462

Jia, W., Yang, G., Gu, L., & Zheng, T. (2017). Dynamics modelling of a mobile manipulator with powered castor wheels. *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 730–735. https://doi.org/10.1109/ICCIS.2017.8274869

Jia, Z., Smith, W., & Peng, H. (2013). Fast analytical models of wheeled locomotion in deformable terrain for mobile robots. *Robotica*, *31*(1), 35–53. https://doi.org/10.1017/S0263574712000069

Jin, Lixing, Lai, Q., Ma, R., Zhang, W., Duan, X., & Li, C. (2022). Zero-Free-Length Elastic System for Gravity Compensation of Parallel Mechanisms With Delta and Delta-Like Architectures. *Journal of Mechanical Design*, *144*(11). https://doi.org/10.1115/1.4054909

Jin, Long, Li, S., Yu, J., & He, J. (2018). Robot manipulator control using neural networks: A survey. *Neurocomputing*, *285*, 23–34. https://doi.org/10.1016/j.neucom.2018.01.002

Johnson, M., Shrewsbury, B., Bertrand, S., Wu, T., Duran, D., Floyd, M., Abeles, P., Stephen, D., Mertins, N., Lesman, A., Carff, J., Rifenburgh, W., Kaveti, P., Straatman, W., Smith, J., Griffioen, M., Layton, B., de Boer, T., Koolen, T., … Pratt, J. (2015). Team IHMC's Lessons Learned from the DARPA Robotics Challenge Trials. *Journal of Field Robotics*, *32*(2), 192–208. https://doi.org/10.1002/rob.21571

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255–260. https://doi.org/10.1126/science.aaa8415

Joshi, S., Kumra, S., & Sahin, F. (2020). Robotic Grasping using Deep Reinforcement Learning. *IEEE International Conference on Automation Science and Engineering*, *2020-Augus*, 1461–1466. https://doi.org/10.1109/CASE48305.2020.9216986

Juang, C.-W., & Chen, D.-Z. (2022). Spring Configurations and Attachment Angles Determination for Statically Balanced Planar Articulated Manipulators. *Journal of Mechanisms and Robotics*, *14*(5). https://doi.org/10.1115/1.4053733

Jung, S., & Hsia, T. C. (1996). A study of neural network control of robot manipulators. *Robotica*, *14*(1), 7–15. https://doi.org/10.1017/s0263574700018890

Kane, T. R., & Levinson, D. A. (1983). The Use of Kane's Dynamical Equations in Robotics. *The International Journal of Robotics Research*, *2*(3), 3–21. https://doi.org/10.1177/027836498300200301

Kaneko, K., Kaminaga, H., Sakaguchi, T., Kajita, S., Morisawa, M., Kumagai, I., & Kanehiro, F. (2019). Humanoid Robot HRP-5P: An Electrically Actuated Humanoid Robot With High-Power and Wide-Range Joints. *IEEE Robotics and Automation Letters*, *4*(2), 1431–1438. https://doi.org/10.1109/LRA.2019.2896465

Kawatsuma, S., Fukushima, M., & Okada, T. (2012). Emergency response by robots to Fukushima-Daiichi accident: summary and lessons learned. *Industrial Robot*, *39*(5), 428–435. https://doi.org/10.1108/01439911211249715

Kemp, C., Edsinger, A., & Torres-Jara, E. (2007). Challenges for robot manipulation in human environments [Grand Challenges of Robotics]. *IEEE Robotics & Automation Magazine*, *14*(1), 20–29. https://doi.org/10.1109/MRA.2007.339604

Khaldi, B., Harrou, F., Cherif, F., & Sun, Y. (2022). Toward Emerging Cubic-Spline Patterns With a Mobile Robotics Swarm System. *IEEE Transactions on Cognitive and Developmental Systems*, *14*(2), 565–577. https://doi.org/10.1109/TCDS.2021.3054997

Khalil, W., Kleinfinger, J., & Gautier, M. (1986). Reducing the computational burden of the dynamic models of robots. *Proceedings. 1986 IEEE International Conference on Robotics and Automation*,

3, 525–531. https://doi.org/10.1109/ROBOT.1986.1087680

Khalil, Wisama. (2011). Dynamic Modeling of Robots Using Newton-Euler Formulation. In *Lecture Notes in Electrical Engineering: Vol. 89 LNEE* (Issue January 2011, pp. 3–20). https://doi.org/10.1007/978-3-642-19539-6_1

Kim, H.-S., Park, J., Bae, M., Park, D., Park, C., Do, H. M., Choi, T., Kim, D.-H., & Kyung, J. (2022). Advanced 2-DOF Counterbalance Mechanism Based on Gear Units and Springs to Minimize Required Torques of Robot Arm. *IEEE Robotics and Automation Letters*, *7*(3), 6320–6326. https://doi.org/10.1109/LRA.2022.3168936

Kim, J., Mishra, A. K., Limosani, R., Scafuro, M., Cauli, N., Santos-Victor, J., Mazzolai, B., & Cavallo, F. (2019). Control strategies for cleaning robots in domestic applications: A comprehensive review. *International Journal of Advanced Robotic Systems*, *16*(4), 172988141985743. https://doi.org/10.1177/1729881419857432

Klamt, T., Kamedula, M., Karaoguz, H., Kashiri, N., Laurenzi, A., Lenz, C., Leonardis, D., Mingo Hoffman, E., Muratore, L., Pavlichenko, D., Porcini, F., Rodriguez, D., Ren, Z., Schilling, F., Schwarz, M., Solazzi, M., Felsberg, M., Frisoli, A., Gustmann, M., … Holmquist, K. (2019). Flexible Disaster Response of Tomorrow: Final Presentation and Evaluation of the CENTAURO System. *IEEE Robotics and Automation Magazine*, *26*(4), 59–72. https://doi.org/10.1109/MRA.2019.2941248

Klug, C., Schmalstieg, D., Gloor, T., & Arth, C. (2019). A Complete Workflow for Automatic Forward Kinematics Model Extraction of Robotic Total Stations Using the Denavit-Hartenberg Convention. *Journal of Intelligent & Robotic Systems*, *95*(2), 311–329. https://doi.org/10.1007/s10846-018-0931-4

Kober, J., Bagnell, J. A., & Peters, J. (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, *32*(11), 1238–1274. https://doi.org/10.1177/0278364913495721

Kobilarov, M. (2014). Nonlinear Trajectory Control of Multi-body Aerial Manipulators. *Journal of Intelligent & Robotic Systems*, *73*(1–4), 679–692. https://doi.org/10.1007/s10846-013-9934-3

Koch, W., Mancuso, R., West, R., & Bestavros, A. (2018). Reinforcement Learning for UAV Attitude Control. *ACM Transactions on Cyber-Physical Systems*, *3*(2). https://doi.org/10.1145/3301273

Kuipers, J. (2012). Quaternions and Rotation Sequences. In *Quaternions and Rotation Sequences*. Princeton University Press. https://doi.org/10.7546/giq-1-2000-127-143

Kuppens, P. R., Bessa, M. A., Herder, J. L., & Hopkins, J. B. (2021). Compliant Mechanisms That Use Static Balancing to Achieve Dramatically Different States of Stiffness. *Journal of Mechanisms and Robotics*, *13*(2). https://doi.org/10.1115/1.4049438

Lacasse, M.-A., Lachance, G., Boisclair, J., Ouellet, J., & Gosselin, C. (2013). On the design of a statically balanced serial robot using remote counterweights. *2013 IEEE International Conference on Robotics and Automation*, 4189–4194. https://doi.org/10.1109/ICRA.2013.6631169

Leal-Naranjo, J. A., Ceccarelli, M., & Torres-San Miguel, C. R. (2017). Mechanical Design of a Prosthetic Human Arm and its Dynamic Simulation. In *Advances in Intelligent Systems and Computing* (Vol. 540, Issue November, pp. 482–490). https://doi.org/10.1007/978-3-319-49058-8_52

Leal Ghezzi, T., & Campos Corleta, O. (2016). 30 Years of Robotic Surgery. *World Journal of Surgery*, *40*(10), 2550–2557. https://doi.org/10.1007/s00268-016-3543-9

Lee, D., & Seo, T. (2017). Lightweight Multi-DOF Manipulator With Wire-Driven Gravity Compensation Mechanism. *IEEE/ASME Transactions on Mechatronics*, *22*(3), 1308–1314. https://doi.org/10.1109/TMECH.2017.2681102

Lee, J., & Cho, B.-K. (2023). Design of 9-DOF humanoid arms inspired by the human's inner shoulder to enhance versatility and workspace. *Robotics and Autonomous Systems*, *166*, 104447. https://doi.org/10.1016/j.robot.2023.104447

Lehnert, C., English, A., McCool, C., Tow, A. W., & Perez, T. (2017). Autonomous Sweet Pepper Harvesting for Protected Cropping Systems. *IEEE Robotics and Automation Letters*, *2*(2), 872–

879. https://doi.org/10.1109/LRA.2017.2655622

Lemburg, J., De Gea Fernández, J., Eich, M., Mronga, D., Kampmann, P., Vogt, A., Aggarwal, A., Shi, Y., & Kirchner, F. (2011). AILA - Design of an autonomous mobile dual-arm robot. *Proceedings - IEEE International Conference on Robotics and Automation*, 5147–5153. https://doi.org/10.1109/ICRA.2011.5979775

Li, Chengshu, Xia, F., Martin-Martin, R., & Savarese, S. (2019). HRL4IN: Hierarchical Reinforcement Learning for Interactive Navigation with Mobile Manipulators. *Proceedings of Machine Learning Research*, *100*(CoRL), 603–616. http://arxiv.org/abs/1910.11432

Li, Chong. (2019). Deep Reinforcement Learning. In *Reinforcement Learning for Cyber-Physical Systems* (pp. 125–154). Chapman and Hall/CRC. https://doi.org/10.1201/9781351006620-6

Li, F., Jiang, Q., Quan, W., Song, R., & Li, Y. (2019). Manipulation Skill Acquisition for Robotic Assembly using Deep Reinforcement Learning. *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, *2019-July*, 13–18. https://doi.org/10.1109/AIM.2019.8868579

Li, S., Zhang, Y., & Jin, L. (2017). Kinematic Control of Redundant Manipulators Using Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2243–2254. https://doi.org/10.1109/TNNLS.2016.2574363

Liang, C.-J., Kamat, V. R., & Menassa, C. C. (2020). Teaching robots to perform quasi-repetitive construction tasks through human demonstration. *Automation in Construction*, *120*(December 2019), 103370. https://doi.org/10.1016/j.autcon.2020.103370

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., & Wierstra, D. (2015). Continuous control with deep reinforcement learning. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*. http://arxiv.org/abs/1509.02971

Liu, L., Dugas, D., Cesari, G., Siegwart, R., & Dube, R. (2020). Robot navigation in crowded environments using deep reinforcement learning. *IEEE International Conference on Intelligent Robots and Systems*, 5671–5677. https://doi.org/10.1109/IROS45743.2020.9341540

Luh, J. Y. S., Walker, M. W., & Paul, R. P. C. (1980). On-Line Computational Scheme for Mechanical Manipulators. *Journal of Dynamic Systems, Measurement, and Control*, *102*(2), 69–76. https://doi.org/10.1115/1.3149599

Madsen, O., Bøgh, S., Schou, C., Andersen, R. S., Damgaard, J. S., Pedersen, M. R., & Krüger, V. (2015). Integration of mobile manipulators in an industrial production. *Industrial Robot*, *42*(1), 11–18. https://doi.org/10.1108/IR-09-2014-0390

Malczyk, P., & Frączek, J. (2012). A divide and conquer algorithm for constrained multibody system dynamics based on augmented Lagrangian method with projections-based error correction. *Nonlinear Dynamics*, *70*(1), 871–889. https://doi.org/10.1007/s11071-012-0503-2

Marques, F., Roupa, I., Silva, M. T., Flores, P., & Lankarani, H. M. (2021). Examination and comparison of different methods to model closed loop kinematic chains using Lagrangian formulation with cut joint, clearance joint constraint and elastic joint approaches. *Mechanism and Machine Theory*, *160*, 104294. https://doi.org/10.1016/j.mechmachtheory.2021.104294

Martini, A., Troncossi, M., Carricato, M., & Rivola, A. (2015). Static balancing of a parallel kinematics machine with Linear-Delta architecture: theory, design and numerical investigation. *Mechanism and Machine Theory*, *90*, 128–141. https://doi.org/10.1016/j.mechmachtheory.2015.03.003

Massoud, M., Abdellatif, A., & Atia, M. R. A. (2021). Mechatronic Design and Path planning optimization for an Omni wheeled mobile robot for indoor applications. *31st International Conference on Computer Theory and Applications, ICCTA 2021 - Proceedings*, 90–98. https://doi.org/10.1109/ICCTA54562.2021.9916626

Menzel, D. H., & Peshkin, M. (1955). Fundamental Formulas of Physics. *American Journal of Physics*, *23*(8), 549–550. https://doi.org/10.1119/1.1934100

Mishra, D. K., Thomas, A., Kuruvilla, J., Kalyanasundaram, P., Prasad, K. R., & Haldorai, A. (2022). Design of mobile robot navigation controller using neuro-fuzzy logic system. *Computers and Electrical Engineering*, *101*(March), 108044.

https://doi.org/10.1016/j.compeleceng.2022.108044

Mitzner, T. L., Boron, J. B., Fausset, C. B., Adams, A. E., Charness, N., Czaja, S. J., Dijkstra, K., Fisk, A. D., Rogers, W. A., & Sharit, J. (2010). Older adults talk technology: Technology usage and attitudes. *Computers in Human Behavior*, *26*(6), 1710–1721. https://doi.org/10.1016/j.chb.2010.06.020

Mizuuchi, I., Tajima, R., Yoshikai, T., Sato, D., Nagashima, K., Inaba, M., Kuniyoshi, Y., & Inoue, H. (2002). The design and control of the flexible spine of a fully tendon-driven humanoid "Kenta." *IEEE/RSJ International Conference on Intelligent Robots and System*, *3*(October), 2527–2532. https://doi.org/10.1109/IRDS.2002.1041649

Mori, M., MacDorman, K., & Kageki, N. (2012). The Uncanny Valley [From the Field]. *IEEE Robotics & Automation Magazine*, *19*(2), 98–100. https://doi.org/10.1109/MRA.2012.2192811

Muhayyuddin, Moll, M., Kavraki, L., & Rosell, J. (2018). Randomized Physics-Based Motion Planning for Grasping in Cluttered and Uncertain Environments. *IEEE Robotics and Automation Letters*, *3*(2), 712–719. https://doi.org/10.1109/LRA.2017.2783445

Müller, A. (2018). Screw and Lie group theory in multibody kinematics. *Multibody System Dynamics*, *43*(1), 37–70. https://doi.org/10.1007/s11044-017-9582-7

Murphy, R. R., Tadokoro, S., Nardi, D., Jacoff, A., Fiorini, P., Choset, H., & Erkmen, A. M. (2008). Search and Rescue Robotics. In *Springer Handbook of Robotics* (Issue January, pp. 1151–1173). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-30301-5_51

Nagabandi, A., Kahn, G., Fearing, R. S., & Levine, S. (2018). Neural Network Dynamics for Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 7559–7566. https://doi.org/10.1109/ICRA.2018.8463189

Nakamoto, H., & Matsuhira, N. (2017). Development of an arm for collaborative robot equipped with gravity compensation mechanism according to payload. *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 40–45. https://doi.org/10.1109/AIM.2017.8013992

Negrete, M., Savage, J., & Contreras-Toledo, L. A. (2018). A Motion-Planning System for a Domestic Service Robot. *SPIIRAS Proceedings*, *5*(60), 5. https://doi.org/10.15622/sp.60.1

Nguyen, V. L. (2022a). Design of a Compact Gear-Spring Mechanism for Static Balancing of Variable Payloads. *Journal of Mechanical Design*, *144*(12). https://doi.org/10.1115/1.4055121

Nguyen, V. L. (2022b). Gravity Balancing of a Two-Degree-of-Freedom Parallel Robotic Platform With Variable Payloads. *Journal of Mechanical Design*, *145*(2). https://doi.org/10.1115/1.4056258

Nguyen, V. L. (2023). Realization of a Gear-Spring Balancer With Variable Payloads and Its Application to Serial Robots. *Journal of Mechanisms and Robotics*, *15*(4). https://doi.org/10.1115/1.4055739

Nguyen, V. L., Lin, C.-Y., & Kuo, C.-H. (2020). Gravity compensation design of Delta parallel robots using gear-spring modules. *Mechanism and Machine Theory*, *154*, 104046. https://doi.org/10.1016/j.mechmachtheory.2020.104046

Oh, P., Sohn, K., Jang, G., Jun, Y., & Cho, B.-K. (2017). Technical Overview of Team DRC-Hubo@UNLV's Approach to the 2015 DARPA Robotics Challenge Finals. *Journal of Field Robotics*, *34*(5), 874–896. https://doi.org/10.1002/rob.21686

Otten, E. (2003). Inverse and forward dynamics: models of multi–body systems. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, *358*(1437), 1493–1500. https://doi.org/10.1098/rstb.2003.1354

Oztemel, E., & Gursev, S. (2020). Literature review of Industry 4.0 and related technologies. *Journal of Intelligent Manufacturing*, *31*(1), 127–182. https://doi.org/10.1007/s10845-018-1433-8

Paletta, L., Fellner, M., Schüssler, S., Zuschnegg, J., Steiner, J., Lerch, A., Lammer, L., & Prodromou, D. (2018). AMIGO – Towards Social Robot based Motivation for Playful Multimodal Intervention in Dementia. *Proceedings of the 11th PErvasive Technologies Related to Assistive Environments*

Conference, 421–427. https://doi.org/10.1145/3197768.3203182

Pandey, A. K., & Gelin, R. (2018). A Mass-Produced Sociable Humanoid Robot: Pepper: The First Machine of Its Kind. *IEEE Robotics & Automation Magazine*, *25*(3), 40–48. https://doi.org/10.1109/MRA.2018.2833157

Park, I. W., Kim, J. Y., Lee, J., & Oh, J. H. (2005). Mechanical design of humanoid robot platform KHR-3 (KAIST humanoid robot - 3: HUBO). *Proceedings of 2005 5th IEEE-RAS International Conference on Humanoid Robots*, *2005*(October), 321–326. https://doi.org/10.1109/ICHR.2005.1573587

Pee, L. G., Pan, S. L., & Cui, L. (2019). Artificial intelligence in healthcare robots: A social informatics study of knowledge embodiment. *Journal of the Association for Information Science and Technology*, *70*(4), 351–369. https://doi.org/10.1002/asi.24145

Peek, S. T. M., Luijkx, K. G., Rijnaard, M. D., Nieboer, M. E., Van Der Voort, C. S., Aarts, S., Van Hoof, J., Vrijhoef, H. J. M., & Wouters, E. J. M. (2016). Older Adults' Reasons for Using Technology while Aging in Place. *Gerontology*, *62*(2), 226–237. https://doi.org/10.1159/000430949

Perrusquía, A., & Yu, W. (2020). Human-in-the-Loop Control Using Euler Angles. *Journal of Intelligent & Robotic Systems*, *97*(2), 271–285. https://doi.org/10.1007/s10846-019-01058-2

Pinciroli, C., Talamali, M. S., Reina, A., Marshall, J. A. R., & Trianni, V. (2018). *Simulating Kilobots Within ARGoS: Models and Experimental Validation BT - Swarm Intelligence* (M. Dorigo, M. Birattari, C. Blum, A. L. Christensen, A. Reina, & V. Trianni (eds.); pp. 176–187). Springer International Publishing.

Pitonakova, L., Giuliani, M., Pipe, A., & Winfield, A. (2018). *Feature and Performance Comparison of the V-REP, Gazebo and ARGoS Robot Simulators* (pp. 357–368). https://doi.org/10.1007/978-3-319-96728-8_30

Polydoros, A. S., & Nalpantidis, L. (2017). Survey of Model-Based Reinforcement Learning: Applications on Robotics. *Journal of Intelligent & Robotic Systems*, *86*(2), 153–173. https://doi.org/10.1007/s10846-017-0468-y

Poursina, M., & Anderson, K. S. (2013). An extended divide-and-conquer algorithm for a generalized class of multibody constraints. *Multibody System Dynamics*, *29*(3), 235–254. https://doi.org/10.1007/s11044-012-9324-9

Radford, N. A., Strawser, P., Hambuchen, K., Mehling, J. S., Verdeyen, W. K., Donnan, A. S., Holley, J., Sanchez, J., Nguyen, V., Bridgwater, L., Berka, R., Ambrose, R., Myles Markee, M., Fraser-Chanpong, N. J., McQuin, C., Yamokoski, J. D., Hart, S., Guo, R., Parsons, A., ... Akinyode, J. (2015). Valkyrie: NASA's First Bipedal Humanoid Robot. *Journal of Field Robotics*, *32*(3), 397–419. https://doi.org/10.1002/rob.21560

Ran, C., & Cheng, X. (2016). A Direct and Non-Singular UKF Approach Using Euler Angle Kinematics for Integrated Navigation Systems. *Sensors (Basel, Switzerland)*, *16*(9), 1415. https://doi.org/10.3390/s16091415

Reddy, K. V., Kodati, M., Chatra, K., & Bandyopadhyay, S. (2016). A comprehensive kinematic analysis of the double wishbone and MacPherson strut suspension systems. *Mechanism and Machine Theory*, *105*, 441–470. https://doi.org/10.1016/j.mechmachtheory.2016.06.001

Reinecke, J., Deutschmann, B., Dietrich, A., & Hutter, M. (2020). An Anthropomorphic Robust Robotic Torso for Ventral/Dorsal and Lateral Motion With Weight Compensation. *IEEE Robotics and Automation Letters*, *5*(3), 3876–3883. https://doi.org/10.1109/LRA.2020.2983386

Rocha, C. R., Tonetto, C. P., & Dias, A. (2011). A comparison between the Denavit–Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators. *Robotics and Computer-Integrated Manufacturing*, *27*(4), 723–728. https://doi.org/10.1016/j.rcim.2010.12.009

Rodrigues da Silva, M., Marques, F., Tavares da Silva, M., & Flores, P. (2022). A compendium of contact force models inspired by Hunt and Crossley's cornerstone work. *Mechanism and Machine Theory*, *167*(July 2021), 104501. https://doi.org/10.1016/j.mechmachtheory.2021.104501

Roennau, A., Heppner, G., Nowicki, M., & Dillmann, R. (2014). LAURONV: A versatile six-legged walking robot with advanced maneuverability. *IEEE/ASME International Conference on Advanced*

Intelligent Mechatronics, AIM, *August*, 82–87. https://doi.org/10.1109/AIM.2014.6878051

Roithmayr, C. M., & Hodges, D. H. (2016). *Dynamics: Theory and Application of Kane's Method*. Cambridge University Press. https://doi.org/DOI: 10.1017/CBO9781139047524

Rone, W. S., & Ben-Tzvi, P. (2014). Continuum Robot Dynamics Utilizing the Principle of Virtual Power. *IEEE Transactions on Robotics*, *30*(1), 275–287. https://doi.org/10.1109/TRO.2013.2281564

Rosenthal, D. E., & Sherman, M. A. (1986). High Performance Multibody Simulations Via Symbolic Equation Manipulation and Kane's Method. *Journal of the Astronautical Sciences*, *34*(3), 223–239.

Rostami, S. M. H., Sangaiah, A. K., Wang, J., & Liu, X. (2019). Obstacle avoidance of mobile robots using modified artificial potential field algorithm. *EURASIP Journal on Wireless Communications and Networking*, *2019*(1), 70. https://doi.org/10.1186/s13638-019-1396-2

Roure, F., Moreno, G., Soler, M., Faconti, D., Serrano, D., Astolfi, P., Bardaro, G., Gabrielli, A., Bascetta, L., & Matteucci, M. (2018). *GRAPE: Ground Robot for vineyArd Monitoring and ProtEction*. https://doi.org/10.1007/978-3-319-70833-1_21

Rudnicka, E., Napierała, P., Podfigurna, A., Męczekalski, B., Smolarczyk, R., & Grymowicz, M. (2020). The World Health Organization (WHO) approach to healthy ageing. *Maturitas*, *139*(May), 6–11. https://doi.org/10.1016/j.maturitas.2020.05.018

Russo, A., Sinatra, R., & Xi, F. (2005). Static balancing of parallel robots. *Mechanism and Machine Theory*, *40*(2), 191–202. https://doi.org/10.1016/j.mechmachtheory.2004.06.011

Sangiovanni, B., Incremona, G. P., Piastra, M., & Ferrara, A. (2021). Self-Configuring Robot Path Planning With Obstacle Avoidance via Deep Reinforcement Learning. *IEEE Control Systems Letters*, *5*(2), 397–402. https://doi.org/10.1109/LCSYS.2020.3002852

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, *2*(3), 160. https://doi.org/10.1007/s42979-021-00592-x

Schappler, M., Tappe, S., & Ortmaier, T. (2019). *Resolution of Functional Redundancy for 3T2R Robot Tasks using Two Sets of Reciprocal Euler Angles BT - Advances in Mechanism and Machine Science* (T. Uhl (ed.); pp. 1701–1710). Springer International Publishing.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms*. 1–12. http://arxiv.org/abs/1707.06347

Shabalina, K., Sagitov, A., & Magid, E. (2018). Comparative Analysis of Mobile Robot Wheels Design. *2018 11th International Conference on Developments in ESystems Engineering (DeSE)*, *2018-Septe*, 175–179. https://doi.org/10.1109/DeSE.2018.00041

Shishehgar, M., Kerr, D., & Blake, J. (2018). A systematic review of research into how robotic technology can help older people. *Smart Health*, *7–8*(March), 1–18. https://doi.org/10.1016/j.smhl.2018.03.002

Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2009). *Robotics Modelling, Planning and Control* (1st ed.). Springer-Verlag. https://doi.org/10.1007/978-1-84628-642-1

Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). *Introduction to Autonomous Mobile Robots, second edition*. MIT Press. https://books.google.pt/books?id=4of6AQAAQBAJ

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., & Hassabis, D. (2017). Mastering the game of Go without human knowledge. *Nature*, *550*(7676), 354–359. https://doi.org/10.1038/nature24270

Singh, R. P., VanderVoort, R. J., & Likins, P. W. (1985). Dynamics of flexible bodies in tree topology - A computer-oriented approach. *Journal of Guidance, Control, and Dynamics*, *8*(5), 584–590. https://doi.org/10.2514/3.20026

Smarr, C.-A., Mitzner, T. L., Beer, J. M., Prakash, A., Chen, T. L., Kemp, C. C., & Rogers, W. A. (2014). Domestic Robots for Older Adults: Attitudes, Preferences, and Potential. *International Journal of Social Robotics*, *6*(2), 229–247. https://doi.org/10.1007/s12369-013-0220-0

Stentz, A., Herman, H., Kelly, A., Meyhofer, E., Haynes, G. C., Stager, D., Zajac, B., Bagnell, J. A., Brindza, J., Dellin, C., George, M., Gonzalez-mora, J., Hyde, S., Jones, M., Laverne, M., Likhachev, M., Lister, L., Powers, M., Ramos, O., ... Wellington, C. (2015). *CHIMP , the CMU Highly Intelligent Mobile Platform*. *32*(2), 209–228. https://doi.org/10.1002/rob

Stolfi, D. H., & Danoy, G. (2022). An Evolutionary Algorithm to Optimise a Distributed UAV Swarm Formation System. *Applied Sciences*, *12*(20), 10218. https://doi.org/10.3390/app122010218

Stückler, J., Schwarz, M., & Behnke, S. (2016). Mobile manipulation, tool use, and intuitive interaction for cognitive service robot cosero. *Frontiers Robotics AI*, *3*(NOV). https://doi.org/10.3389/frobt.2016.00058

Sturm, J., Jain, A., Stachniss, C., Kemp, C. C., & Burgard, W. (2010). Operating articulated objects based on experience. *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2739–2744. https://doi.org/10.1109/IROS.2010.5653813

Su, H.-J., Dorozhkin, D. V., & Vance, J. M. (2009). A Screw Theory Approach for the Conceptual Design of Flexible Joints for Compliant Mechanisms. *Journal of Mechanisms and Robotics*, *1*(4), 41001–41009. https://doi.org/10.1115/1.3211024

Takeuchi, K., Yamazaki, Y., & Yoshifuji, K. (2020). Avatar Work: Telework for Disabled People Unable to Go Outside by Using Avatar Robots "OriHime-D" and Its Verification. *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, *March*, 53–60. https://doi.org/10.1145/3371382.3380737

Teng, T. W., Veerajagadheswar, P., Ramalingam, B., Yin, J., Elara Mohan, R., & Gómez, B. F. (2020). Vision Based Wall Following Framework: A Case Study With HSR Robot for Cleaning Application. *Sensors*, *20*(11), 3298. https://doi.org/10.3390/s20113298

Thomas, M. J., Sanjeev, M. M., Sudheer, A. P., & M.L., J. (2020). Comparative study of various machine learning algorithms and Denavit–Hartenberg approach for the inverse kinematic solutions in a 3- PP SS parallel manipulator. *Industrial Robot: The International Journal of Robotics Research and Application*, *47*(5), 683–695. https://doi.org/10.1108/IR-11-2019-0233

Tolou, N., & Herder, J. L. (2009). Concept and modeling of a statically balanced compliant laparoscopic grasper. *Proceedings of the ASME Design Engineering Technical Conference*, *7*(PARTS A AND B), 163–170. https://doi.org/10.1115/DETC2009-86694

Turno, S., & Malczyk, P. (2023). FPGA acceleration of planar multibody dynamics simulations in the Hamiltonian–based divide–and–conquer framework. *Multibody System Dynamics*, *57*(1), 25–53. https://doi.org/10.1007/s11044-022-09860-x

United Nations Department of Economic and Social Affairs - Population Division. (2022). World Population Prospects 2022: Summary of Results. In *World Population Prospects*. UN DESA/POP/2022/TR/NO. 3. https://www.un.org/development/desa/pd/sites/www.un.org.development.desa.pd/files/wpp2022_summary_of_results.pdf

Urrea, C., & Pascal, J. (2018). Design, simulation, comparison and evaluation of parameter identification methods for an industrial robot. *Computers and Electrical Engineering*, *67*, 791–806. https://doi.org/10.1016/j.compeleceng.2016.09.004

Valverde, A., & Tsiotras, P. (2018). Spacecraft Robot Kinematics Using Dual Quaternions. *Robotics*, *7*, 64. https://doi.org/10.3390/robotics7040064

van Leeuwen, K. M., van Loon, M. S., van Nes, F. A., Bosmans, J. E., de Vet, H. C. W., Ket, J. C. F., Widdershoven, G. A. M., & Ostelo, R. W. J. G. (2019). What does quality of life mean to older adults? A thematic synthesis. *PLOS ONE*, *14*(3), e0213263. https://doi.org/10.1371/journal.pone.0213263

Vermeulen, M., & Wisse, M. (2010). Intrinsically Safe Robot Arm: Adjustable Static Balancing and Low Power Actuation. *International Journal of Social Robotics*, *2*(3), 275–288. https://doi.org/10.1007/s12369-010-0048-9

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell,

R., Ewalds, T., Georgiev, P., Oh, J., Horgan, D., Kroiss, M., Danihelka, I., Huang, A., Sifre, L., Cai, T., Agapiou, J. P., Jaderberg, M., ... Silver, D. (2019). Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, *575*(7782), 350–354. https://doi.org/10.1038/s41586-019-1724-z

Waldron, K., & Schmiedeler, J. (2008). Kinematics. In *Springer Handbook of Robotics* (pp. 9–33). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-30301-5_2

Wang, C., Zhang, Q., Tian, Q., Li, S., Wang, X., Lane, D., Petillot, Y., & Wang, S. (2020). Learning Mobile Manipulation through Deep Reinforcement Learning. *Sensors*, *20*(3), 939. https://doi.org/10.3390/s20030939

Wang, J., & Kong, X. (2019). A geometric approach to the static balancing of mechanisms constructed using spherical kinematic chain units. *Mechanism and Machine Theory*, *140*, 305–320. https://doi.org/10.1016/j.mechmachtheory.2019.06.003

Wang, Z., Tian, Q., & Hu, H. (2016). Dynamics of spatial rigid–flexible multibody systems with uncertain interval parameters. *Nonlinear Dynamics*, *84*(2), 527–548. https://doi.org/10.1007/s11071-015-2504-4

Wisspeintner, T., Zant, T. van der, Iocchi, L., & Schiffer, S. (2009). RoboCup@Home: Scientific Competition and Benchmarking for Domestic Service Robots. *Interaction Studies. Social Behaviour and Communication in Biological and Artificial Systems*, *10*(3), 392–426. https://doi.org/10.1075/is.10.3.06wis

Wu, A., Shi, Z., Li, Y., Wu, M., Guan, Y., Zhang, J., & Wei, H. (2015). Formal Kinematic Analysis of a General 6R Manipulator Using the Screw Theory. *Mathematical Problems in Engineering*, *2015*, 1–7. https://doi.org/10.1155/2015/549797

Wurman, P. R., D'Andrea, R., & Mountz, M. (2008). Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses. *AI Magazine*, *29*(1 SE-Articles), 9. https://doi.org/10.1609/aimag.v29i1.2082

Xu, D., Zhao, D., & Yi, J. (2007). Dynamic Model and Control for an Omnidirectional Mobile Manipulator. In T.-J. Tarn, S.-B. Chen, & C. Zhou (Eds.), *Robotic Welding, Intelligence and Automation* (pp. 21–30). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-73374-4_3

Yamamoto, T., Terada, K., Ochiai, A., Saito, F., Asahara, Y., & Murase, K. (2019). Development of Human Support Robot as the research platform of a domestic mobile manipulator. *ROBOMECH Journal*, *6*(1), 4. https://doi.org/10.1186/s40648-019-0132-3

Yamauchi, B. M. (2004). PackBot: A versatile platform for military robotics. In G. R. Gerhart, C. M. Shoemaker, & D. W. Gage (Eds.), *Unmanned Ground Vehicle Technology VI* (Vol. 5422, pp. 228–237). https://doi.org/10.1117/12.538328

Yang, G., Lin, W., Mustafa, S., Pham, C. B., & Yeo, S. (2005). Kinematic design of a 7-DOF cable-driven humanoid arm: a solution-in-nature approach. In *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.* (Vol. 1, Issue February). IEEE. https://doi.org/10.1109/AIM.2005.1501031

Yu, Z., Huang, Q., Ma, G., Chen, X., Zhang, W., Li, J., & Gao, J. (2014). Design and Development of the Humanoid Robot BHR-5. *Advances in Mechanical Engineering*, *2014*(August). https://doi.org/10.1155/2014/852937

Yun, S.-H., Seo, J., Yoon, J., Song, H., Kim, Y.-S., & Kim, Y.-J. (2019). 3-DOF Gravity Compensation Mechanism for Robot Waists with the Variations of Center of Mass. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3565–3570. https://doi.org/10.1109/IROS40897.2019.8968046

Záda, V., & Belda, K. (2017). Application of Hamiltonian mechanics to control design for industrial robotic manipulators. *2017 22nd International Conference on Methods and Models in Automation and Robotics, MMAR 2017*, *4*, 390–395. https://doi.org/doi.org/10.5281/zenodo.1330465

Zhang, J., Shen, C., & Li, R. (2022). A Robotic System Method and Rebar Construction with Off-the-

Shelf Robots. *Computing in Civil Engineering 2021*, 1204–1211. https://doi.org/10.1061/9780784483893.147

Zhou, C., Huang, B., & Fränti, P. (2022). A review of motion planning algorithms for intelligent robots. *Journal of Intelligent Manufacturing*, *33*(2), 387–424. https://doi.org/10.1007/s10845-021-01867-z

Žlajpah, L. (2008). Simulation in robotics. *Mathematics and Computers in Simulation*, *79*(4), 879–897. https://doi.org/10.1016/j.matcom.2008.02.017

Zupan, E., Saje, M., & Zupan, D. (2013). Dynamics of spatial beams in quaternion description based on the Newmark integration scheme. *COMPUTATIONAL MECHANICS*, *51*, 47–64. https://doi.org/10.1007/s00466-012-0703-0

LIST OF PUBLICATIONS

Papers in Scientific Journals

1. Ribeiro, T., **Gonçalves, F.**, Garcia, I. S., Lopes, G., & Ribeiro, A. F. (2021). CHARMIE: A Collaborative Healthcare and Home Service and Assistant Robot for Elderly Care. Applied Sciences, 11(16), 7248. https://doi.org/10.3390/app11167248
2. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022). A Recursive Algorithm for the Forward Kinematic Analysis of Robotic Systems Using Euler Angles. Robotics, 11(1), 15. https://doi.org/10.3390/robotics11010015
3. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2023). Multibody model of the human-inspired robot CHARMIE. Multibody System Dynamics. https://doi.org/10.1007/s11044-023-09933-5
4. da Silva, M. R., Coelho, J., **Gonçalves, F.**, Novais, F., & Flores, P. (2024). Multibody dynamics in robotics with focus on contact events. Robotica, 1–33. https://doi.org/10.1017/S026357472400050X

Proceedings in Scientific Conferences

1. Ribeiro, T., **Goncalves, F.**, Garcia, I., Lopes, G., & Ribeiro, A. F. (2019). Q-Learning for Autonomous Mobile Robot Obstacle Avoidance. In 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) (pp. 1–7). IEEE. https://doi.org/10.1109/ICARSC.2019.8733621
2. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022). Dynamic Modeling of a Human-Inspired Robot Based on a Newton-Euler Approach. In A. Kecskeméthy & V. Parenti-Castelli (Eds.) (pp. 79–90). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-06409-8_8
3. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2022). Multibody model of the collaborative human-inspired robot charmie. In Atas da DSM2022.

Abstracts in Scientific Conferences

1. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2023). Dynamic Modeling and Simulation of the Omnidirectional Locomotion of a Mobile Manipulator Robot with Four Swedish Wheels and an Independent Suspension System. In Proceedings - ECCOMAS Thematic Conference on Multibody Dynamics.
2. **Gonçalves, F.**, Ribeiro, T., Ribeiro, A. F., Lopes, G., & Flores, P. (2024). Development of a Multibody Simulator to Study the CHARMIE Robot. In Book of Abstracts of MMT Symposium.

Additional Papers

1. Ribeiro, T., **Gonçalves, F.**, Marco, L., Eduardo, F., Rui, S., António, D., Nuno, O., Diogo, M., Syed, F., Lopes, G., & Ribeiro, A. F. (2022). LAR@Home 2023 Team Description Paper. Zenodo. https://doi.org/10.5281/zenodo.15178597
2. Ribeiro, T., Bruno, S., Renata, M., Rui, S., Diogo, M., Nuno, O., Garcia, I. S., **Gonçalves, F.**, Lopes, G., & Ribeiro, A. F. (2023). LAR@Home 2024 Team Description Paper. Zenodo. https://doi.org/10.5281/zenodo.15178581
3. Ribeiro, T., Bruno, S., Renata, M., Rui, S., Diogo, M., José, M., Garcia, I. S., **Gonçalves, F.**, Lopes, G., & Ribeiro, A. F. (2024). LAR@Home 2025 Team Description Paper. Zenodo. https://doi.org/10.5281/zenodo.15178542

APPENDIX I - EQUATION FACTORS FOR THE BEVEL GEAR ANALYSIS

The goal of this appendix is to use the mathematical formulation from (Budynas & Nisbett, 2011), which applies the ANSI/AGMA 2003-B97 methodology to analyze the strength of bevel gears. The following set of equations is used to verify the gear's maximum applied bending stress σ_{F} and its resistance to it σ_{FP}, as well as the maximum applied contact stress σ_{H} and the respective strength σ_{HP}.

$$\sigma_{\mathrm{F}} = \frac{W^{\mathrm{t}}}{b}\frac{K_{\mathrm{A}}K_{\upsilon}}{m_{\mathrm{et}}}\frac{Y_{\mathrm{x}}K_{\mathrm{H}\beta}}{Y_{\beta}Y_{\mathrm{J}}} \tag{A1.1}$$

$$\sigma_{\mathrm{FP}} = \frac{\sigma_{\mathrm{Flim}}Y_{\mathrm{NT}}}{S_{\mathrm{F}}K_{\theta}Y_{\mathrm{z}}} \tag{A1.2}$$

$$\sigma_{\mathrm{H}} = Z_{\mathrm{E}}\sqrt{\frac{W^{\mathrm{t}}}{bd_{\mathrm{p}}Z_{\mathrm{I}}}K_{\mathrm{A}}K_{\upsilon}K_{\mathrm{H}\beta}Z_{\mathrm{x}}Z_{\mathrm{xc}}} \tag{A1.3}$$

$$\sigma_{\mathrm{HP}} = \frac{\sigma_{\mathrm{H\,lim}}Z_{\mathrm{NT}}Z_{W}}{S_{\mathrm{H}}K_{\theta}Z_{\mathrm{Z}}} \tag{A1.4}$$

Solving these four equations solves the dimensioning problem, but they require the determination of a set of factors and parameters. The first step was to group the input data for the study in Table A1.1. This includes the properties of the selected gears, the properties of their materials, the maximum loads and speeds determined from the multibody simulation, and the choice of design parameters such as the desired reliability, safety factors, and number of life cycles.

The force applied to the gear's teeth W^{T} can be directly determined from the maximum torque T and the gear pitch diameter d_{p} using the expression:

$$W^{\mathrm{T}} = \frac{2T}{d_{\mathrm{p}}} \tag{A1.5}$$

Table A1.1 – Properties and design parameters of the bevel gears used in the arms of the CHARMIE robot

Description	Variable	Gear 8	Gear 10	Gear 11
Outer transverse modulus [mm]	m_{et}	1.5	1.0	1.0
Pitch Diameter [mm]	d_{p}	33	18	16
Net face width[mm]	b	8.0	4.0	4.7
Number of teeth	z	22	18	16
Young's Modulus [MPa]	E	190000	190000	2850
Poisson's Ratio	v	0.3	0.3	0.4
Allowable Bending Stress Number	σ_{Flim}	91	91	64
Allowable Contact Stress Number	σ_{Hlim}	762	762	64
Torque [N.mm]	T	4400	760	190
Gear Maximum Velocity [rpm]	n	32.52	33.25	56.4
Reliability	R	0.995	0.995	0.995
Bending Safety Factor	s_{F}	2	2	2
Contact Safety Factor	s_{H}	1.414	1.414	1.414
Transmission Accuracy Number	Q_v	6	6	6
Number of life cycles	n_L	10^7	10^7	10^7

With the input parameters defined, it is then necessary to determine the value of the equation factors for each of the gears.

- Overload Factor K_{A}

The overload factor refers to loads exceeding the nominal transmitted loads due to shocks or impacts. Movements will be uniform; therefore, the overload factor is 1 for all gears.

- Load-Distribution Factor $K_{\mathrm{H\beta}}$

The load-distribution factor considers the effects of the mounting type of the gears. In gears 8 and 10 one of the elements is straddle-mounted, while on gear 11 neither of the members is straddle-mounted. This affects the selection of the auxiliary parameter K_{mb}:

$$K_{\mathrm{mb}} = \begin{cases} 1.00 & \text{both member straddle-mounted} \\ 1.10 & \text{one member straddle-mounted} \\ 1.25 & \text{neither member straddle-mounted} \end{cases} \tag{A1.6}$$

The load-distribution factor $K_{\mathrm{H\beta}}$ is then determined with the expression:

$$K_{\mathrm{H}\beta} = K_{\mathrm{mb}} + 5.6(10^{-6})b^2 \tag{A1.7}$$

- Dynamic Factor K_{υ}

The dynamic factor refers to the gear-tooth effects related to speed and load. First, the pitch-line velocity at pitch diameter υ_{et} must be determined:

$$\upsilon_{\mathrm{et}} = 5.236\left(10^{-5}\right) d_{\mathrm{p}} n \tag{A1.8}$$

The dynamic factor K_{υ} is then determined with the expression:

$$K_{\upsilon} = \left(\frac{A + \sqrt{200\upsilon_{\mathrm{et}}}}{A} \right)^{B} \tag{A1.9}$$

where the auxiliary factors B and A depend on the transmission accuracy number Q_{υ}, that accounts for the precision of the spacing of the tooth profiles, and are calculated by the following notation:

$$B = 0.25(12 - Q_{\upsilon})^{2/3} \tag{A1.10}$$

$$A = 50 + 56(1 - B) \tag{A1.11}$$

- Temperature Factor K_{θ}

The temperature factor K_{θ}refers to the effect of temperature on the gear's resistance, and it is directly determined by the expression:

$$K_{\theta} = \begin{cases} 1 & 0^{\mathrm{o}}\mathrm{C} \le \theta \le 120^{\mathrm{o}}\mathrm{C} \\ \dfrac{273 + \theta}{393} & \theta > 120^{\mathrm{o}}\mathrm{C} \end{cases} \tag{A1.12}$$

where θ is the temperature reached by the gears during operation.

- Size Factor for Bending Y_{x}

The size factor for bending Y_{x} is determined by the gear modulus m_{et} by the following expression:

$$Y_{\mathrm{x}} = \begin{cases} 0.5 & m_{\mathrm{et}} < 1.6 \text{ mm} \\ 0.4867 + 0.0008339 m_{\mathrm{et}} & 1.6 \text{ mm} \le m_{\mathrm{et}} \le 50 \text{ mm} \end{cases} \tag{A1.13}$$

- Lengthwise Curvature Factor for Bending Strength Y_{β}

For straight-bevel gears, the value of this factor will always be 1.

- Bending Strength Geometry Factor Y_J

The bending strength geometry factor is defined by the number of teeth on the pair of connecting gears. Since no speed multiplication is required on CHARMIE's bevel gears, there will be a 1:1 transmission ratio, so both gears have the same number of teeth. The bending strength geometry factors for the required pairs of gears in this study are shown in Table A1.2; other values are available in (Budynas & Nisbett, 2011):

Table A1.2 – Bending strength geometry factors for straight-bevel gears with a 20° normal pressure angle and 90° shaft angle.

Gear Pair Number	Number of Teeth on Main Gear	Number of Teeth on Mate Gear	Y_J
Gear 8 and 10	22	22	0.205
Gear 11	16	16	0.180

- Stress-Cycle Factor for Bending Strength Y_{NT}

The stress-cycle factor for bending strength is determined by the number of life cycles the gears must perform. This stress factor is determined by the expression:

$$Y_{NT} = \begin{cases} 2.7 & 10^2 \le n_L \le 10^3 \\ 6.1514 n_L^{-0.1192} & 10^3 \le n_L \le 3\left(10^6\right) \\ 1.683 n_L^{-0.0323} & 3\left(10^6\right) \le n_L \le 10^{10} \quad \text{critical} \\ 1.3558 n_L^{-0.0178} & 3\left(10^6\right) \le n_L \le 10^{10} \quad \text{general} \end{cases} \tag{A1.14}$$

For the CHARMIE robot, all gears were dimensioned for ten million life cycles of operation.

- Reliability Factor Y_Z

The reliability factor Y_Z is influenced by the desired reliability R for the gears, and it is determined by the expression:

$$Y_Z = \begin{cases} 0.50 - 0.25\log(1-R) & 0.99 \le R \le 0.999 \\ 0.70 - 0.15\log(1-R) & 0.90 \le R < 0.99 \end{cases} \tag{A1.15}$$

- Reliability Factor Z_Z

The second reliability factor Z_Z is determined directly by the expression:

$$Z_Z = \sqrt{Y_Z} \tag{A1.16}$$

- Stress-Cycle Factor for Pitting Resistance Z_{NT}

The stress-cycle factor for pitting resistance is determined by the number of life cycles the gears must perform. This stress factor is determined by the formulation:

$$Z_{\mathrm{NT}} = \begin{cases} 2 & 10^3 \le n_{\mathrm{L}} < 10^4 \\ 3.4822 n_{\mathrm{L}}^{-0.0602} & 10^4 \le n_{\mathrm{L}} \le 10^{10} \end{cases} \tag{A1.17}$$

- Hardness-Ratio Factor Z_{W}

The hardness-ratio factor is determined by the hardness of the material of the pair of contacting gears. In CHARMIE, only gears of the same material will be in contact, so this factor will always be 1.

- Size Factor for Pitting Resistance Z_{x}

This factor determines the influence of the gear teeth face width b regarding the pitting resistance. This factor is determined by Equation A1.17.

$$Z_{\mathrm{x}} = \begin{cases} 0.5 & b < 12.7 \text{ mm} \\ 0.00492b + 0.4375 & 12.7 \text{ mm} \le b \le 114.3 \text{ mm} \\ 1 & b > 114.3 \text{ mm} \end{cases} \tag{A1.18}$$

- Crowning Factor for Pitting Z_{xc}

Most bevel gears are crowned, a modification in the flank of the tooth to prevent contact at the ends of the gear tooth due to assembly or manufacturing errors and to concentrate loads on the center of the teeth. However, if the crowning is too large, the contact area becomes smaller, which compromises the teeth's strength. Not crowning the gear teeth also reduced the strength, which is contemplated by the following factor:

$$Z_{\mathrm{xc}} = \begin{cases} 1.5 & \text{properly crowned teeth} \\ 2.0 & \text{larger uncrowned teeth} \end{cases} \tag{A1.19}$$

- Pitting Resistance Geometry Factor Z_{I}

The pitting resistance geometry factor is defined by the number of teeth on the pair of connecting gears. The bending strength geometry factors for the required pairs of gears in this study are shown in Table A1.3; other values are available in (Budynas & Nisbett, 2011).

Table A1.3 – Pitting resistance geometry factors for straight-bevel gears with a 20° normal pressure angle and 90° shaft angle.

Gear Pair Number	Number of Teeth on Main Gear	Number of Teeth on Mate Gear	Z_{I}
Gear 8 and 10	22	22	0.063
Gear 11	16	16	0.053

- Elastic Coefficient for Pitting Resistance Z_{E}

The Elastic Coefficient for pitting resistance refers to the influence of the material's properties (Young's Modulus and Poisson Ratio) regarding the pitting effect, and it can be calculated by the expression:

$$Z_{\mathrm{E}} = \sqrt{\frac{1}{2\pi\left[\frac{1-\nu^2}{E}\right]}} \tag{A1.20}$$

This concludes all factors required to study CHARMIE's bevel gears. The determined factors for all gears were grouped up into Table A1.4.

Table A1.4 – Equation factors for the stress and strength analysis of CHARMIE's bevel gears

Factor	Gear 8	Gear 10	Gear 11	Factor	Gear 8	Gear 10	Gear 11
K_{A}	1.000	1.000	1.000	Y_{z}	1.075	1.075	1.075
$K_{\mathrm{H\beta}}$	1.100	1.100	1.250	Z_{Z}	1.037	1.037	1.037
K_{U}	1.046	1.038	1.042	Z_{NT}	1.320	1.320	1.320
K_{θ}	1.000	1.000	1.000	Z_{W}	1.000	1.000	1.000
Y_{x}	0.500	0.500	0.500	Z_{x}	0.500	0.500	0.500
Y_{β}	1.000	1.000	1.000	Z_{xc}	1.500	1.500	1.500
Y_{J}	0.205	0.205	0.180	Z_{I}	0.063	0.063	0.053
Y_{NT}	0.901	0.901	0.901	Z_{E}	182.3	182.3	23.24

These factors were used alongside equations A1.1—A1.4 to determine the stresses and strengths of CHARMIE's bevel gears (see Table A1.5). The calculated strengths and stresses already consider the input safety factors; since strengths are superior to stresses, all gear choices are valid. The ratio between strength and stress can be used alongside the input safety factors to determine the actual resulting safety factor for each bevel gear.

Table A1.5 – Stresses, strengths, and safety factors of the bevel gear used in CHARMIE's arms

Gear	Bending Stress [MPa]	Bending Strength [MPa]	Bending Safety Factor	Contact Stress [MPa]	Contact Strength [MPa]	Contact Safety Factor
8	62.3	85.86	2.8	678.2	1242	2.6
10	38.5	85.86	4.5	532.6	1242	3.3
11	18.3	26.80	2.9	56.08	57.60	1.5

APPENDIX II - LIST OF PARTS OF THE ROBOT

Table A2.1 – List of parts used for the CHARMIE robot

Part ID	Quantity	Part Name
A000	1	Octagonal Base
A001	4	Wheel Protection Plate
B000	8	Suspension Outer Casing
B001	8	Compression Spring Upper Ring
B002	8	Compression Spring Lower Ring
B003	4	Suspension Lower Arm
B004	4	Motor Assembly Outer Casing
C000	8	Compression Spring Centre Axis
C001	4	Wheel Shaft
D000	12	Base L Corner
D001	2	Central Vertical Pillar
D002	1	Centre Linear Bearing Support Plate
D003	4	Lateral Vertical Pillar
D004	1	Front Left Vertical Pillar
D005	1	Front Right Vertical Pillar
D006	1	Base Linear Actuator Support
E000	1	Vertical Prismatic Joint Rod
E001	1	Horizontal Prismatic Joint Rod
E100	2	Base To Squat Static Balancing Rod
E101	2	Base To Spring Rod
E102	2	Squat Static Balancing To Spring Rod
E103	2	Squat Static Balancing To Hip Rod
E104	1	Hip To Torso Rod
E105	1	Hip To Spring Rod
E106	1	Torso To Spring Rod
F000	1	Squatting Lower Limit Switch Support
F001	1	Squatting Lower Limit Switch Holder
F002	1	Squatting Upper Limit Switch Support
F003	1	Squatting Upper Limit Switch Holder
F100	1	Trunk Flexion Lower Limit Switch Support
F101	1	Trunk Flexion Lower Limit Switch Holder
F102	1	Trunk Flexion Upper Limit Switch Support
F103	1	Trunk Flexion Upper Limit Switch Holder
G000	1	Lateral Pillar Brace Support
H000	1	Hip Left Pillar
H001	1	Hip Right Pillar

Part ID	Quantity	Part Name
H002	1	Hip Central Plate
H003	1	Hip Linear Actuator Support
H004	2	Squatting Static Balancing Tube
H005	2	Hip Lateral Linear Bearing Support Plate
I000	1	Torso Steel Tube
I001	1	Clavicle U-Profile
J000	2	Scapula U-Profile
J001	2	Scapula Metal Plate
J002	2	Scapula Motor Support
J003	2	Scapula Bearing Support
J010	2	Shoulder Motor Support
J011	2	Shoulder Bearing Support
J020	4	Proximal Arm Outer Casing
J021	4	Proximal Arm Motor Support
J022	4	Proximal Arm Bearing Support
J023	2	Proximal Arm Shaft Holder Support
J030	2	Distal Arm Outer Casing
J031	2	Distal Arm Outer Casing Mirror
J032	2	Distal Arm Bearing Support
J033	6	Distal Arm Shaft Holder Support
J040	4	Proximal Forearm Outer Casing
J041	4	Proximal Forearm Motor Support
J042	2	Proximal Forearm Bearing Support
J043	4	Proximal Forearm Shaft Holder Support
J050	2	Distal Forearm Outer Casing
J051	2	Distal Forearm Outer Casing Mirror
J052	4	Distal Forearm 5 mm Bearing Support
J053	2	Distal Forearm 6 mm Bearing Support
J054	2	Distal Forearm Shaft Holder Support
J060	1	Hand Lower Half
J061	1	Hand Lower Half Mirror
J062	2	Hand Top Half
J070	4	Claw
K000	2	Scapula Motor Shaft
K001	2	Shoulder Motor Shaft
K002	2	Proximal Arm Motor Shaft
K003	2	Distal Arm Motor Shaft
K004	2	Proximal Forearm Input Shaft
K005	2	Proximal Forearm Motor Shaft
K006	2	Distal Forearm Motor Shaft
K007	2	Hand Input Shaft
K008	2	Hand Motor Shaft
K009	2	Hand Motor Gear Shaft
K010	2	Hand Auxiliary Gear Shaft
K011	4	Claw Input Shaft
L000	1	Lower Cervical Spine Plate

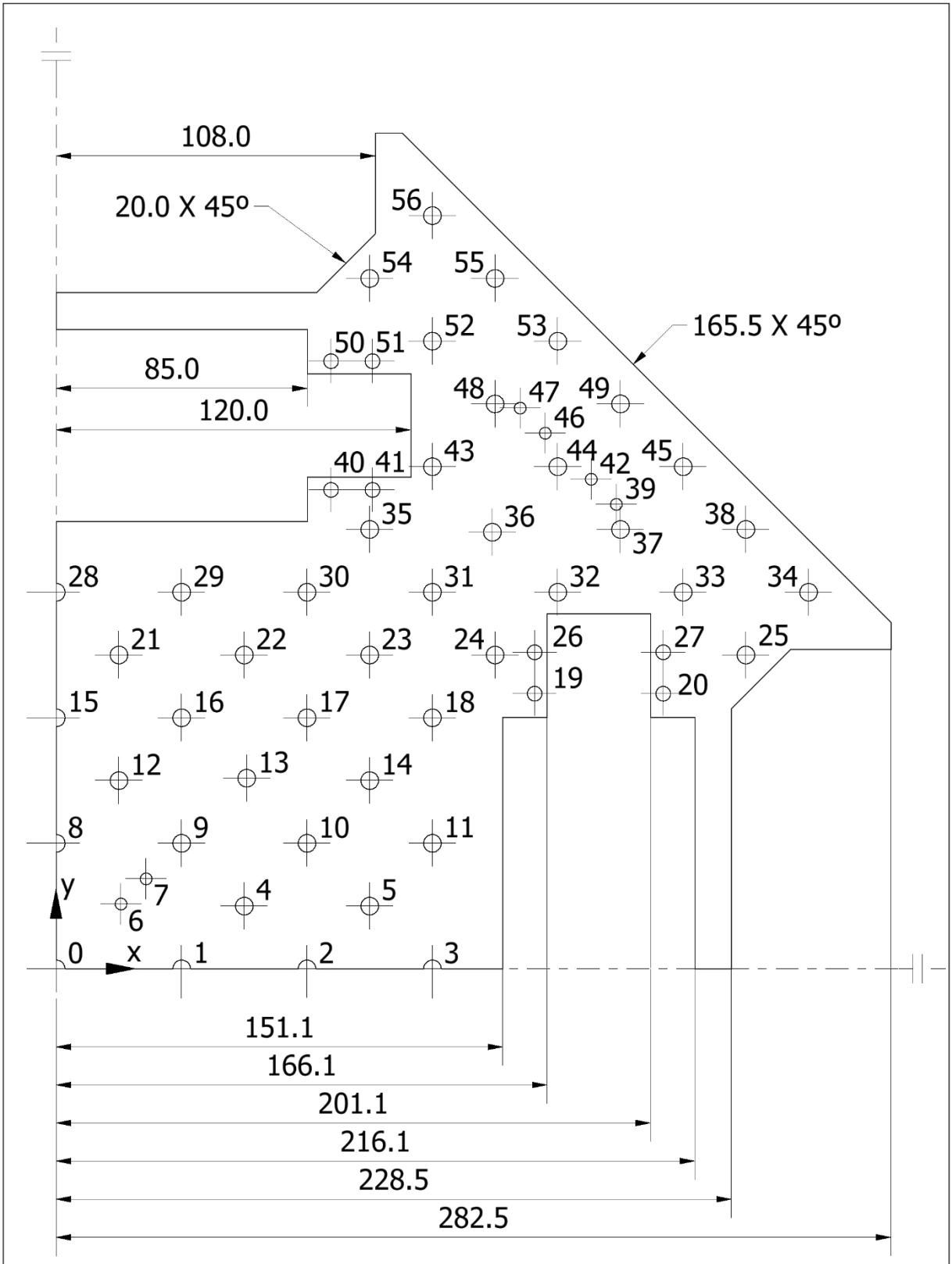
Part ID	Quantity	Part Name
L010	1	Neck Plate
L011	2	Neck L Corner
L020	8	Head L Corner
L021	8	Head L Plate
L022	1	Head Center Tube
L023	4	Head Lateral Tube
L024	4	Head Front-Back Tube
L025	4	Head Vertical Tube
M000	1	Neck Motor Shaft
N000	4	Motor EMG49 24V With Encoders And 49:1 Reduction
N001	8	Motor EC 45flat Ø43.5 50W With Hall Sensors And Cable
N002	4	Motor EC20flat Ø20 5W With Hall Sensors
N003	2	Motor RE13 Ø13 2.5W 4440 rpm/V 2.4V
N004	2	Motor ROB10333 6V 180°
N005	1	Servo ROB10333 Mounting Bracket
N006	1	Metal Servo 25T Round Disc Horn
N100	2	Linear Actuator 12V 1500N 150mm Stroke
N101	4	Mounting Bracket For Linear Actuator
N102	4	Mounting Pin For Linear Actuator
O000	4	Planetary Gearhead GP42C 4 Stages 150:1 Reduction
O001	4	Planetary Gearhead GP32C 3 Stages 51:1 Reduction
O002	4	Planetary Gearhead GP22C 3 Stages 53:1 Reduction
O003	2	Planetary Gearhead GP13A 3 Stages 67:1 Reduction
O010	4	Polymer Mitre Gear Mod1 16T
O011	4	Steel Mitre Gear Mod1 22T
O012	4	Steel Mitre Gear Mod1.5 22T
O020	4	Polymer Spur Gear Mod1.25 12T
O021	4	Polymer Spur Gear Mod1.25 18T
O022	1	Steel Spur Gear Mod0.8 20T
O023	1	Steel Spur Gear Mod0.8 60T
P000	8	LMF8UU Round Flanged Linear Ball Bearing
P001	1	LMK20LUU Rectangular Flanged Linear Ball Bearing
P010	2	SC20UU Pillow Block Linear Ball Bearing
P100	8	KFL08 Flanged Ball Bearing Unit
P101	20	KFL12 Flanged Ball Bearing Unit
P102	2	KFL004 Flanged Ball Bearing Unit
P110	4	UCFL205 Flanged Ball Bearing Unit
P120	2	KP08 Pillow Block Mounted Ball Bearing
P121	4	KP12 Pillow Block Mounted Ball Bearing
P122	4	KP004 Pillow Block Mounted Ball Bearing
Q100	6	SK20 20mm Shaft Support
Q110	4	SHF6 6mm Shaft Support
Q111	6	SHF8 8mm Shaft Support
Q112	2	SHF12 12mm Shaft Support
Q113	5	SHF20 20mm Shaft Support
R000	4	Snap-Action Switch with 16.7mm Lever

Part ID	Quantity	Part Name
W000	4	Aluminium Double Omni Directional Wheel 203mm
W001	4	Aluminium Universal Mounting Hub 10mm Shaft
X000	20	Bronze Flanged Sleeve Bearing 5x9x10
X001	2	Bronze Flanged Sleeve Bearing 6x10x10
Y000	5	Tension Spring L164 Lo220 De35.5 d3.759
Y001	8	Compression Spring Lo148 De19.5 P6.5 d2.108
Z000	202	ISO 7093 - Wide Flat Washer M4
Z001	36	ISO 7093 - Wide Flat Washer M5
Z002	184	ISO 7093 - Wide Flat Washer M6
Z003	66	ISO 7093 - Wide Flat Washer M8
Z100	4	ISO 4762 - Hexagon Socket Head Cap Screw M1.6x10
Z101	12	ISO 4762 - Hexagon Socket Head Cap Screw M2x8
Z102	16	ISO 4762 - Hexagon Socket Head Cap Screw M2.5x10
Z103	8	ISO 4762 - Hexagon Socket Head Cap Screw M2.5x20
Z104	4	ISO 4762 - Hexagon Socket Head Cap Screw M3x6
Z105	5	ISO 4762 - Hexagon Socket Head Cap Screw M3x8
Z106	20	ISO 4762 - Hexagon Socket Head Cap Screw M3x10
Z107	1	ISO 4762 - Hexagon Socket Head Cap Screw M3x12
Z108	32	ISO 4762 - Hexagon Socket Head Cap Screw M3x20
Z109	16	ISO 4762 - Hexagon Socket Head Cap Screw M3x25
Z110	4	ISO 4762 - Hexagon Socket Head Cap Screw M3x35
Z111	2	ISO 4762 - Hexagon Socket Head Cap Screw M3x55
Z112	16	ISO 7045 - Philips Pan Head Screw M4x8
Z113	72	ISO 4762 - Hexagon Socket Head Cap Screw M4x10
Z114	52	ISO 4762 - Hexagon Socket Head Cap Screw M4x12
Z115	56	ISO 4762 - Hexagon Socket Head Cap Screw M4x16
Z116	16	ISO 4762 - Hexagon Socket Head Cap Screw M4x20
Z117	2	ISO 4762 - Hexagon Socket Head Cap Screw M4x25
Z118	2	ISO 4762 - Hexagon Socket Head Cap Screw M4x30
Z119	6	ISO 4762 - Hexagon Socket Head Cap Screw M4x50
Z120	2	ISO 4762 - Hexagon Socket Head Cap Screw M4x55
Z121	2	ISO 4762 - Hexagon Socket Head Cap Screw M4x70
Z122	4	ISO 4762 - Hexagon Socket Head Cap Screw M5x12
Z123	104	ISO 4762 - Hexagon Socket Head Cap Screw M5x16
Z124	12	ISO 4762 - Hexagon Socket Head Cap Screw M5x20
Z125	8	ISO 4762 - Hexagon Socket Head Cap Screw M5x25
Z126	7	ISO 4762 - Hexagon Socket Head Cap Screw M5x30
Z127	104	ISO 4762 - Hexagon Socket Head Cap Screw M6x16
Z128	38	ISO 4762 - Hexagon Socket Head Cap Screw M6x20
Z129	26	ISO 4762 - Hexagon Socket Head Cap Screw M6x25
Z130	4	ISO 4762 - Hexagon Socket Head Cap Screw M6x40
Z131	12	ISO 4762 - Hexagon Socket Head Cap Screw M6x50
Z132	1	ISO 4762 - Hexagon Socket Head Cap Screw M6x60
Z133	6	ISO 4762 - Hexagon Socket Head Cap Screw M8x16
Z134	2	ISO 4762 - Hexagon Socket Head Cap Screw M8x20
Z135	4	ISO 4762 - Hexagon Socket Head Cap Screw M8x25

Part ID	Quantity	Part Name
Z136	8	ISO 4762 - Hexagon Socket Head Cap Screw M8x35
Z137	2	ISO 4762 - Hexagon Socket Head Cap Screw M8x55
Z138	8	ISO 4017 - Hexagon Head Screw M16x35
Z200	8	ISO 4032 - Hexagonal Nut M2.5
Z201	66	ISO 4032 - Hexagonal Nut M3
Z202	212	ISO 4032 - Hexagonal Nut M4
Z203	79	ISO 4032 - Hexagonal Nut M5
Z204	223	ISO 4032 - Hexagonal Nut M6
Z205	22	ISO 4032 - Hexagonal Nut M8
Z206	8	ISO 4032 - Hexagonal Nut M16
Z210	12	DIN 1587 - Hexagon Domed Cap Nut M6
Z300	42	DIN 471 - External Circlip 5 mm Shaft
Z301	12	DIN 471 - External Circlip 6 mm Shaft
Z302	44	DIN 471 - External Circlip 8 mm Shaft
Z303	60	DIN 471 - External Circlip 12 mm Shaft
Z304	24	DIN 471 - External Circlip 20 mm Shaft
Z400	4	ISO 4026 - Hexagon Socket Set Screw M1x3
Z401	4	ISO 4026 - Hexagon Socket Set Screw M2x4
Z402	4	ISO 4026 - Hexagon Socket Set Screw M2.5x4
Z403	4	ISO 4026 - Hexagon Socket Set Screw M4x8
Z404	4	ISO 4026 - Hexagon Socket Set Screw M6x6
Z500	12	Threaded Rod M6x70
Z600	2	Nylon Spacer M3x20
Z601	4	Nylon Spacer M3x25

APPENDIX III - TECHNICAL DRAWINGS

(Large drawings are split into multiple sheets to comply with formatting standards)



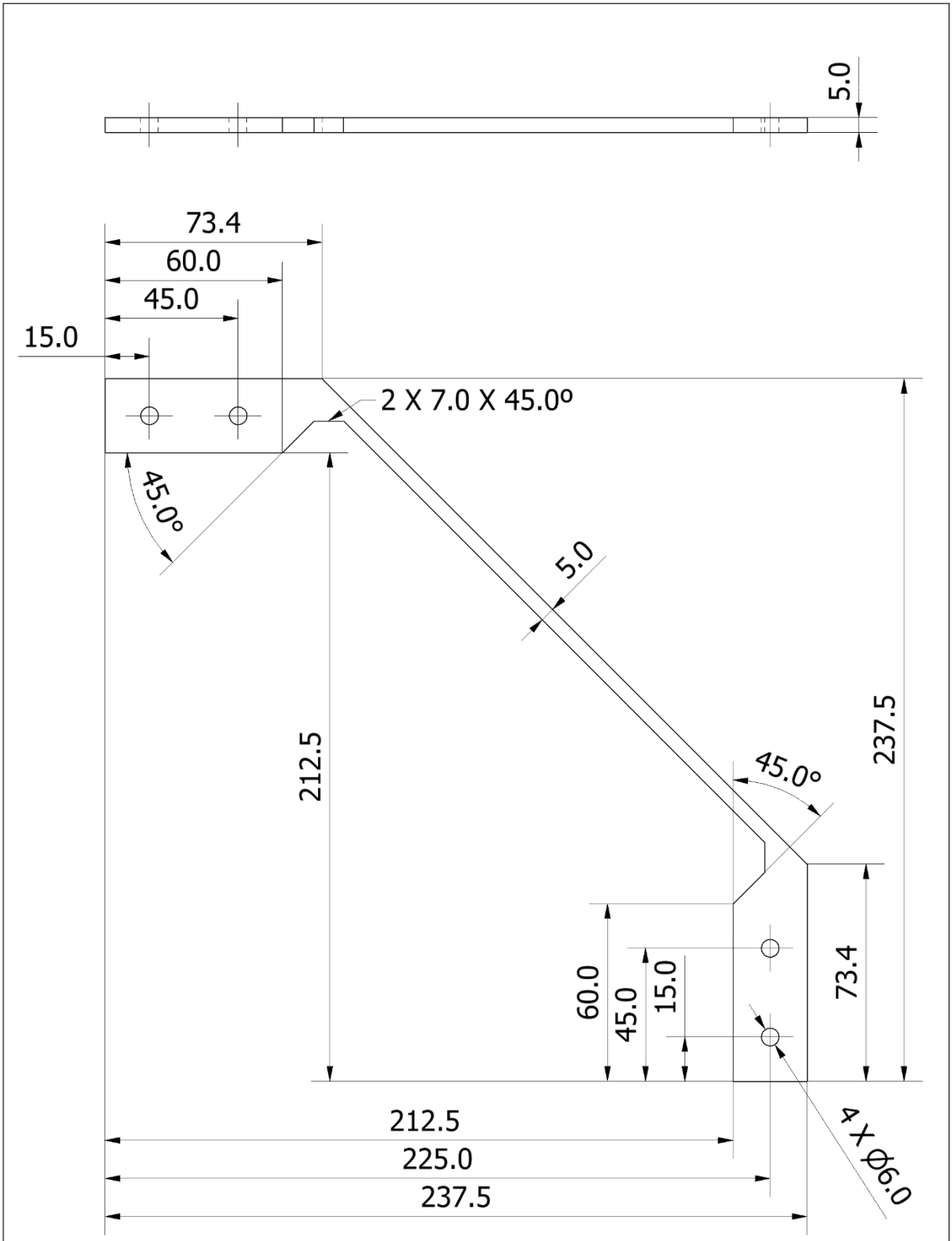
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Octagonal Base		
	A000 - Sheet 1 / 2	Edition	Sheet 1 / 95

Hole Positions							
Hole	ø	x	y	Hole	ø	x	y
0	6.0	0.0	0.0	29	6.0	42.4	127.3
1	6.0	42.4	0.0	30	6.0	84.9	127.3
2	6.0	84.9	0.0	31	6.0	127.3	127.3
3	6.0	127.3	0.0	32	6.0	169.7	127.3
4	6.0	63.6	21.2	33	6.0	212.1	127.3
5	6.0	106.1	21.2	34	6.0	254.6	127.3
6	4.0	21.9	21.9	35	6.0	106.1	148.5
7	4.0	30.4	30.4	36	6.0	148.5	148.5
8	6.0	0	42.4	37	6.0	190.9	148.5
9	6.0	42.4	42.4	38	6.0	233.3	148.5
10	6.0	84.9	42.4	39	4.0	189.5	157.0
11	6.0	127.3	42.4	40	5.0	93.0	161.9
12	6.0	21.2	63.6	41	5.0	107.0	161.9
13	6.0	63.6	63.6	42	4.0	181.0	165.5
14	6.0	106.1	63.3	43	6.0	127.3	169.7
15	6.0	0.0	84.9	44	6.0	169.7	169.7
16	6.0	42.4	84.9	45	6.0	212.1	169.7
17	6.0	84.9	84.9	46	4.0	165.5	181.0
18	6.0	127.3	84.9	47	4.0	157.0	189.5
19	5.0	161.9	93.0	48	6.0	148.5	190.9
20	5.0	205.4	93.0	49	6.0	190.9	190.9
21	6.0	21.2	106.1	50	5.0	93.0	205.4
22	6.0	63.4	106.1	51	5.0	107.0	205.4
23	6.0	106.1	106.1	52	6.0	127.3	212.1
24	6.0	148.5	106.1	53	6.0	169.7	212.1
25	6.0	233.3	106.1	54	6.0	106.1	233.3
26	5.0	161.9	107.0	55	6.0	148.5	233.3
27	5.0	205.4	107.0	56	6.0	127.3	254.6
28	6.0	0.0	127.3	-	-	-	-

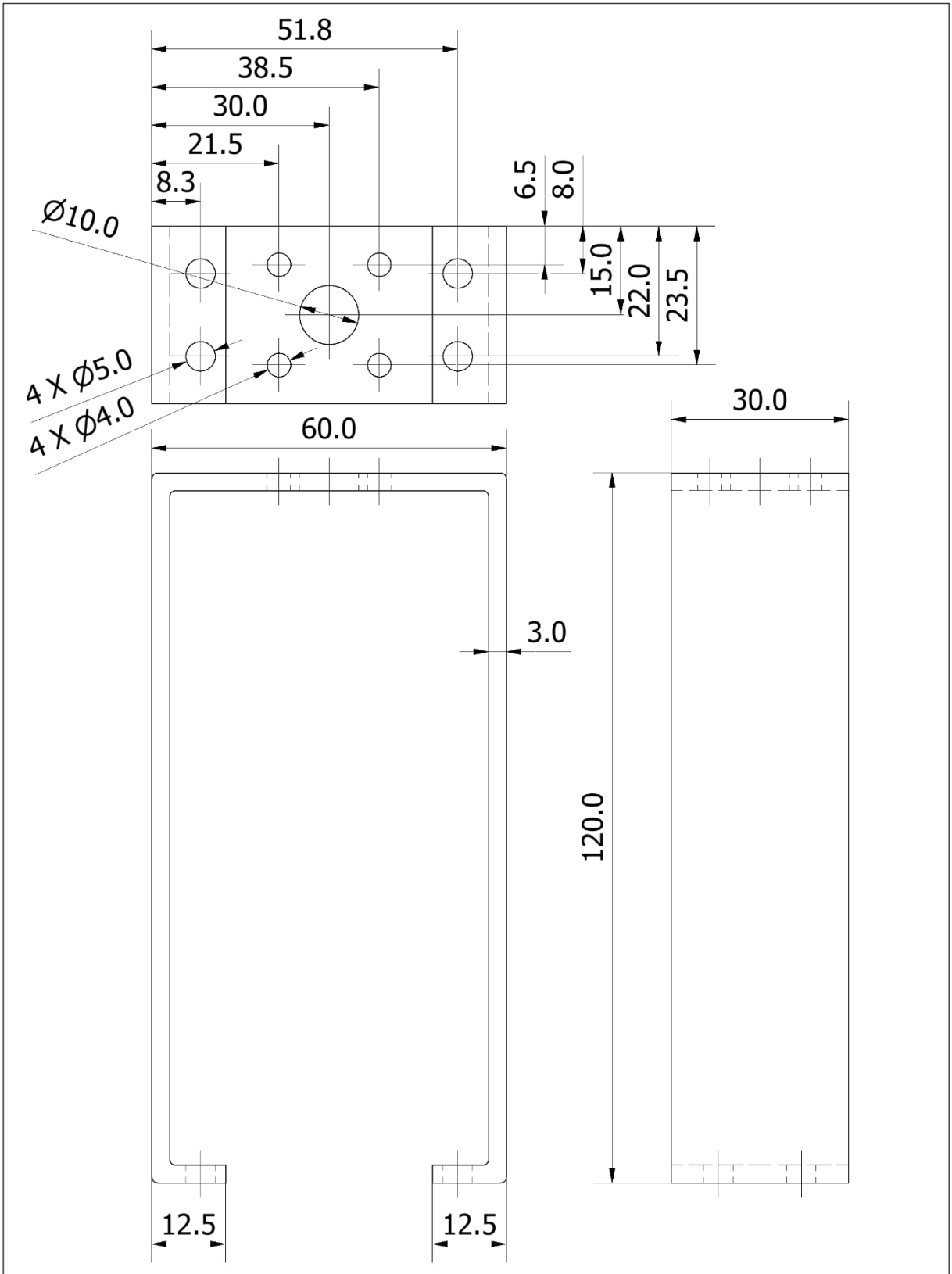
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Octagonal Base		
	A000 - Sheet 2 / 2	Edition	Sheet 2 / 95



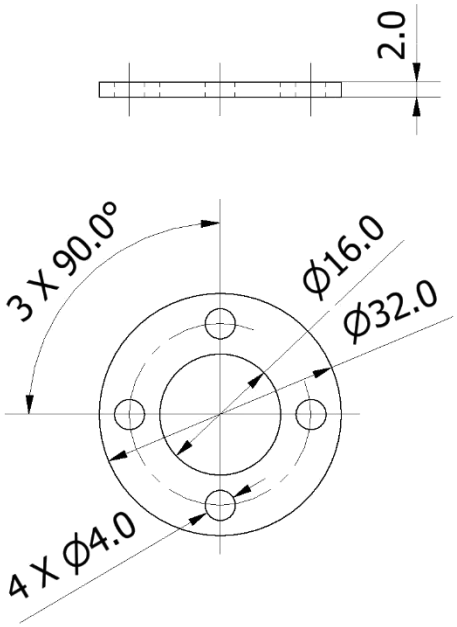
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Wheel Protection Plate		
	A001	Edition	Sheet 3 / 95



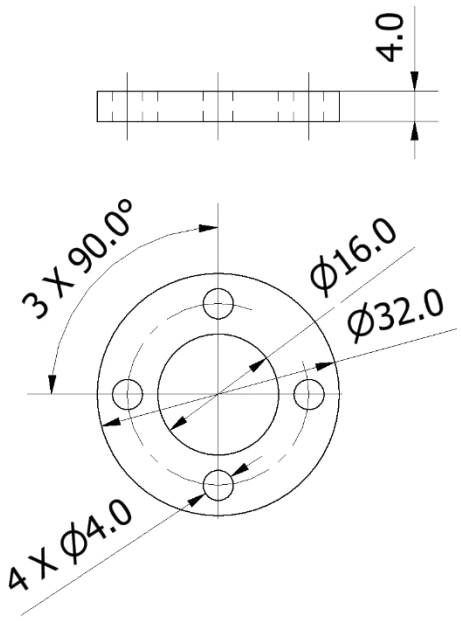
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Suspension Outer Casing		
	B000	Edition	Sheet 4 / 95



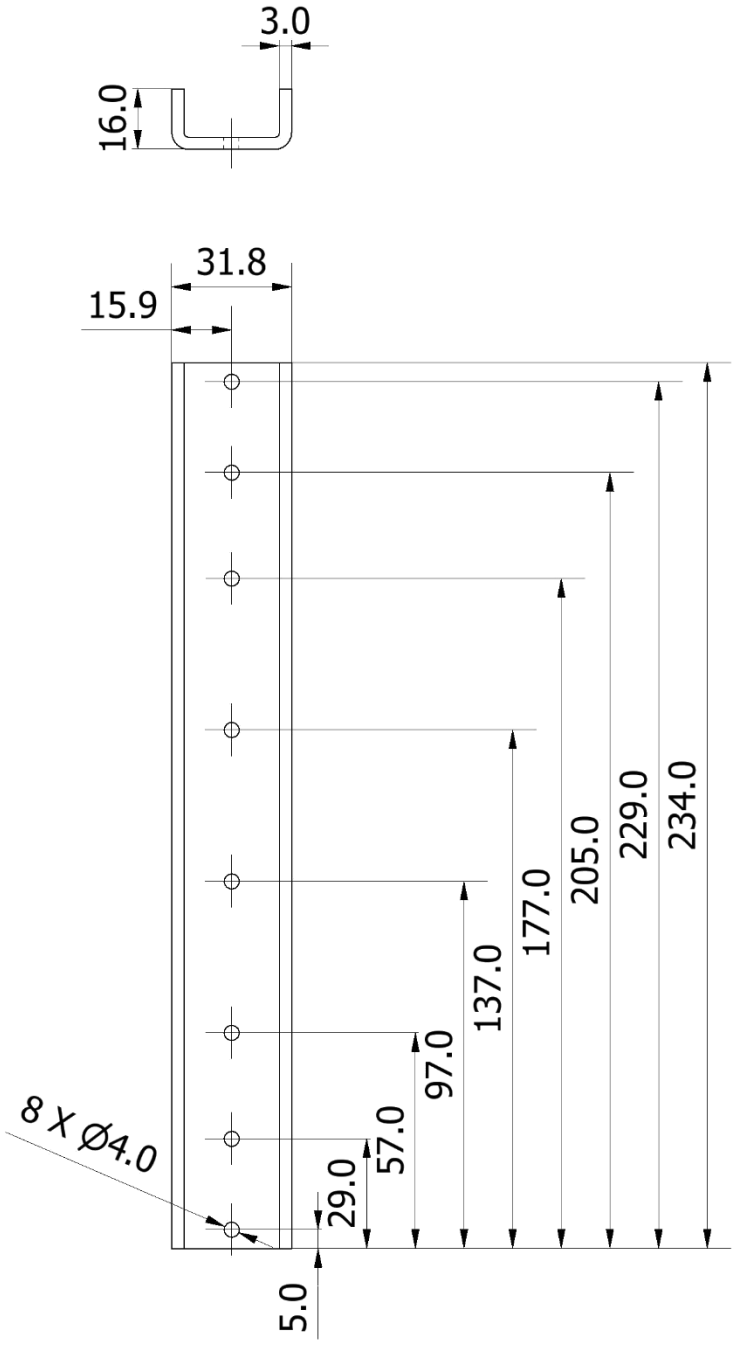
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Compression Spring Upper Ring		
	B001	Edition	Sheet 5 / 95



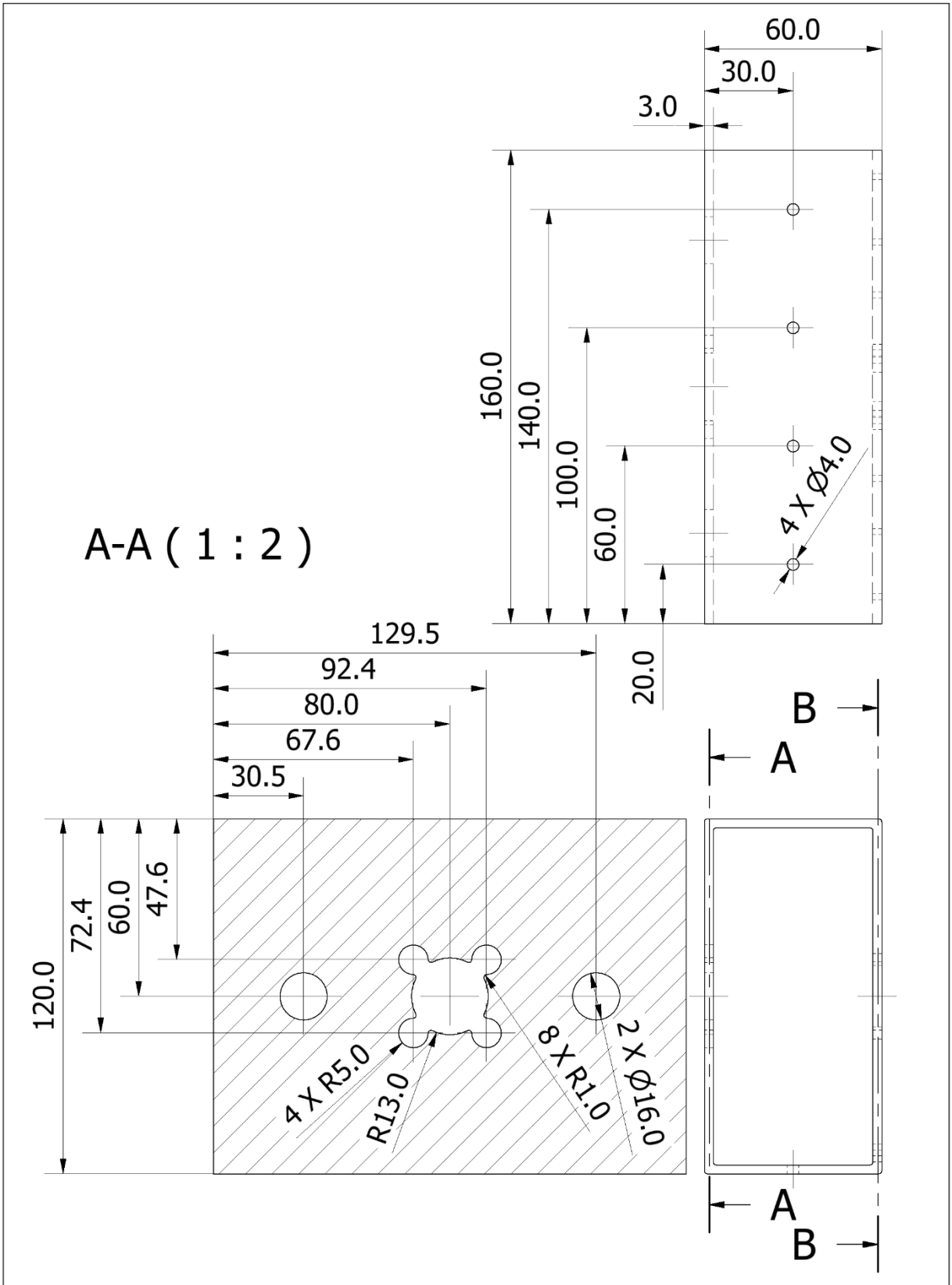
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Compression Spring Lower Ring		
	B002	Edition	Sheet 6 / 95



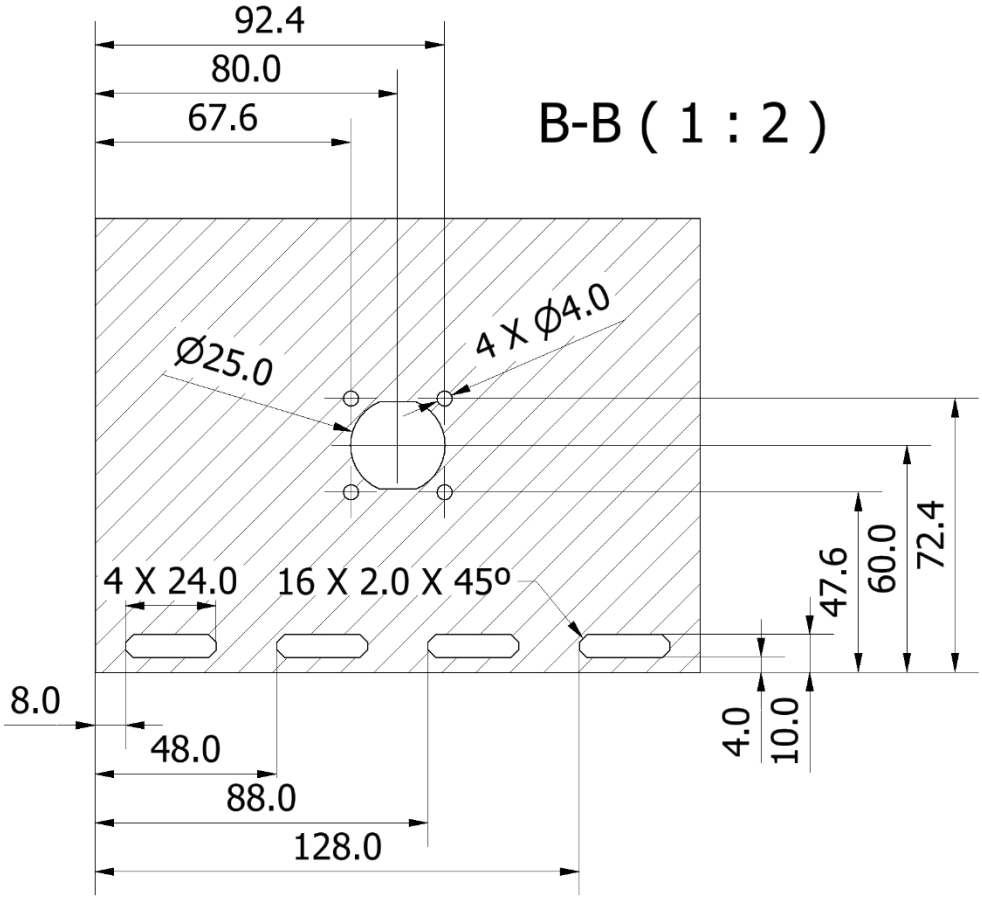
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Suspension Lower Arm		
	B003	Edition	Sheet 7 / 95



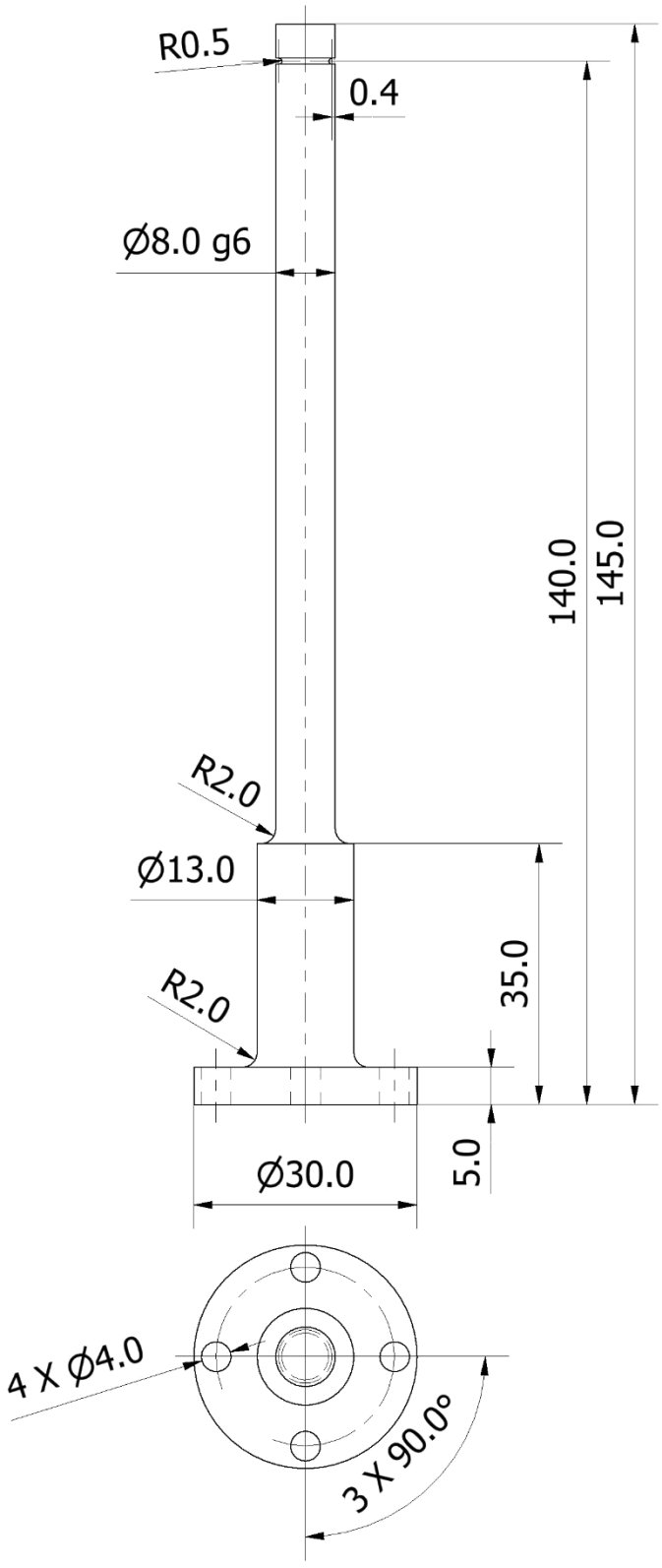
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Motor Assembly Outer Casing		
	B004 - Sheet 1 / 2	Edition	Sheet 8 / 95



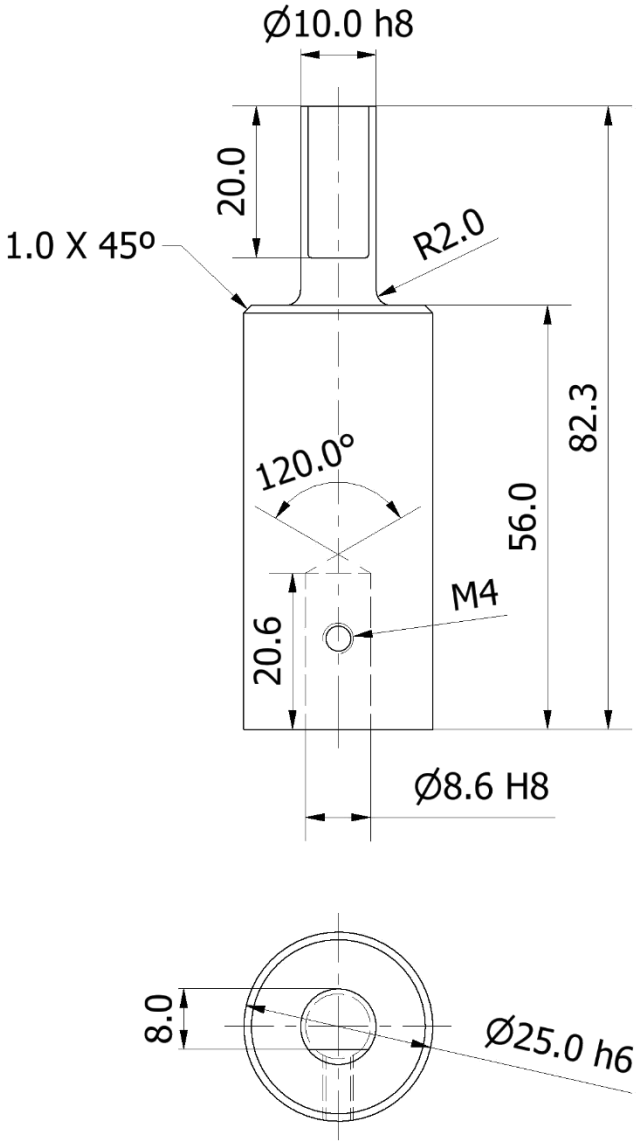
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Motor Assembly Outer Casing		
	B004 - Sheet 2 / 2	Edition	Sheet 9 / 95



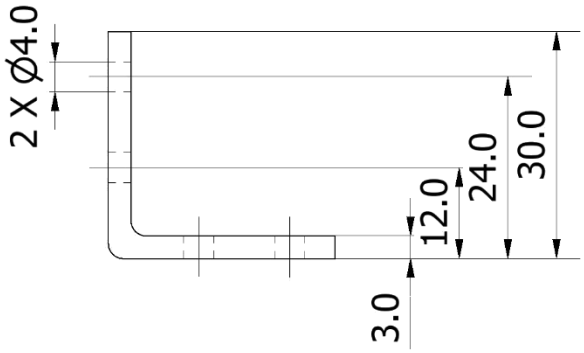
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

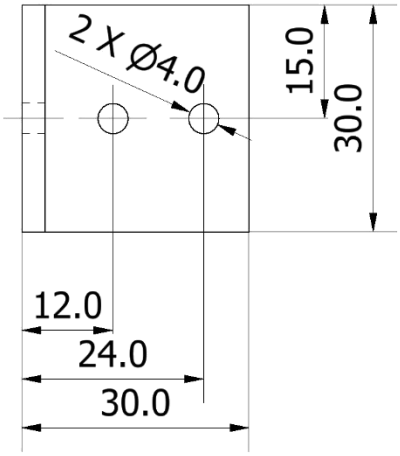
Universidade do Minho Laboratório de Automação e Robótica	Compression Spring Centre Axis		
	C000	Edition	Sheet 10 / 95



Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

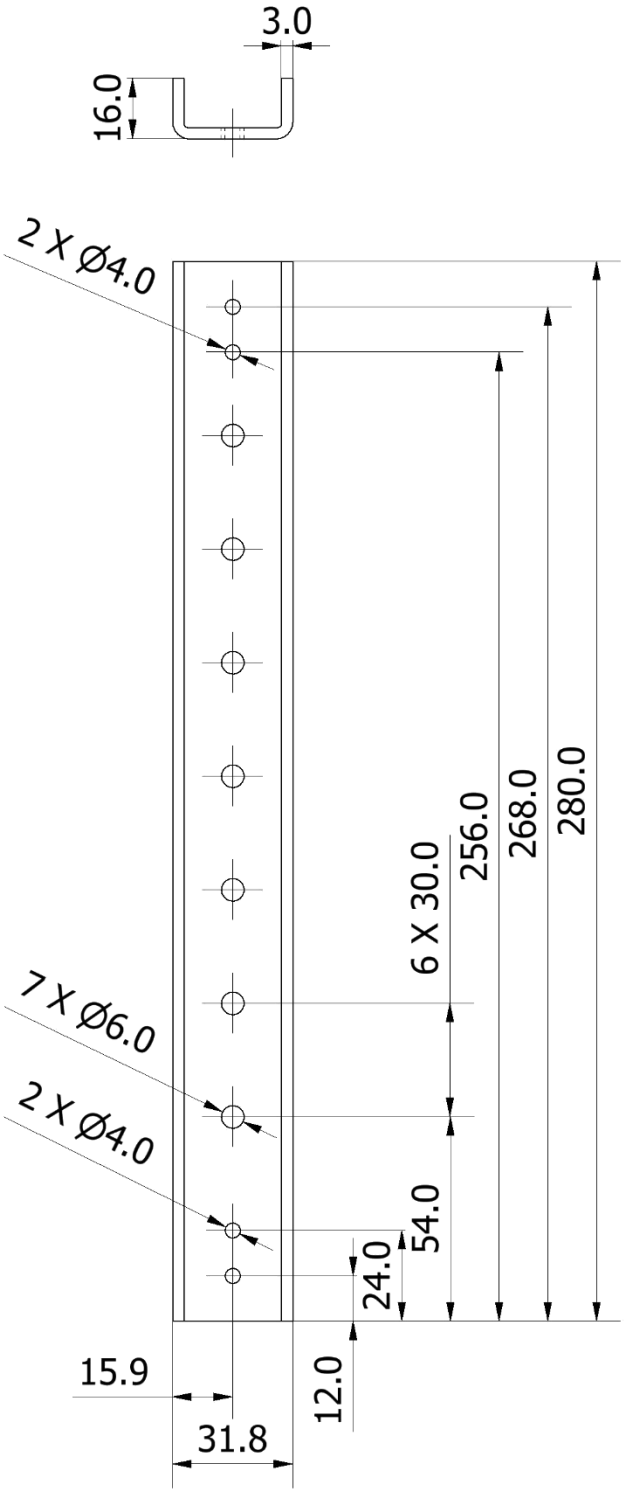
Universidade do Minho Laboratório de Automação e Robótica	Wheel Shaft		
	C001	Edition	Sheet 11 / 95





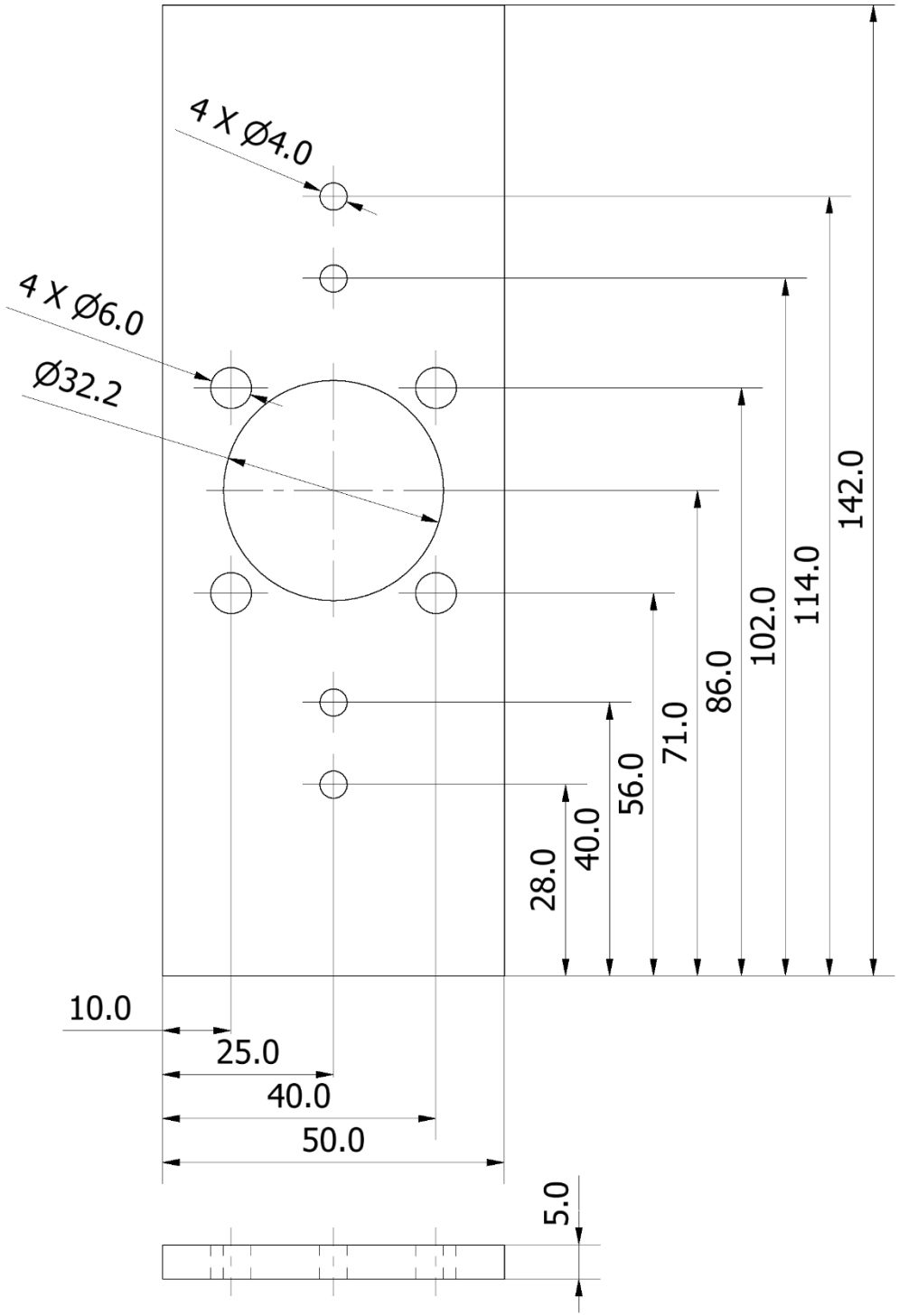
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Base L Corner		
	D000	Edition	Sheet 12 / 95



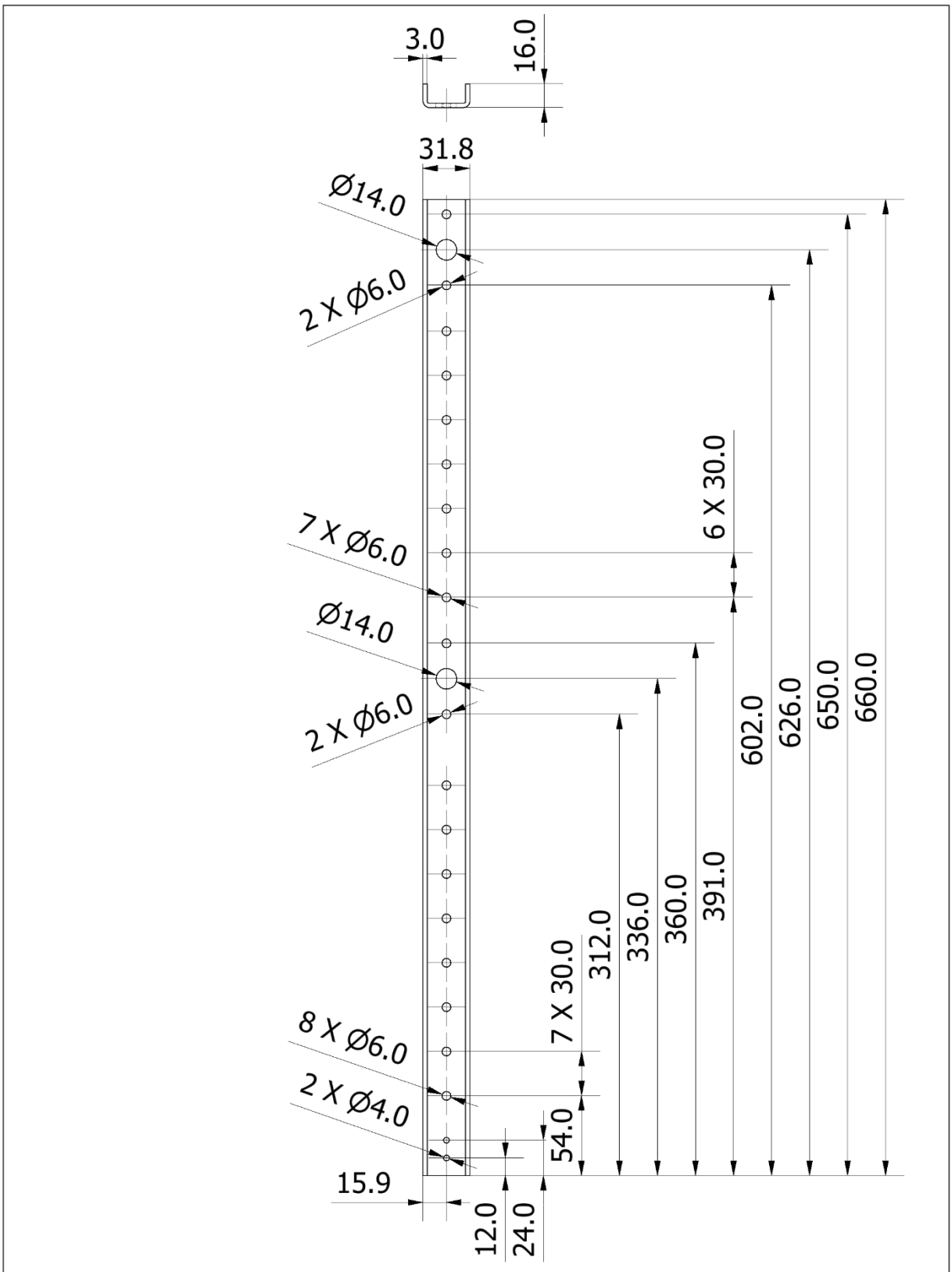
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Central Vertical Pillar		
	D001	Edition	Sheet 13 / 95



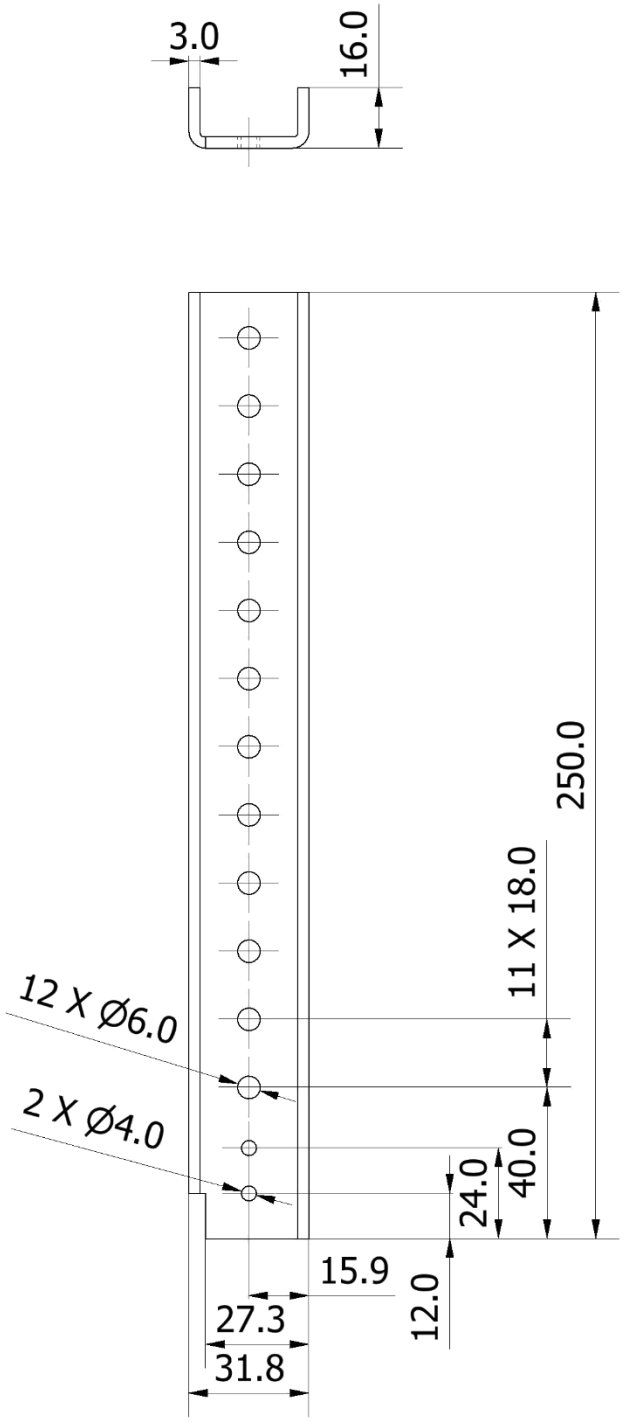
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Centre Linear Bearing Support Plate		
	D002	Edition	Sheet 14 / 95



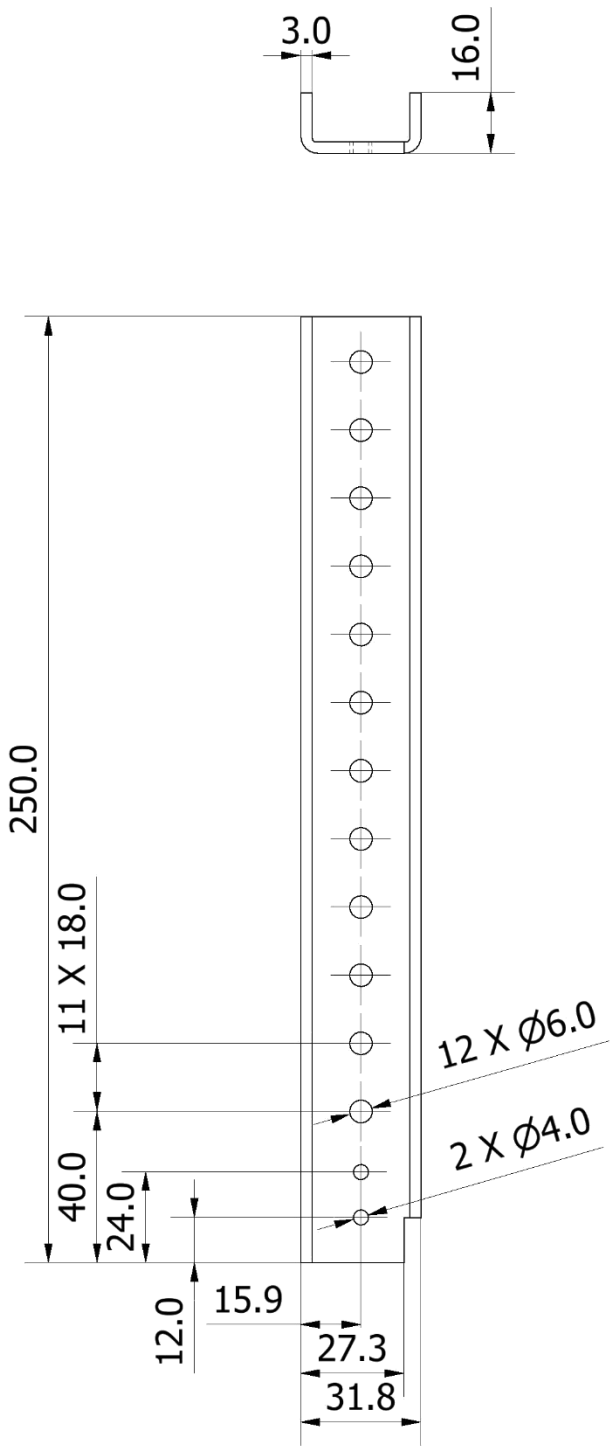
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 4

Universidade do Minho Laboratório de Automação e Robótica	Lateral Vertical Pillar		
	D003	Edition	Sheet 15 / 95



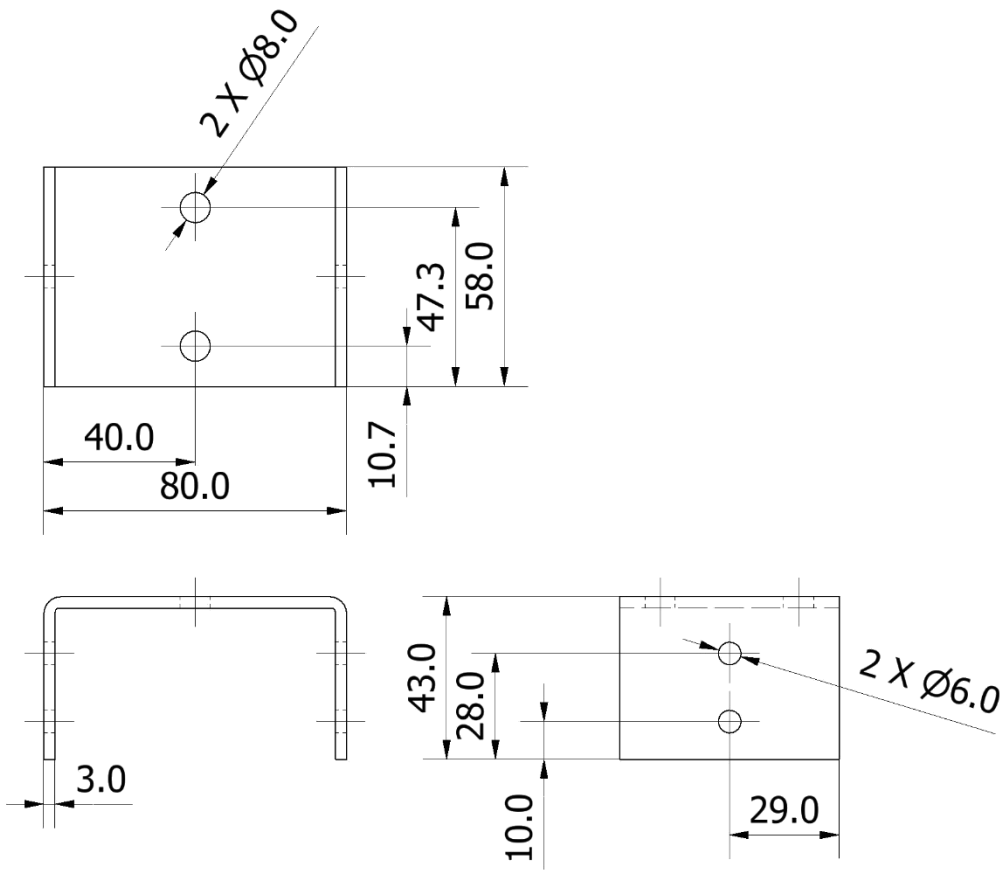
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Front Left Vertical Pillar		
	D004	Edition	Sheet 16 / 95



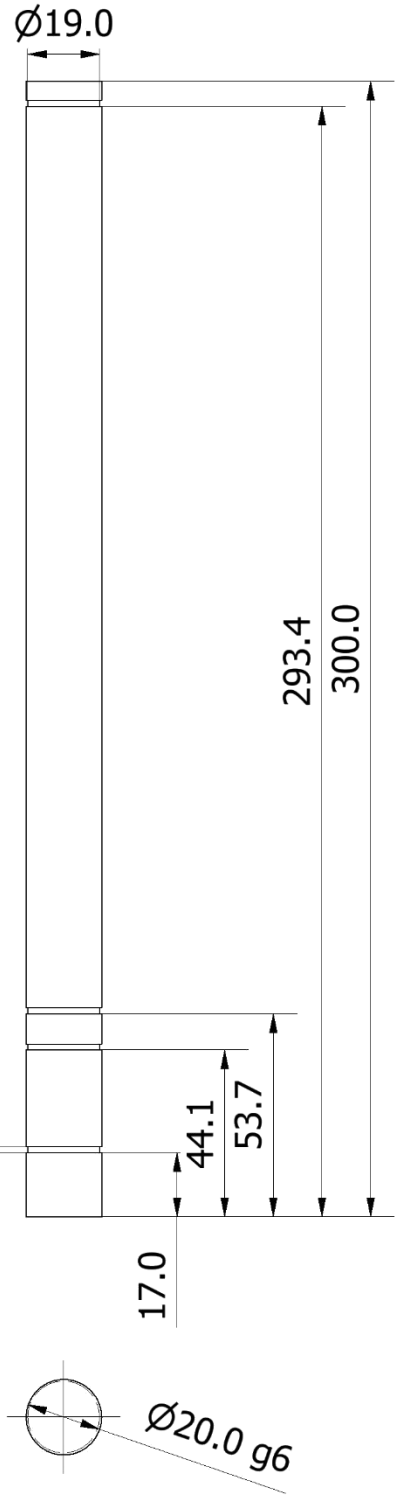
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Front Right Vertical Pillar		
	D005	Edition	Sheet 17 / 95



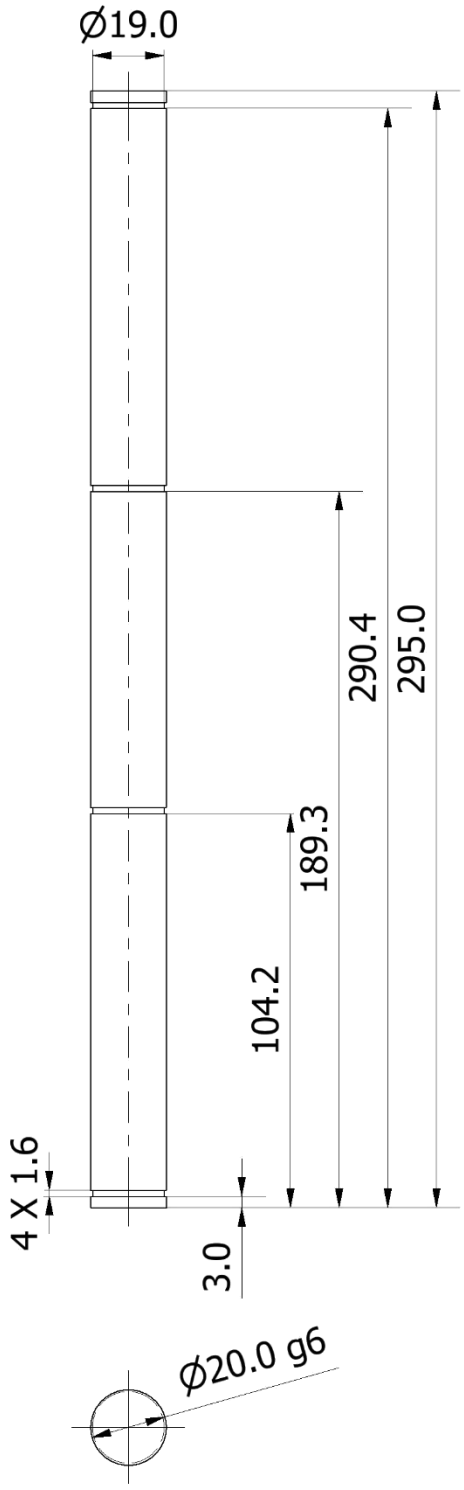
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Base Linear Actuator Support		
	D006	Edition	Sheet 18 / 95



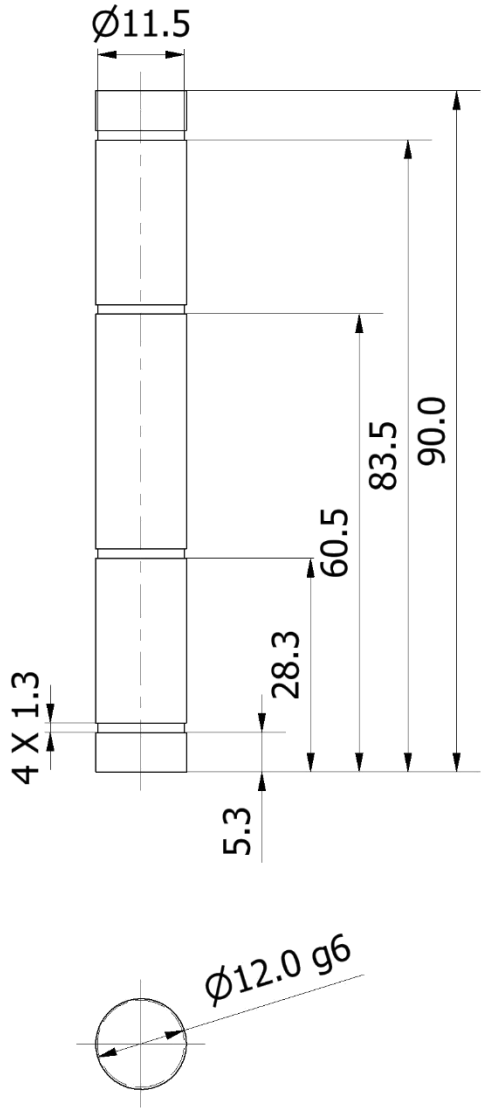
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Vertical Prismatic Joint Rod		
	E000	Edition	Sheet 19 / 95



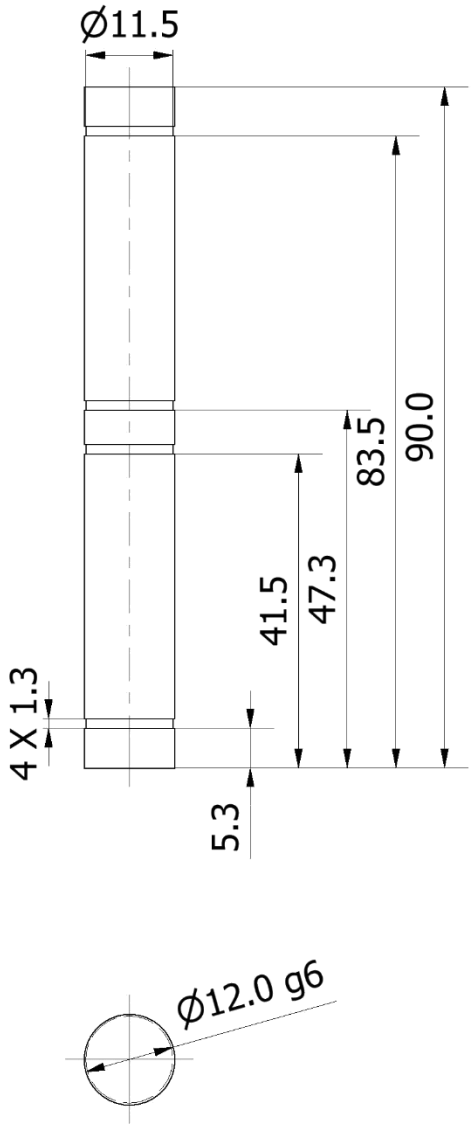
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Horizontal Prismatic Joint Rod		
	E001	Edition	Sheet 20 / 95



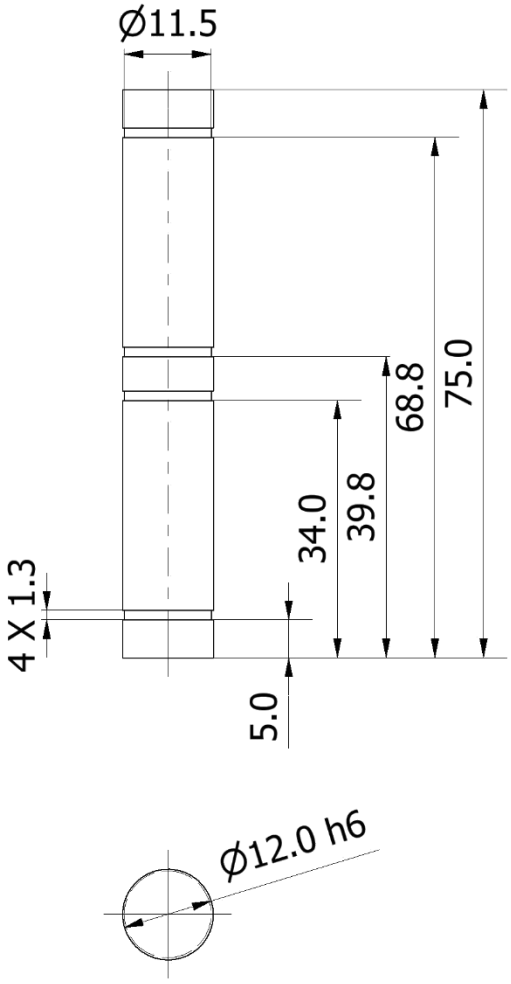
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Base To Squat Static Balancing Joint Rod		
	E100	Edition	Sheet 21 / 95



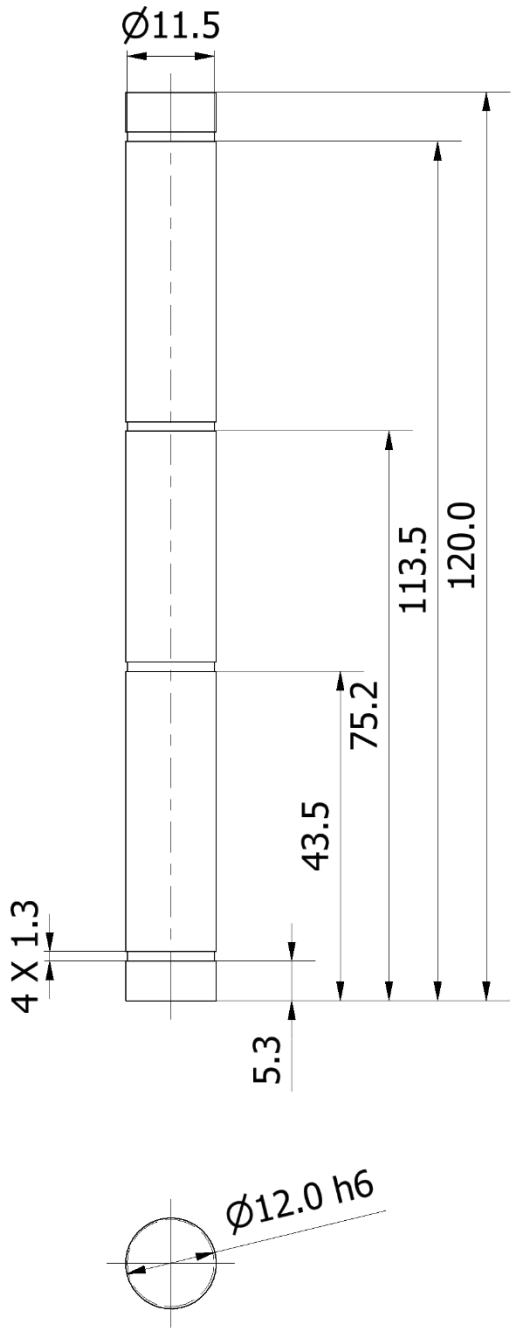
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Base To Spring Rod		
	E101	Edition	Sheet 22 / 95



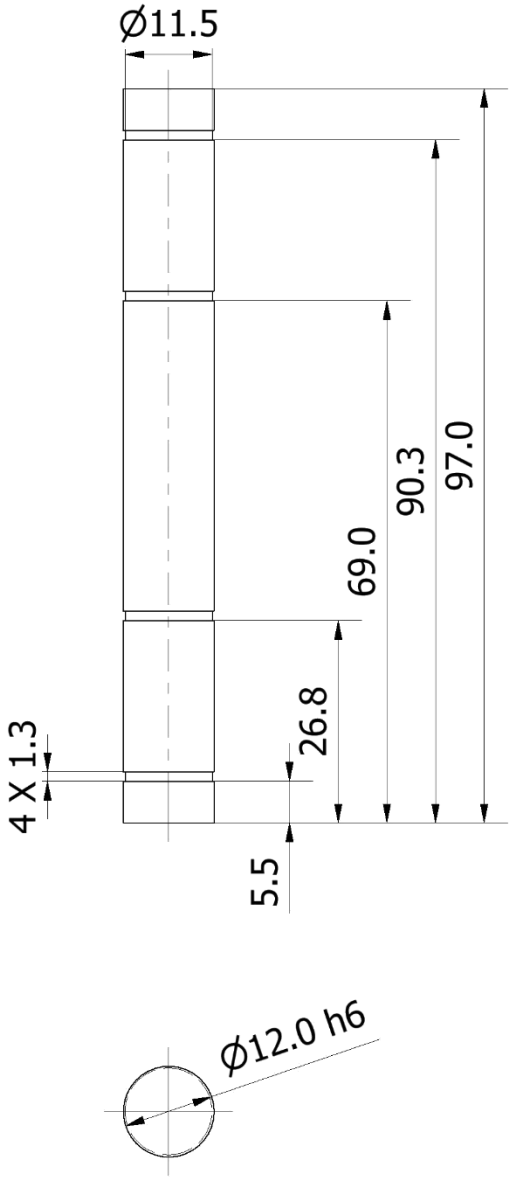
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squat Static Balancing To Spring Rod		
	E102	Edition	Sheet 23 / 95



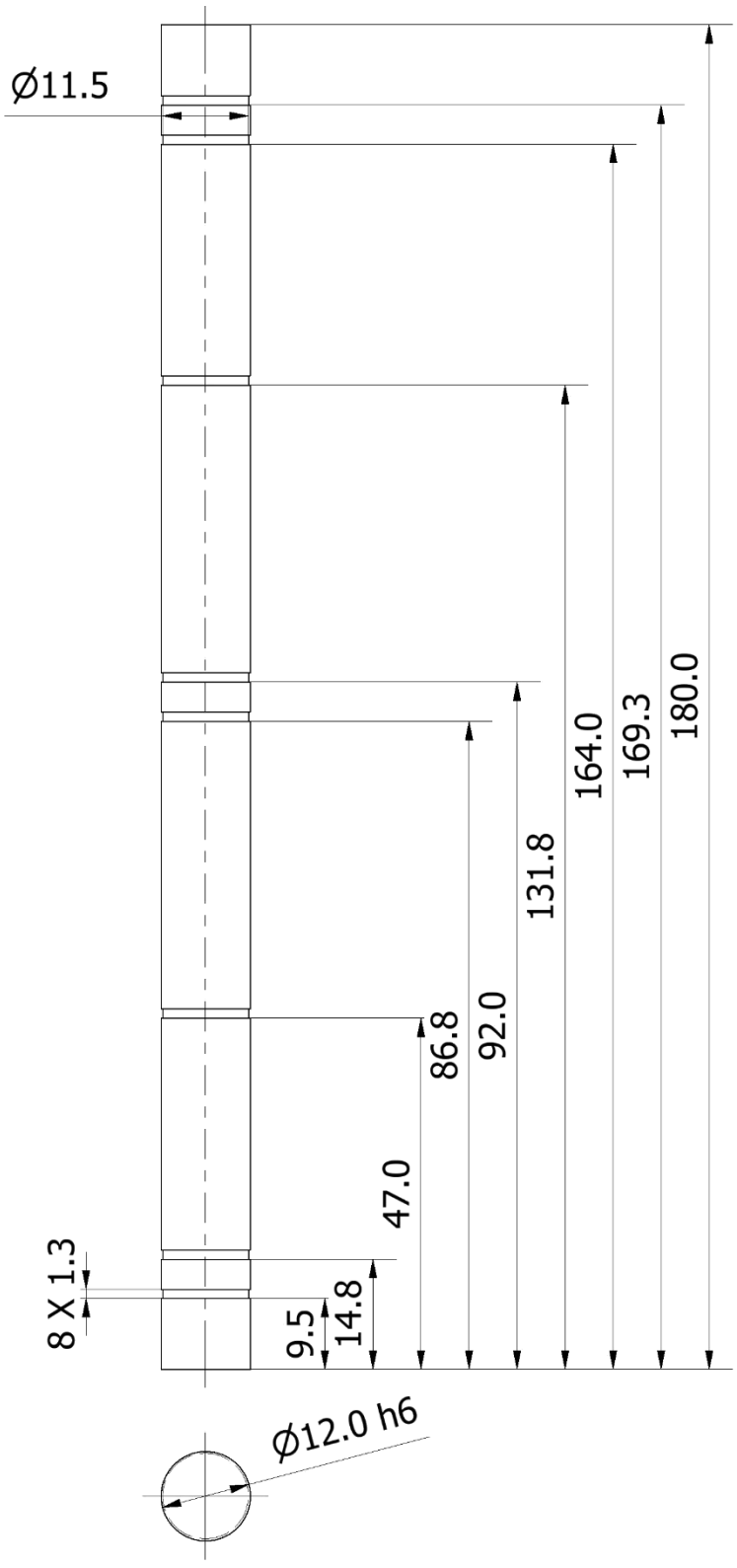
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squat Static Balancing To Hip Rod		
	E103	Edition	Sheet 24 / 95



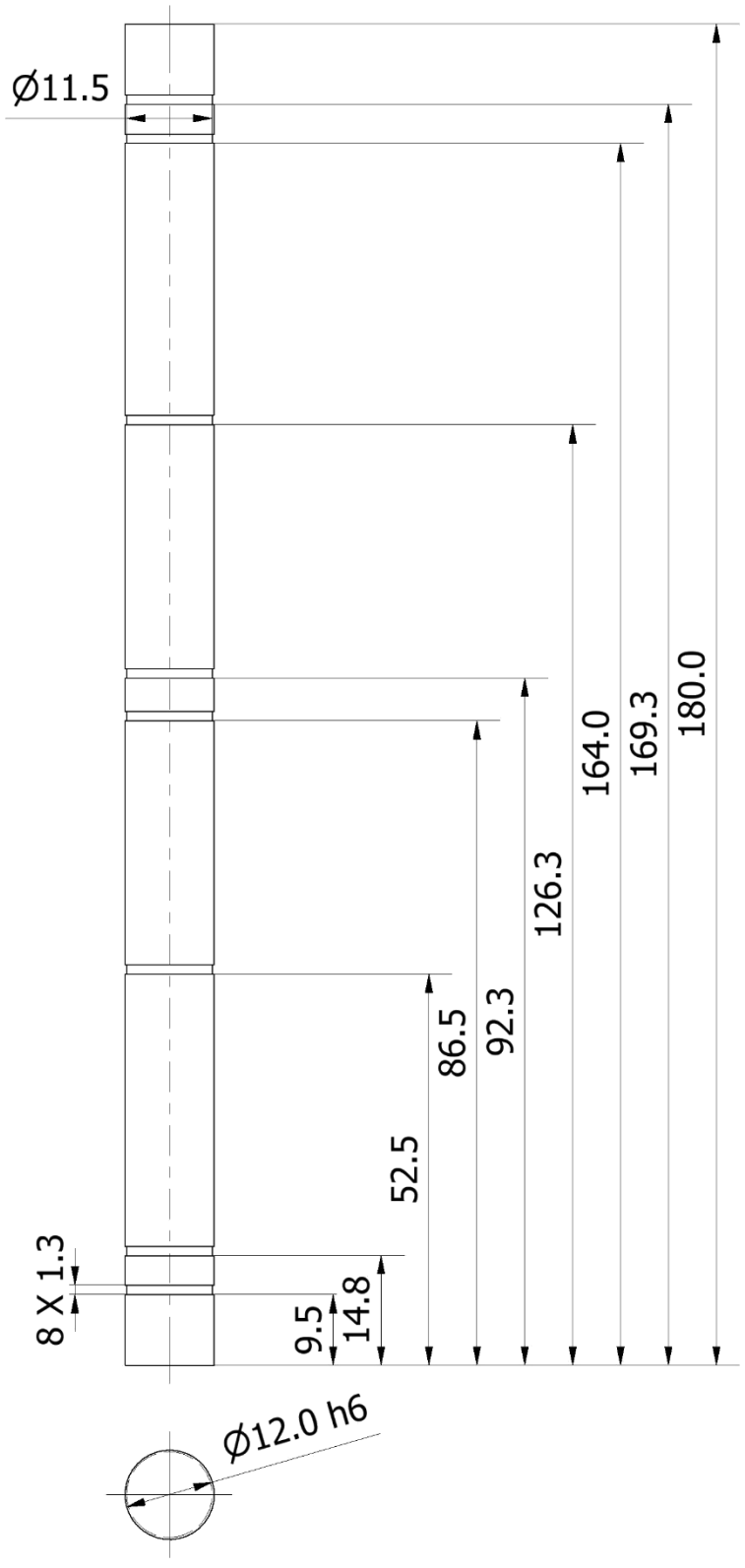
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hip To Torso Rod		
	E104	Edition	Sheet 25 / 95



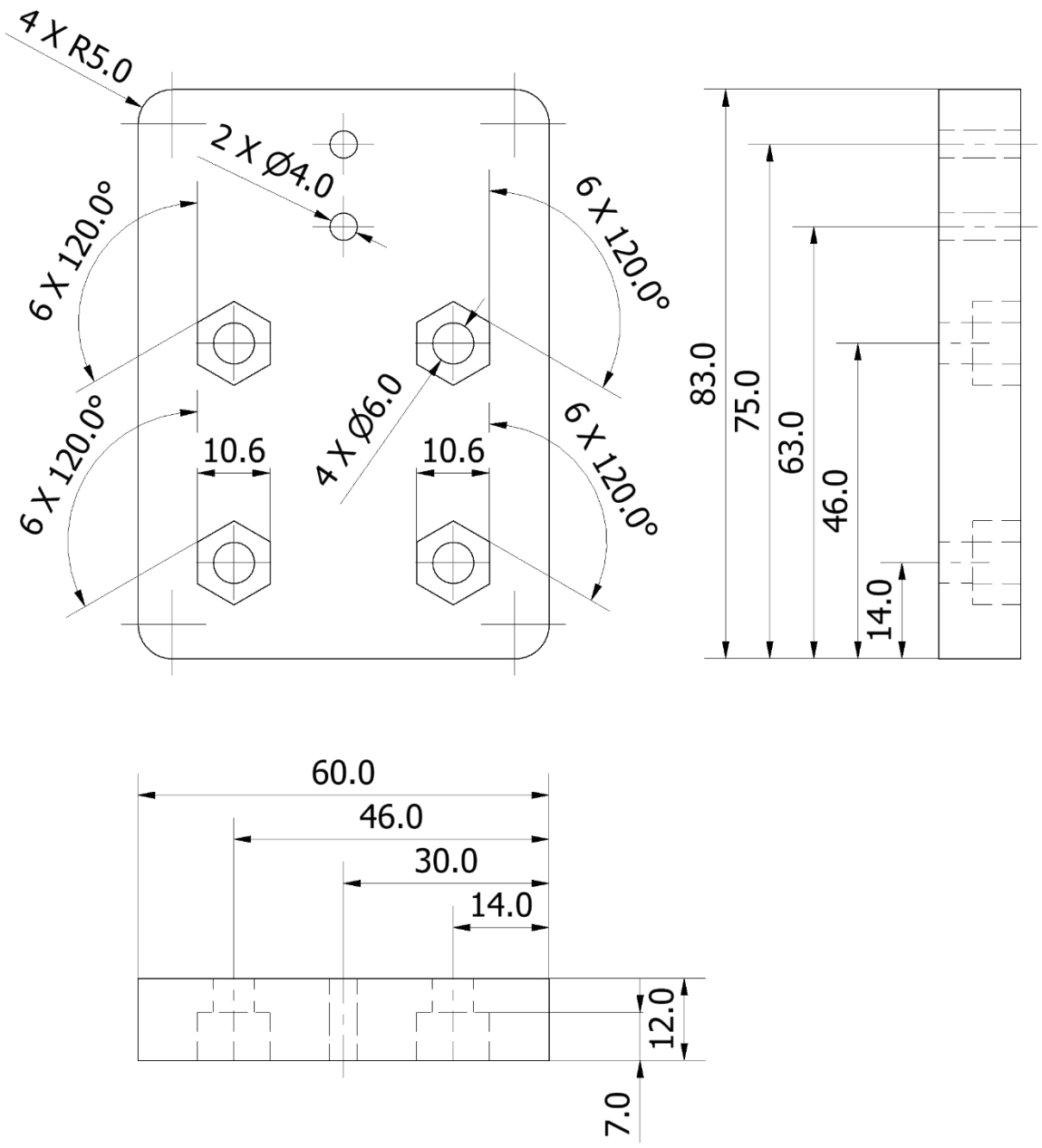
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hip To Spring Rod		
	E105	Edition	Sheet 26 / 95



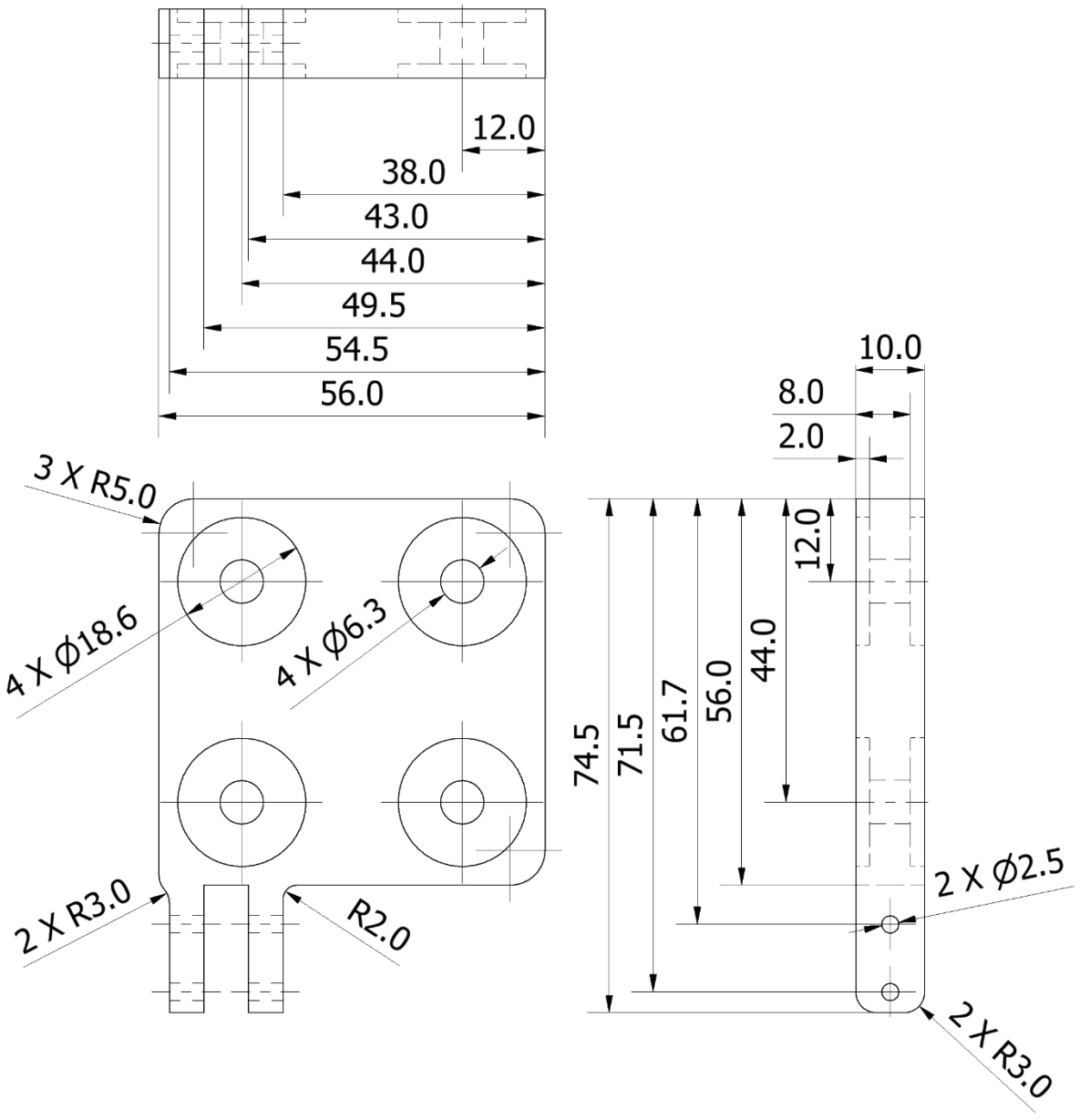
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Torso To Spring Rod		
	E106	Edition	Sheet 27 / 95



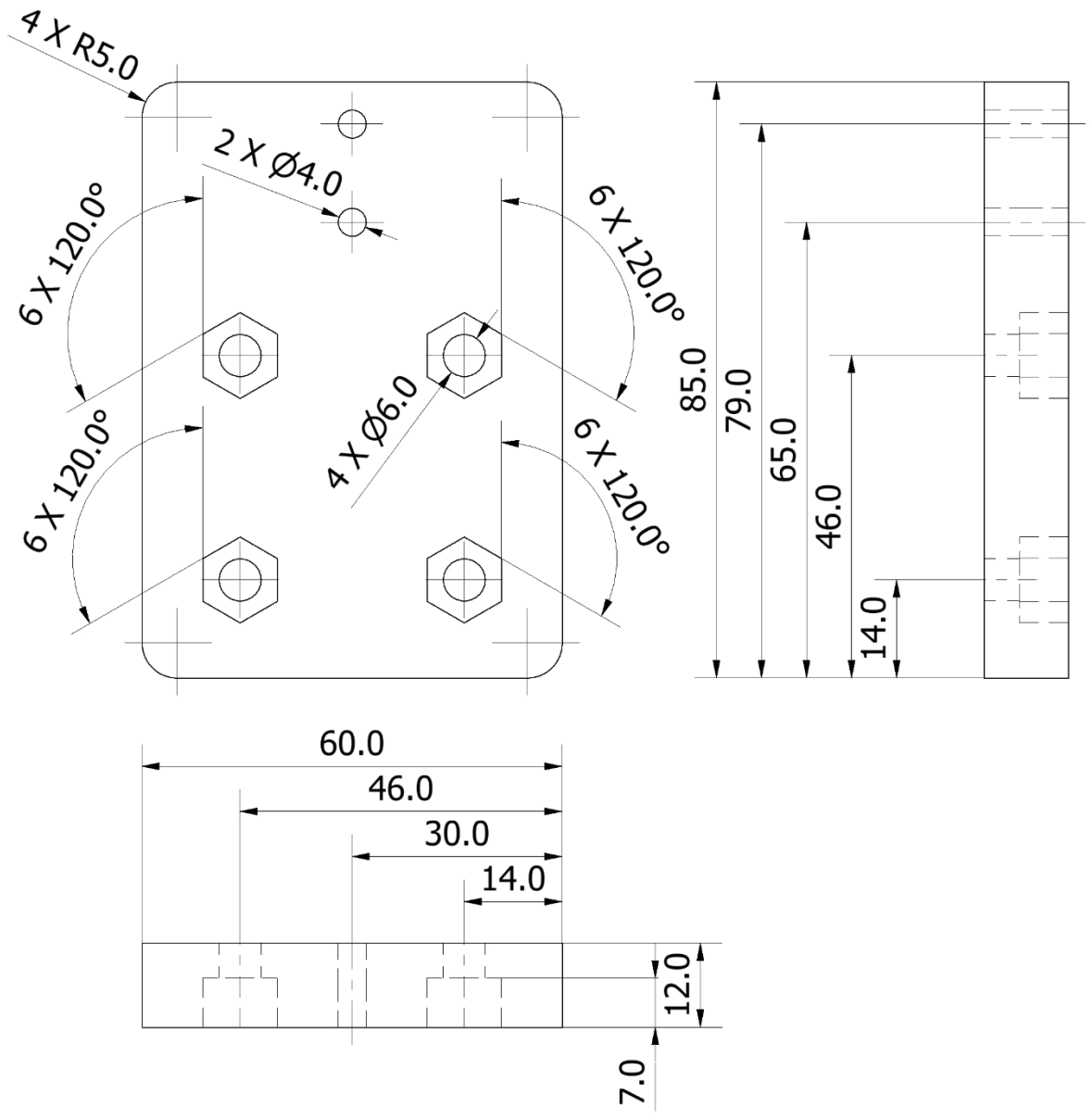
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squatting Lower Limit Switch Support		
	F000	Edition	Sheet 28 / 95



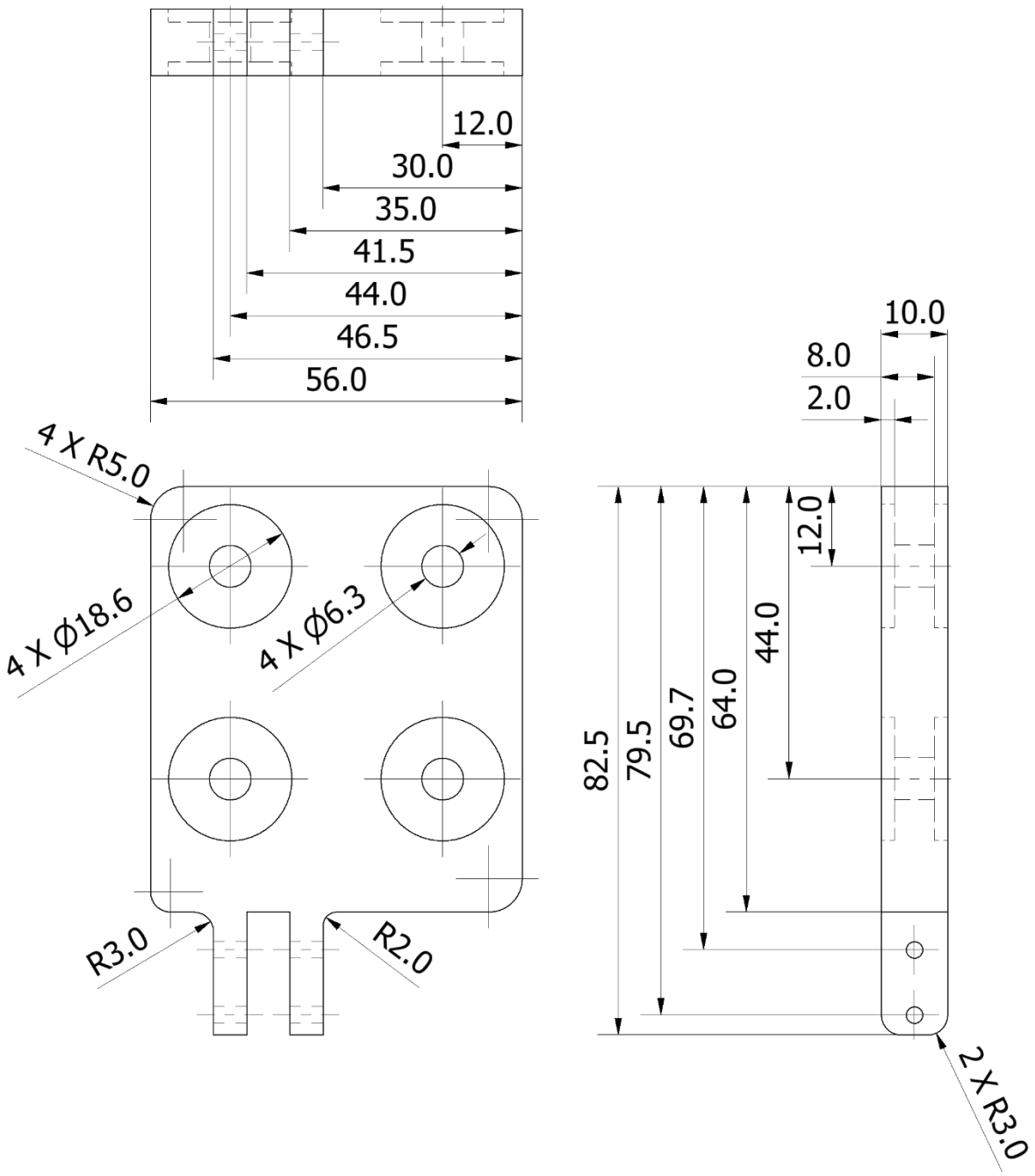
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squatting Lower Limit Switch Holder		
	F001	Edition	Sheet 29 / 95



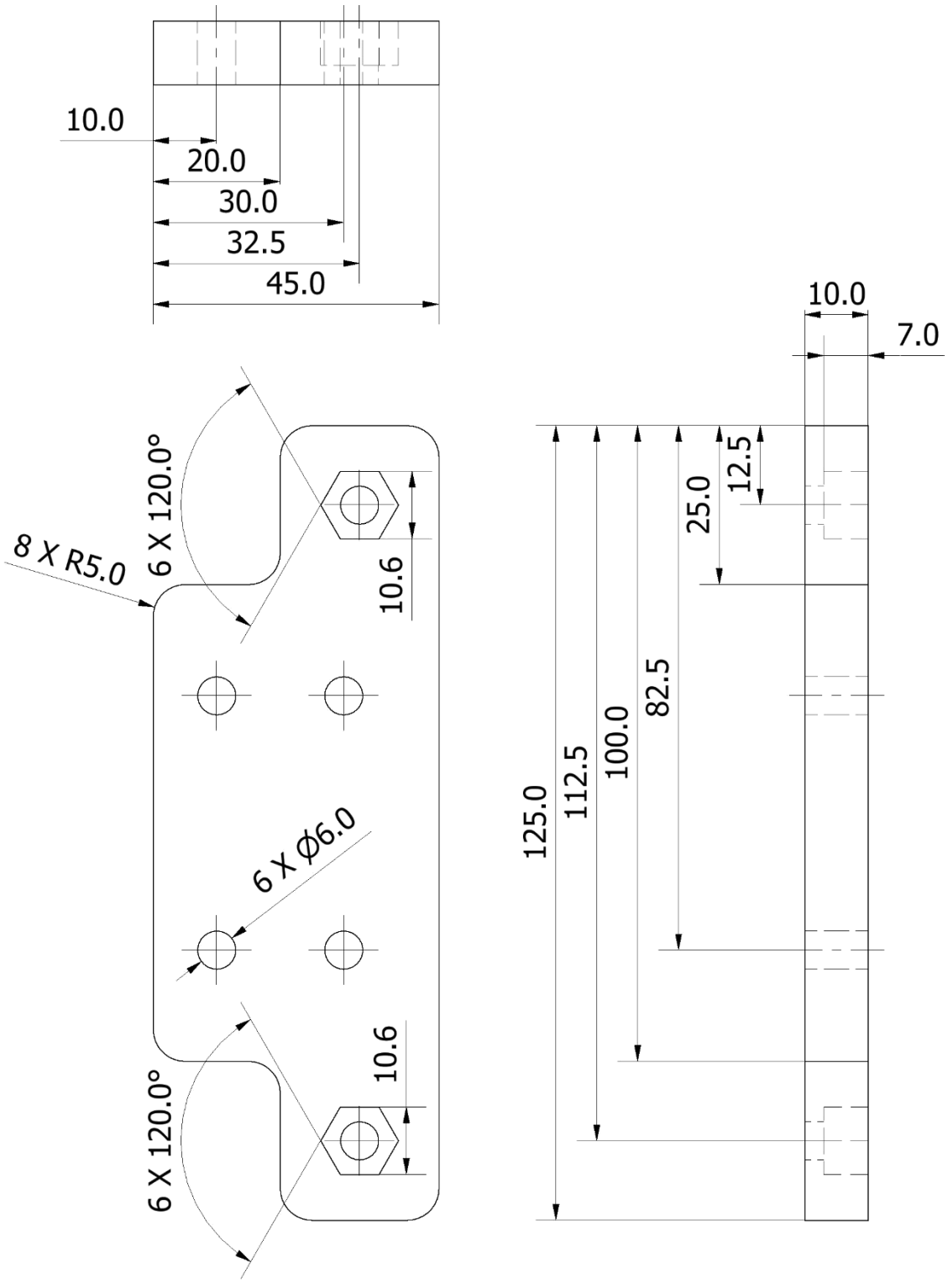
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squatting Upper Limit Switch Support		
	F002	Edition	Sheet 30 / 95



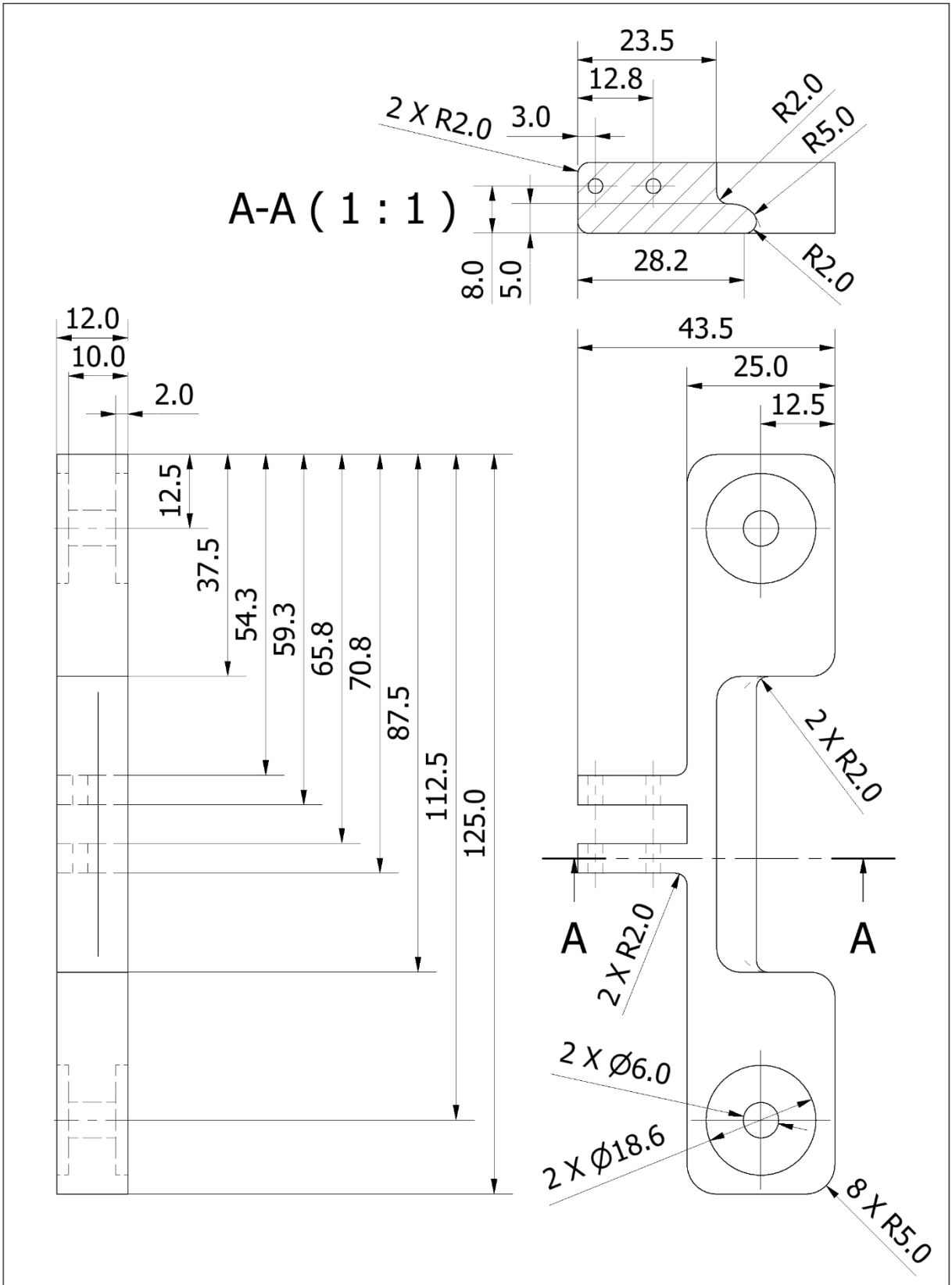
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Squatting Upper Limit Switch Holder		
	F003	Edition	Sheet 31 / 95



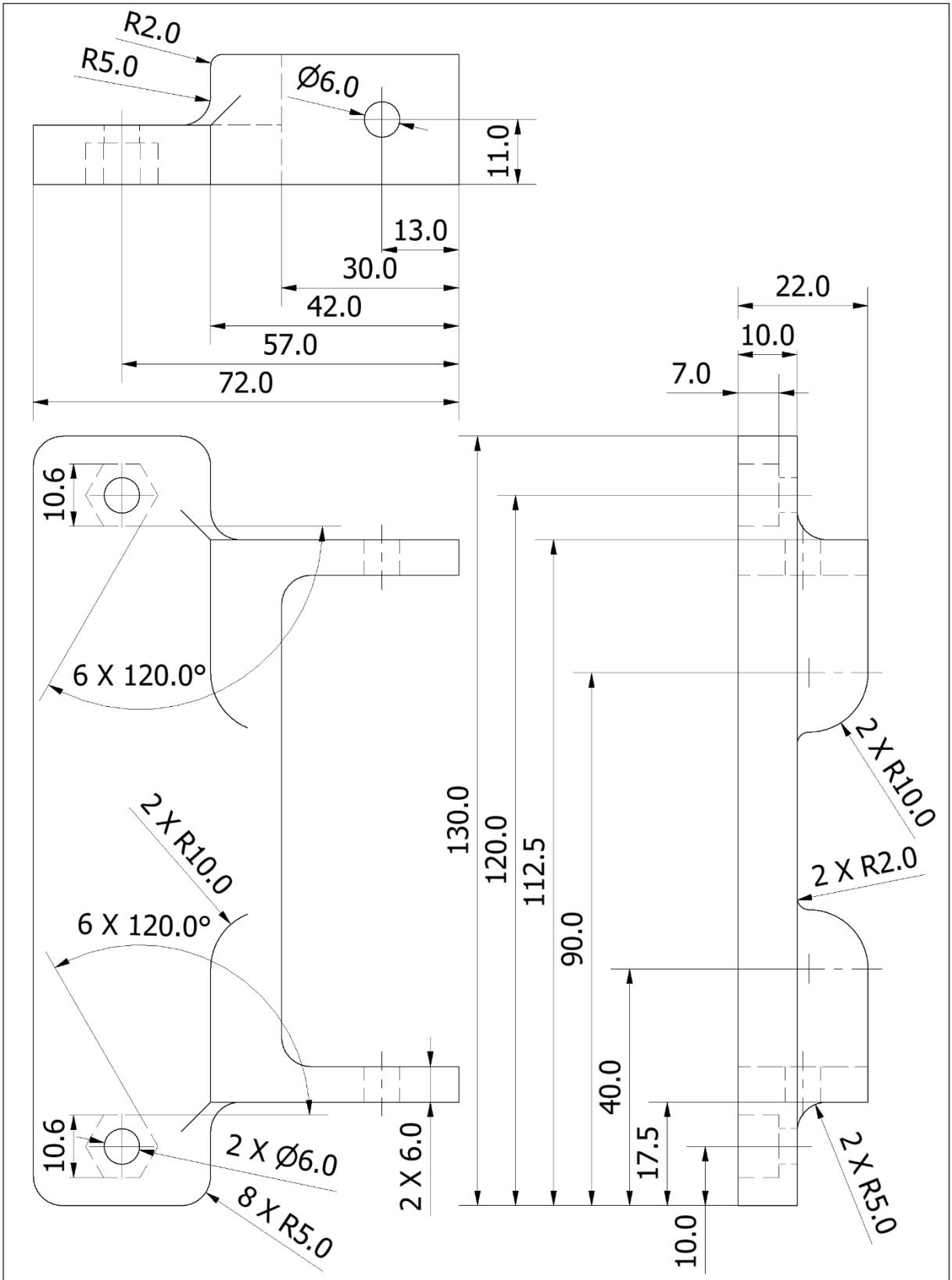
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Trunk Flexion Lower Limit Switch Support		
	F100	Edition	Sheet 32 / 95



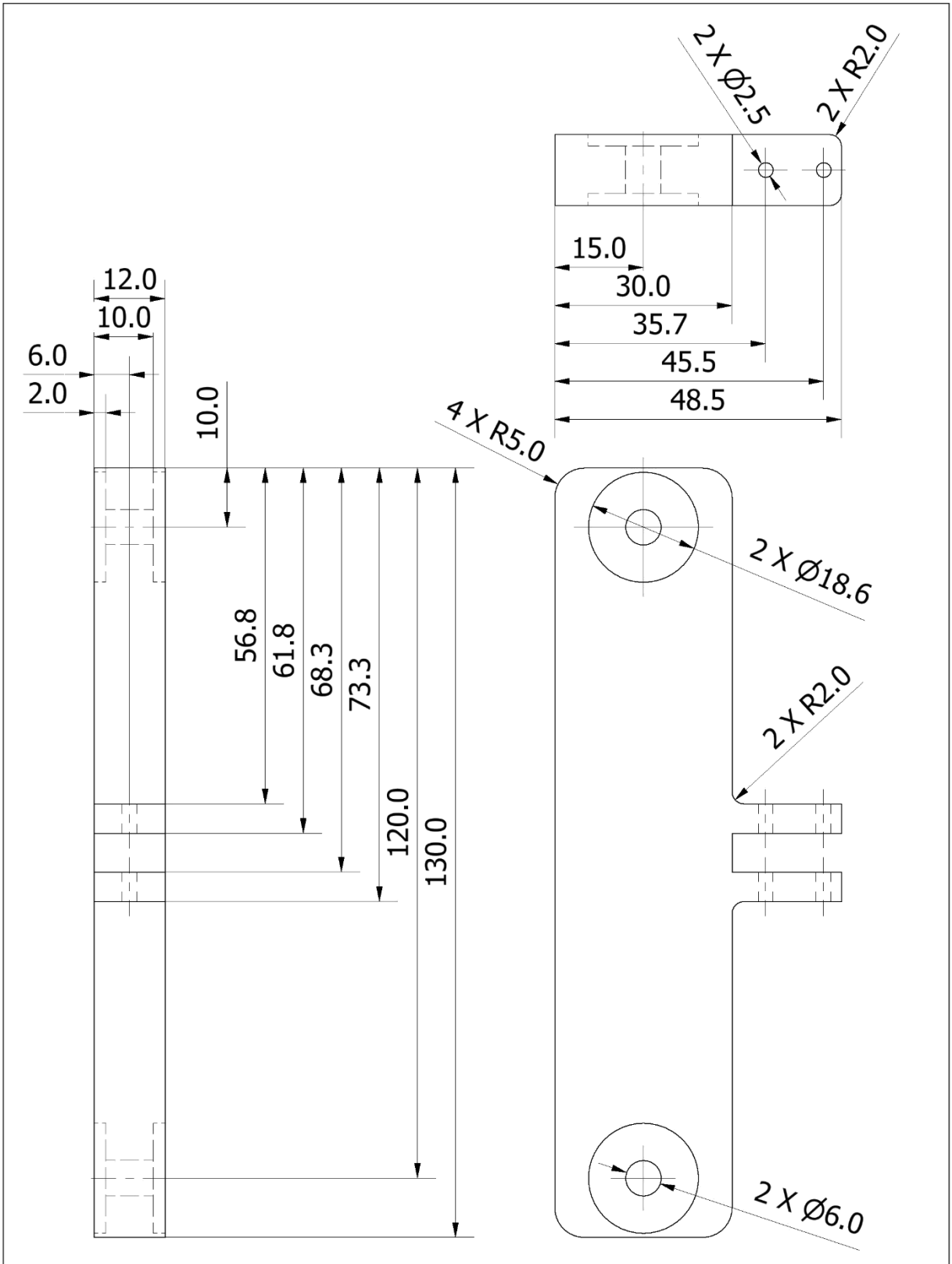
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Trunk Flexion Lower Limit Switch Holder		
	F101	Edition	Sheet 33 / 95



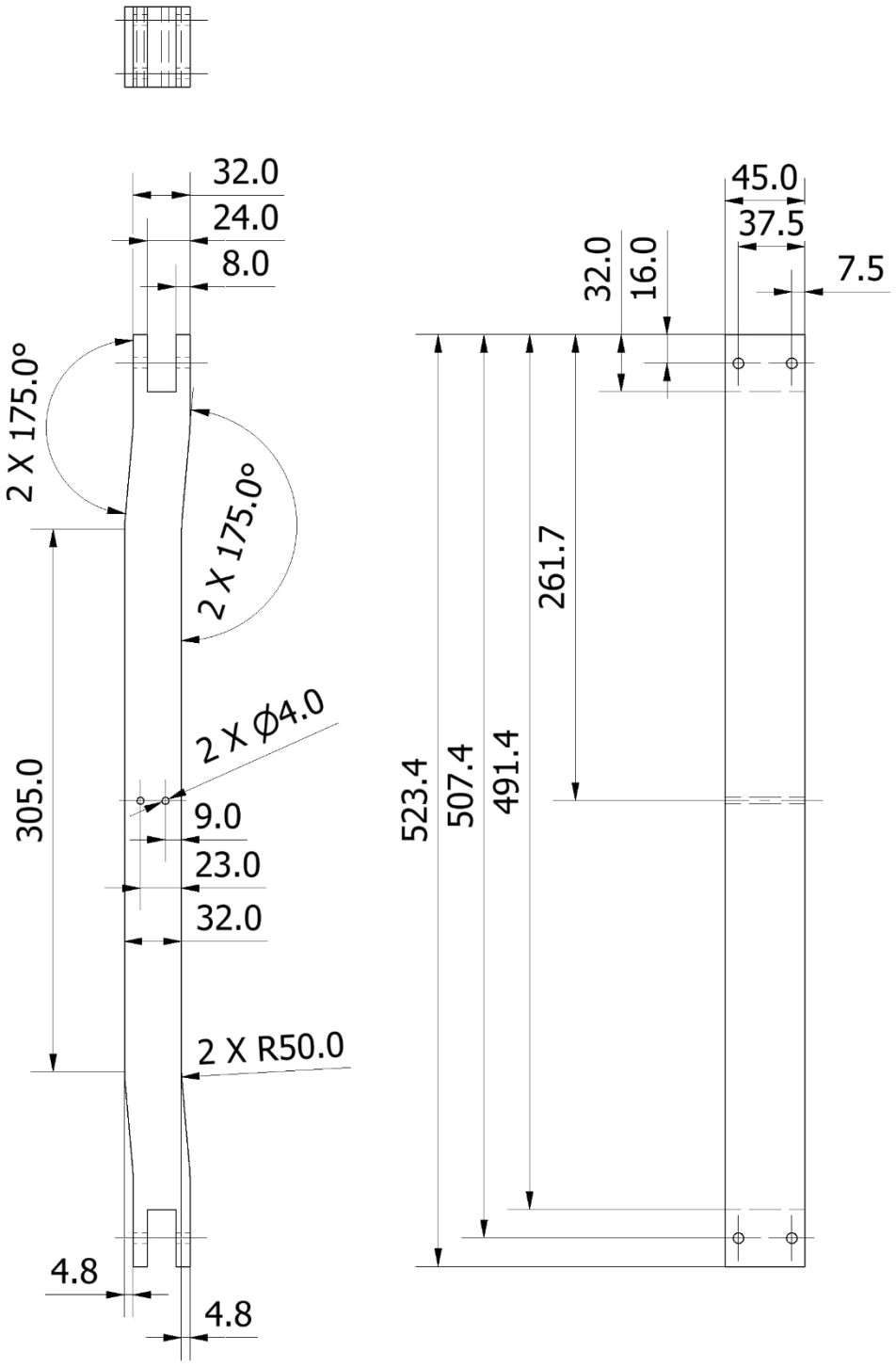
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Trunk Flexion Upper Limit Switch Support		
	F102	Edition	Sheet 34 / 95



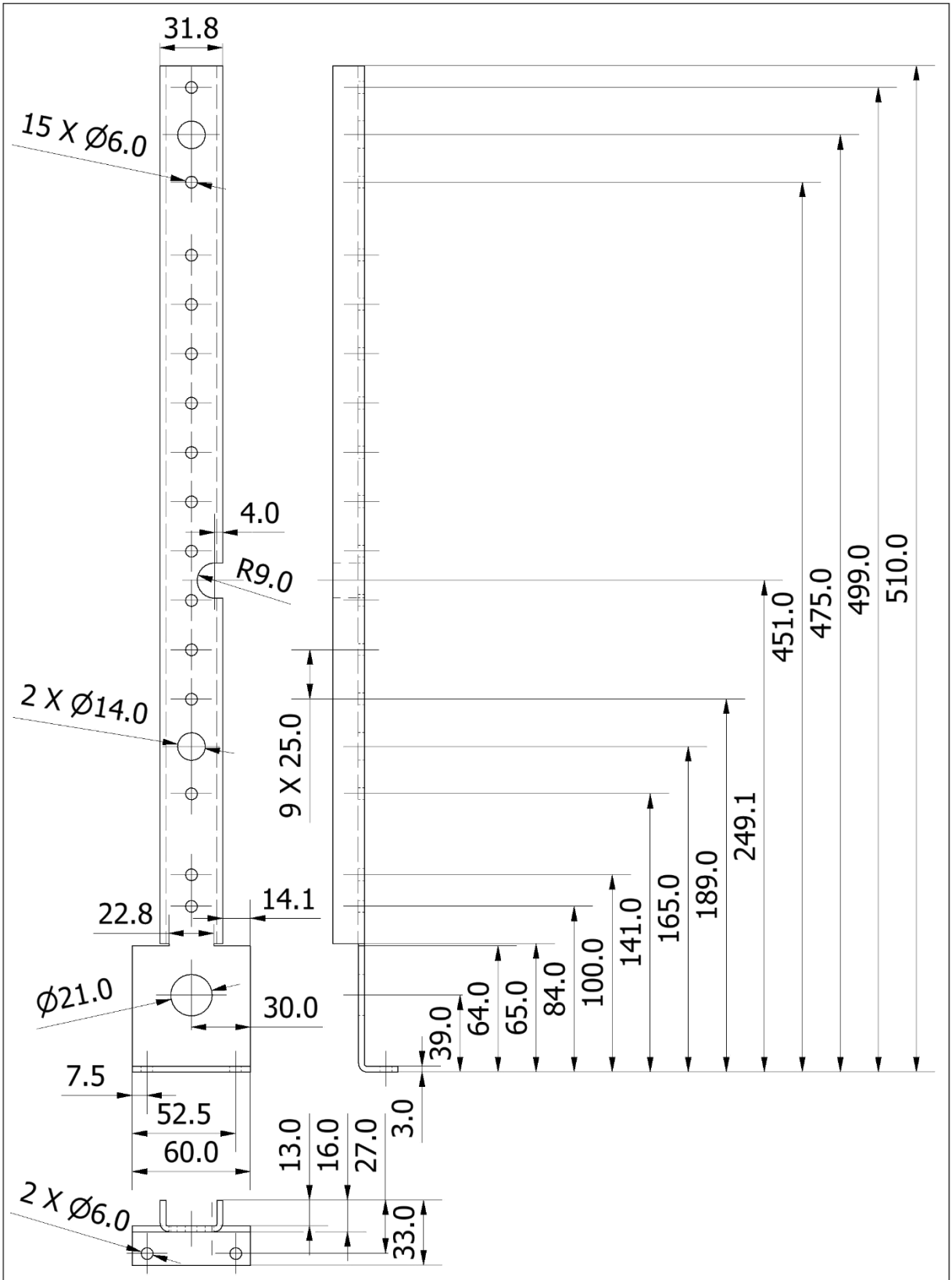
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Trunk Flexion Upper Limit Switch Holder		
	F103	Edition	Sheet 35 / 95



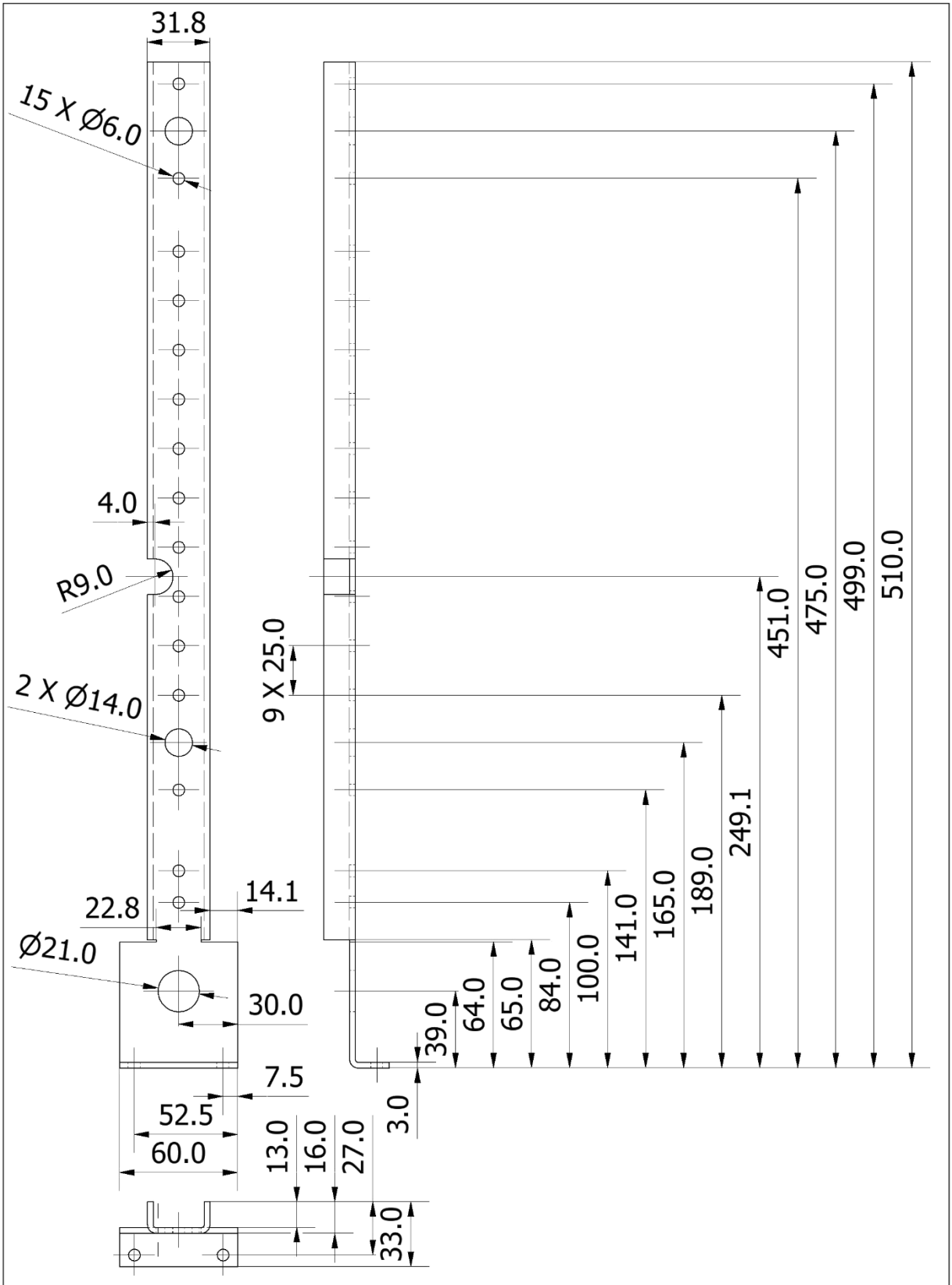
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 4

Universidade do Minho Laboratório de Automação e Robótica	Lateral Pillar Brace Support		
	G000	Edition	Sheet 36 / 95



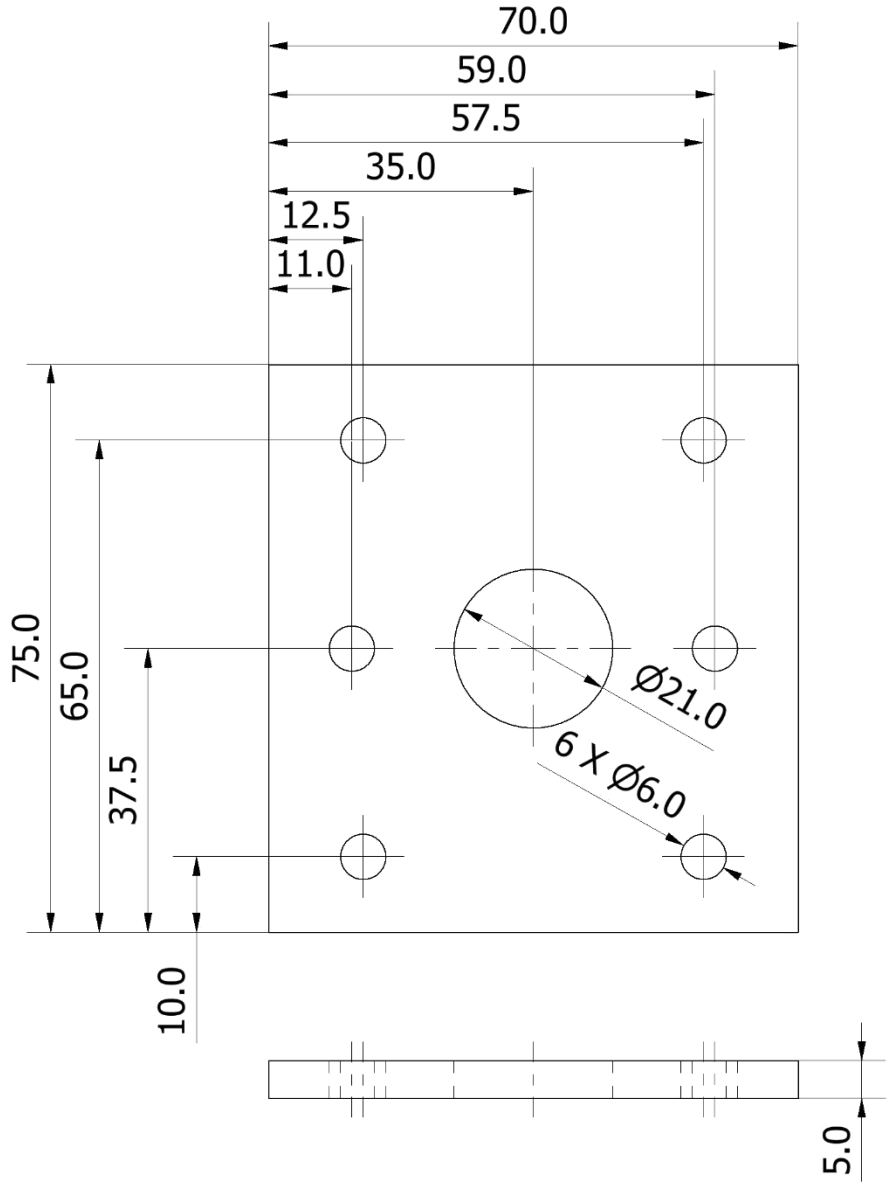
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 3

Universidade do Minho Laboratório de Automação e Robótica	Hip Left Pillar		
	H000	Edition	Sheet 37 / 95



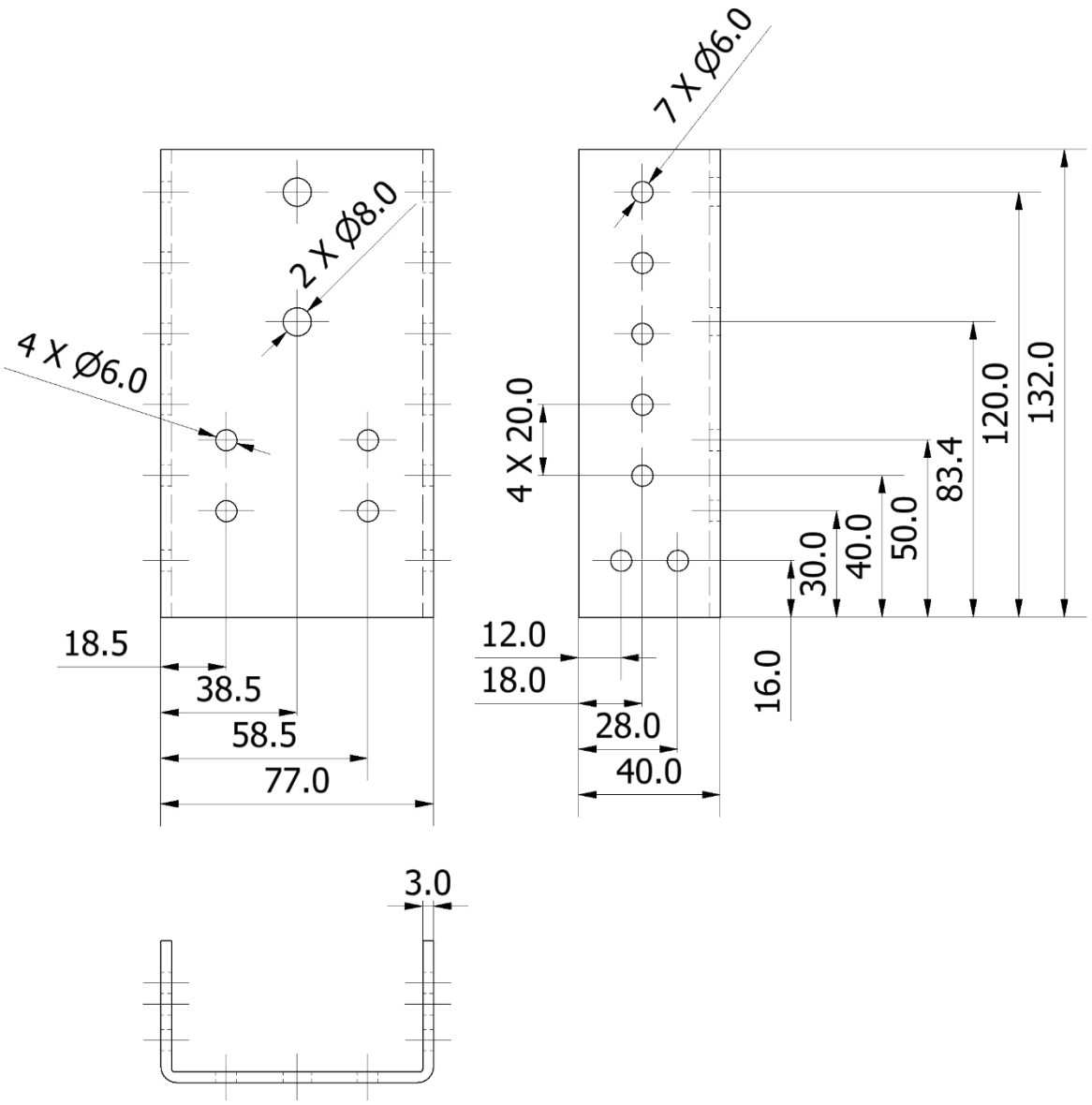
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 3

Universidade do Minho Laboratório de Automação e Robótica	Hip Right Pillar		
	H001	Edition	Sheet 38 / 95



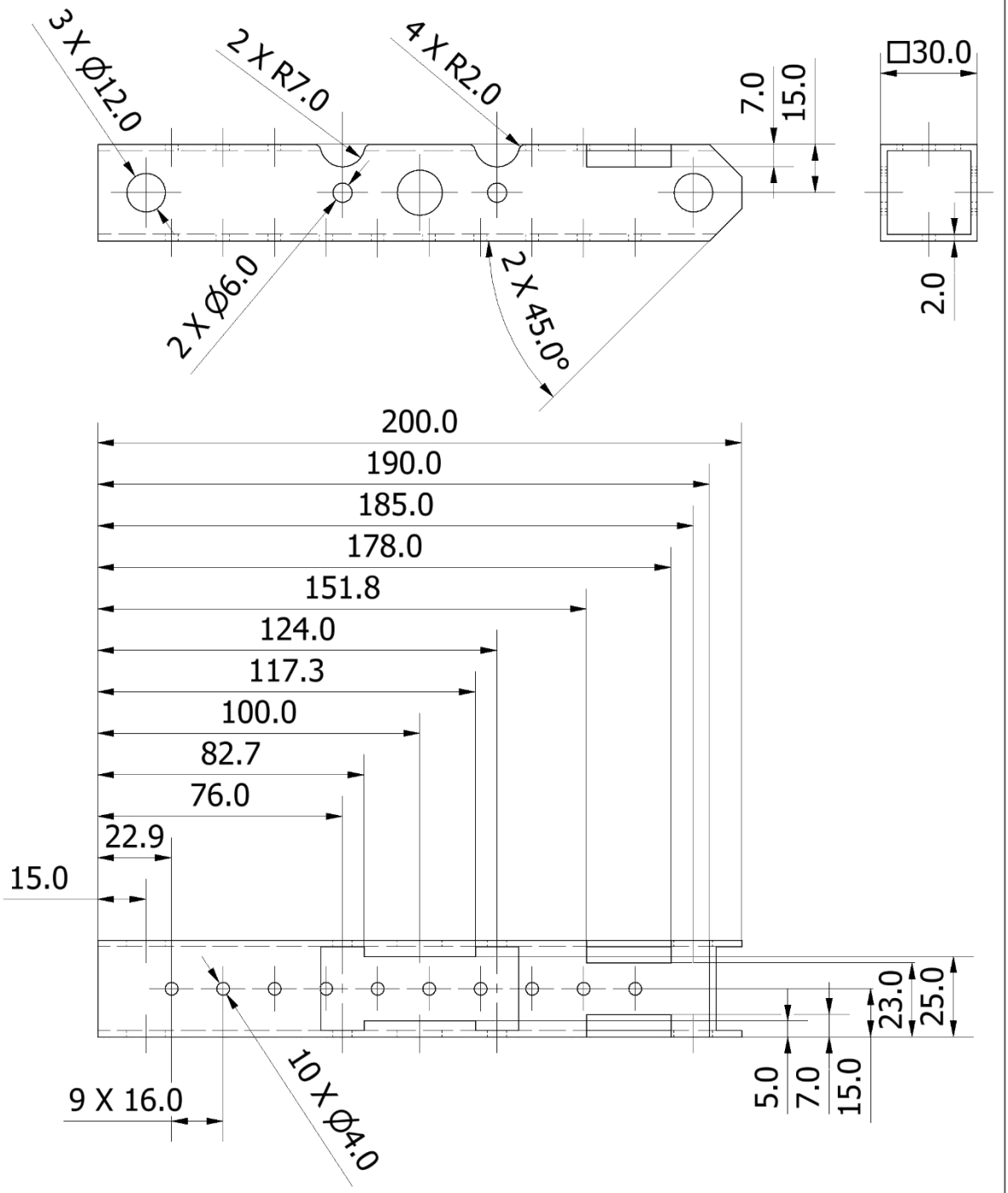
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hip Central Plate		
	H002	Edition	Sheet 39 / 95



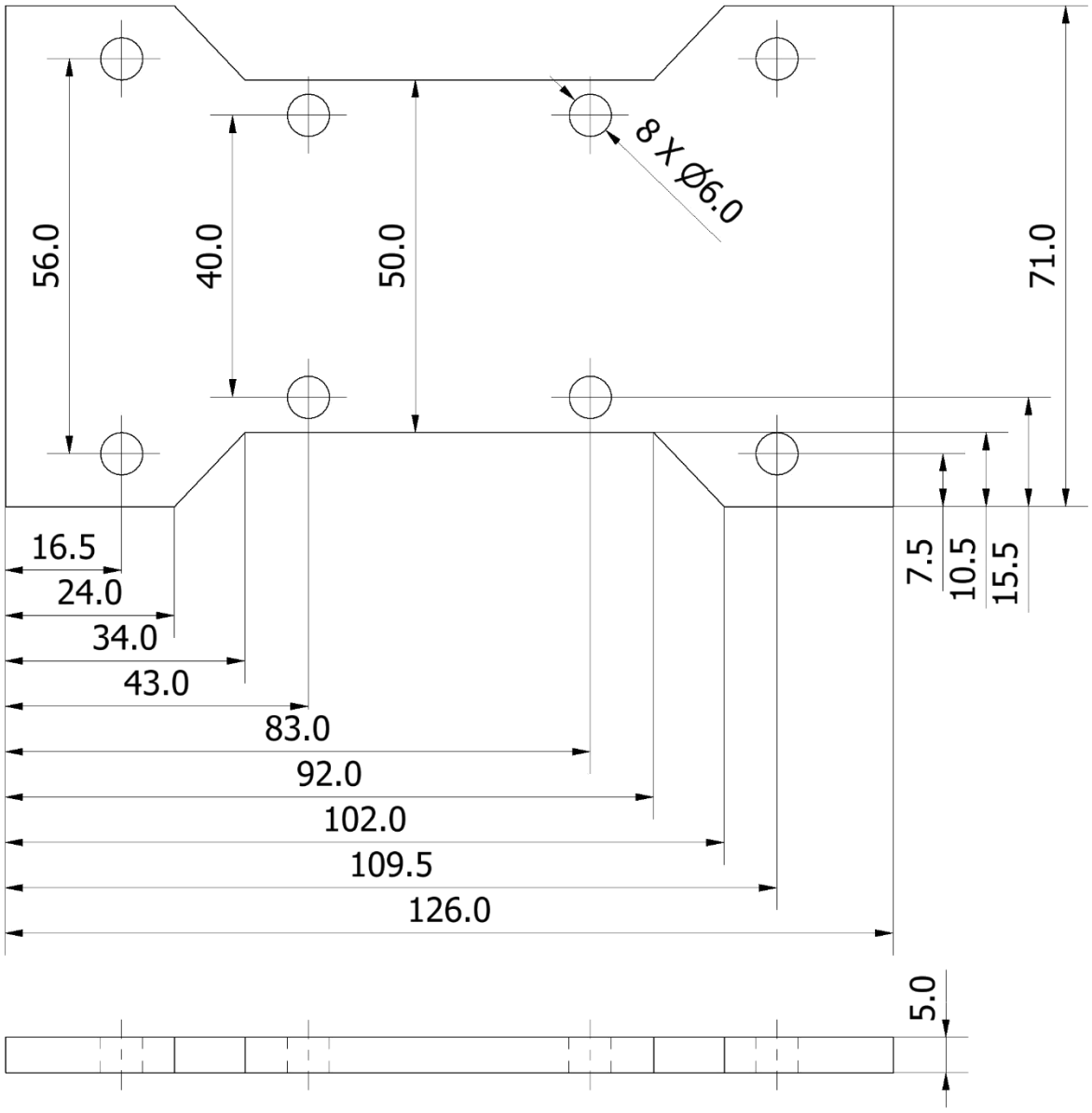
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Hip Linear Actuator Support		
	H003	Edition	Sheet 40 / 95



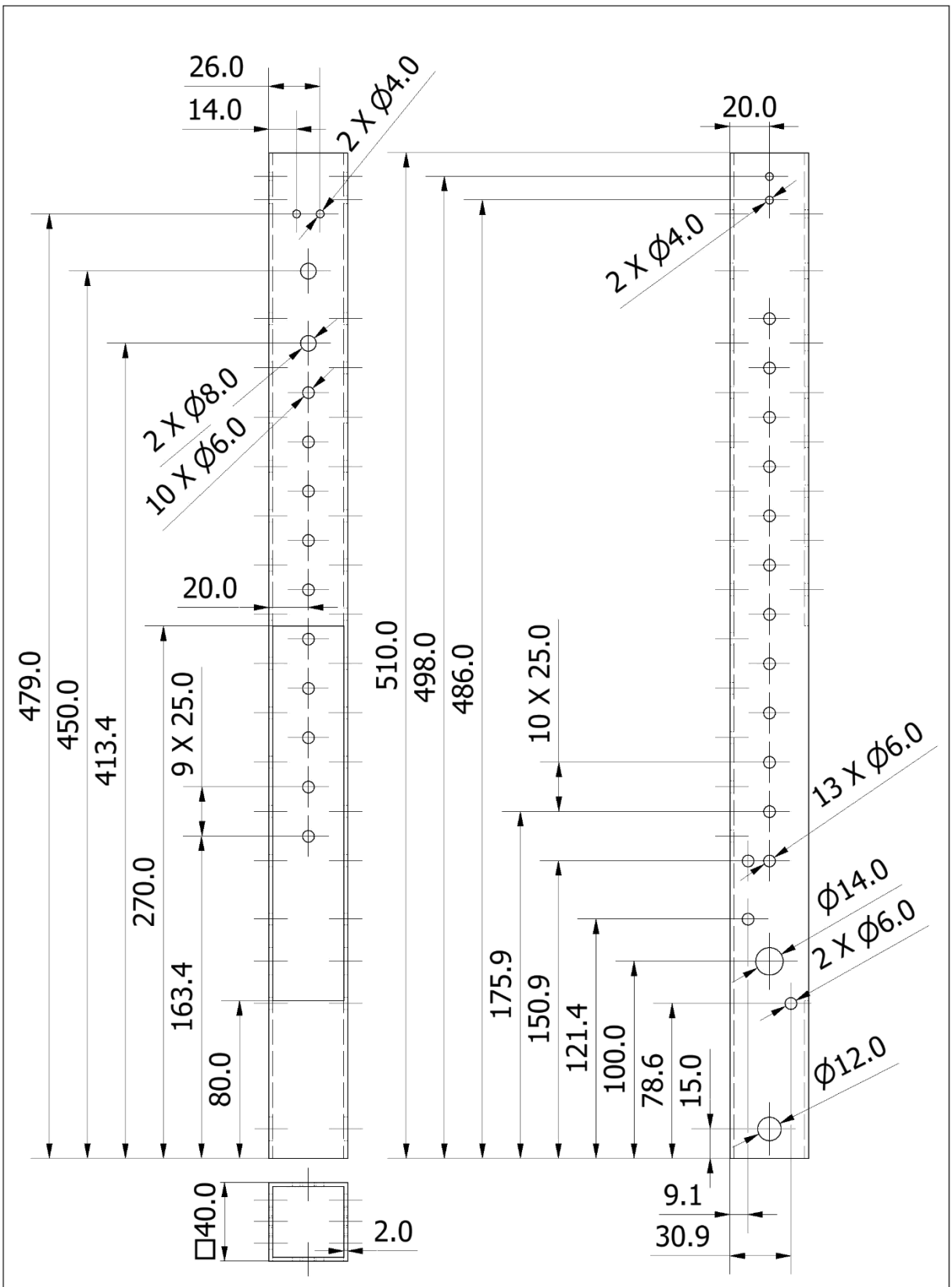
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Squatting Static Balancing Tube		
	H004	Edition	Sheet 41 / 95



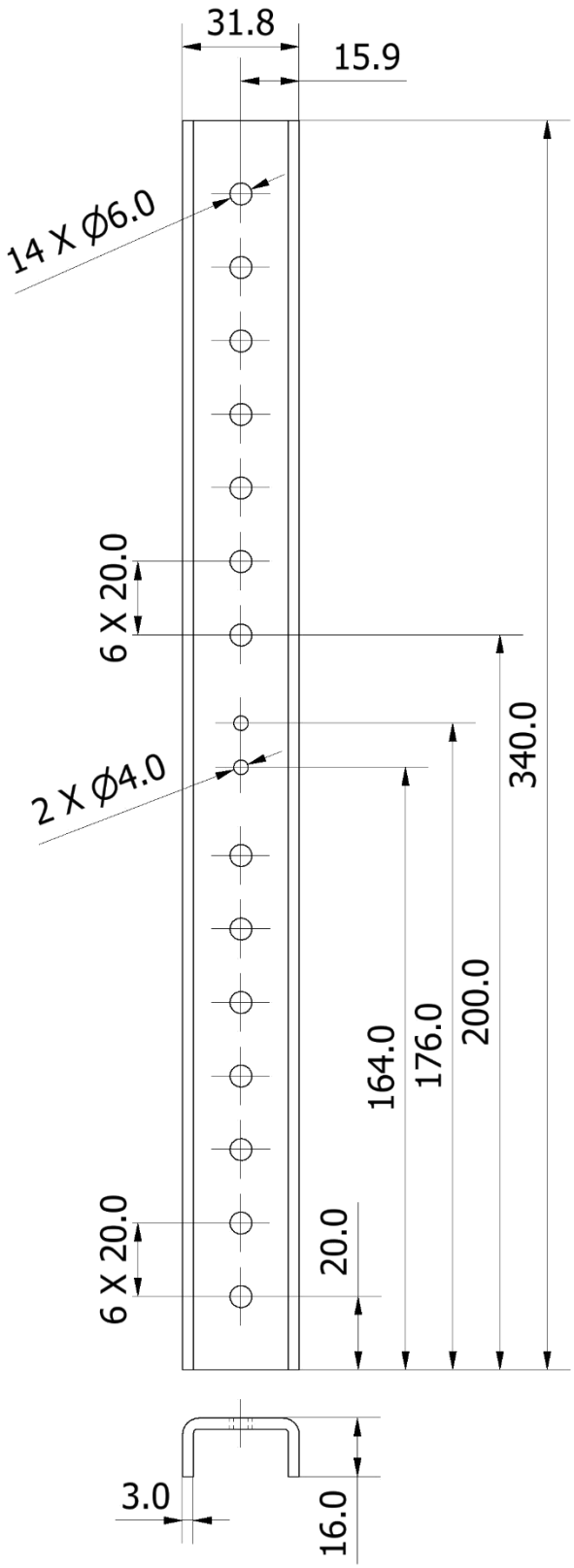
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hip Lateral Linear Bearing Support Plate		
	H005	Edition	Sheet 42 / 95



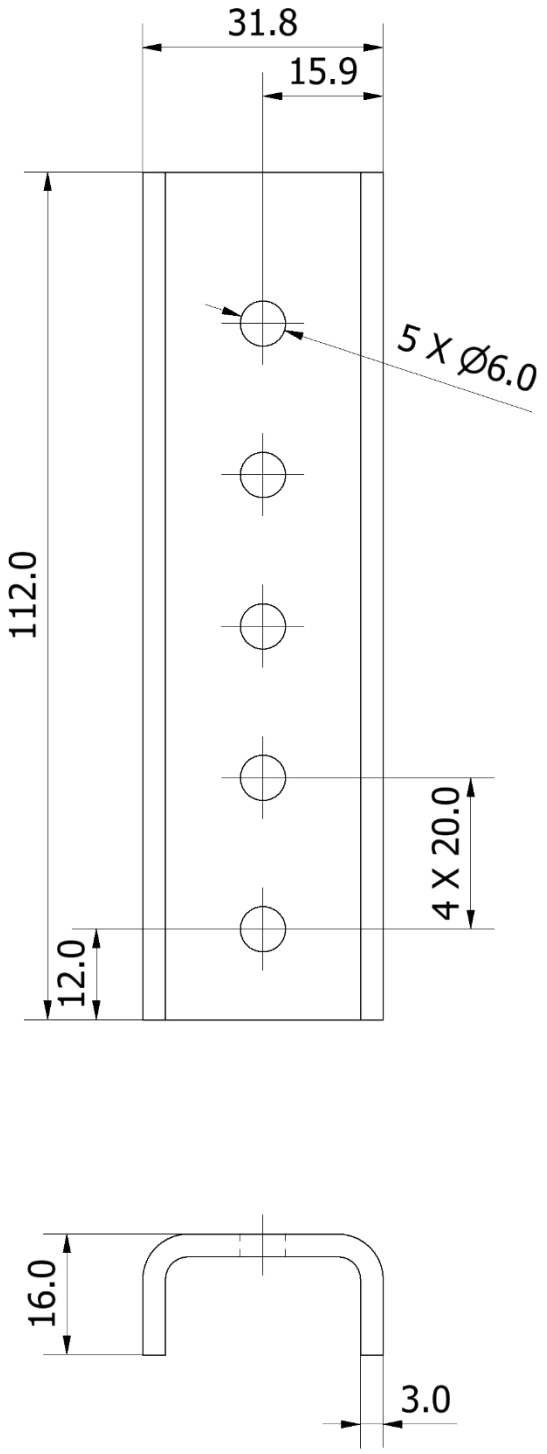
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 3

Universidade do Minho Laboratório de Automação e Robótica	Torso Steel Tube		
	I000	Edition	Sheet 43 / 95



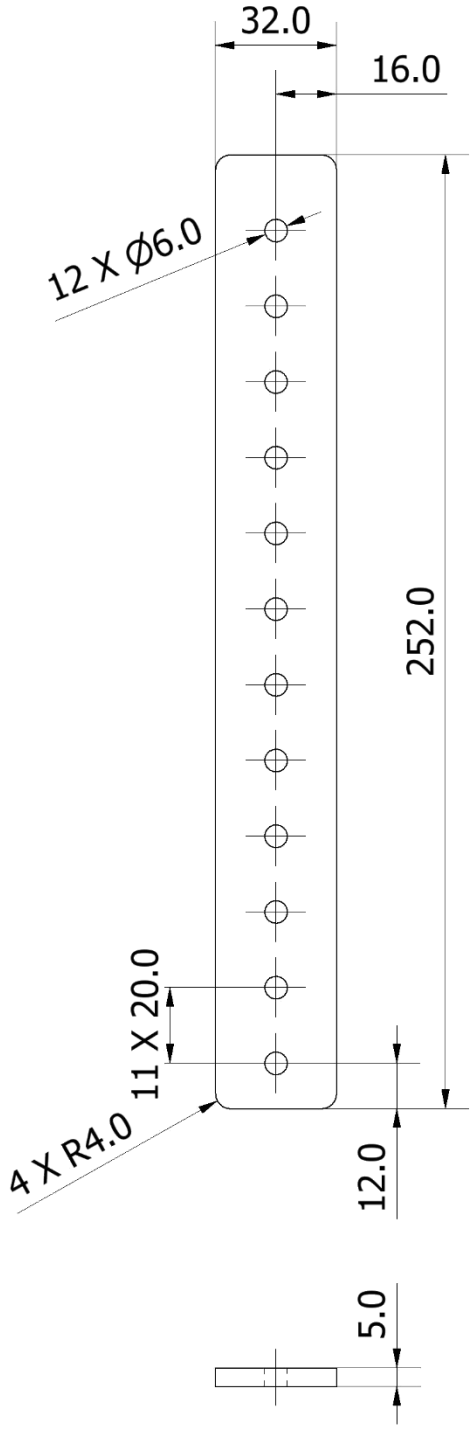
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Clavicle U-Profile		
	I001	Edition	Sheet 44 / 95



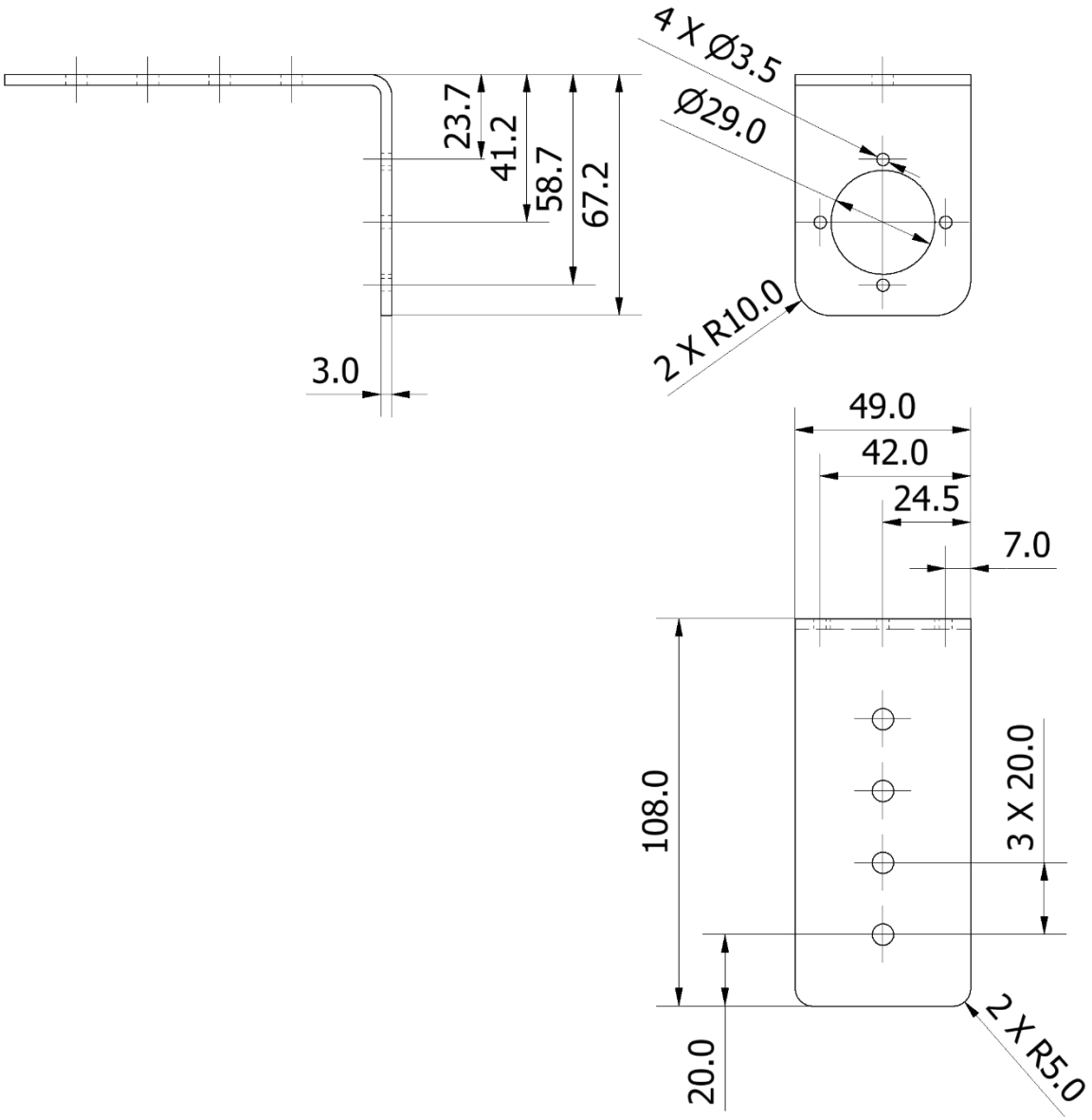
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Scapula U-Profile		
	J000	Edition	Sheet 45 / 95



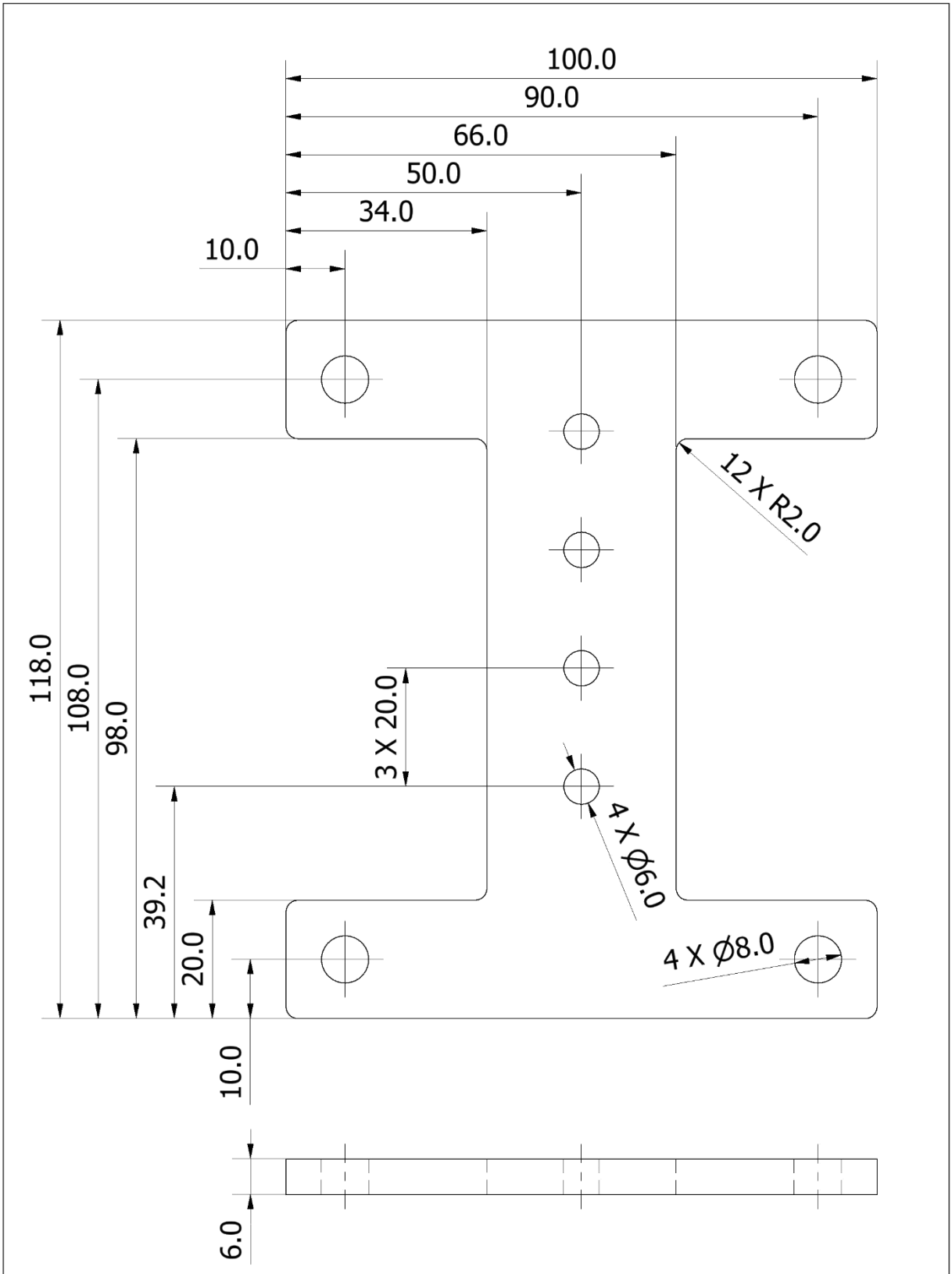
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Scapula Metal Plate		
	J001	Edition	Sheet 46 / 95



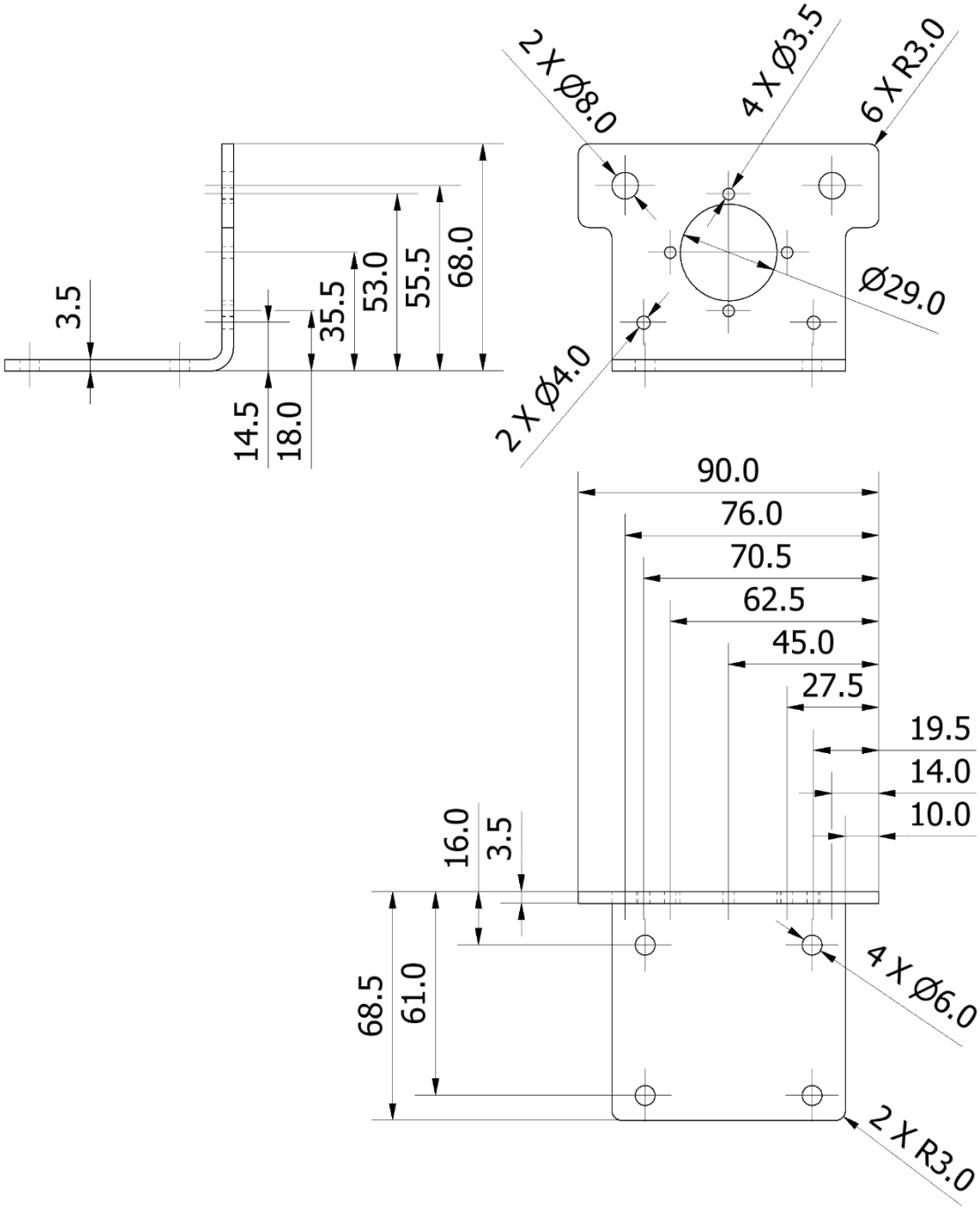
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Scapula Motor Support		
	J002	Edition	Sheet 47 / 95



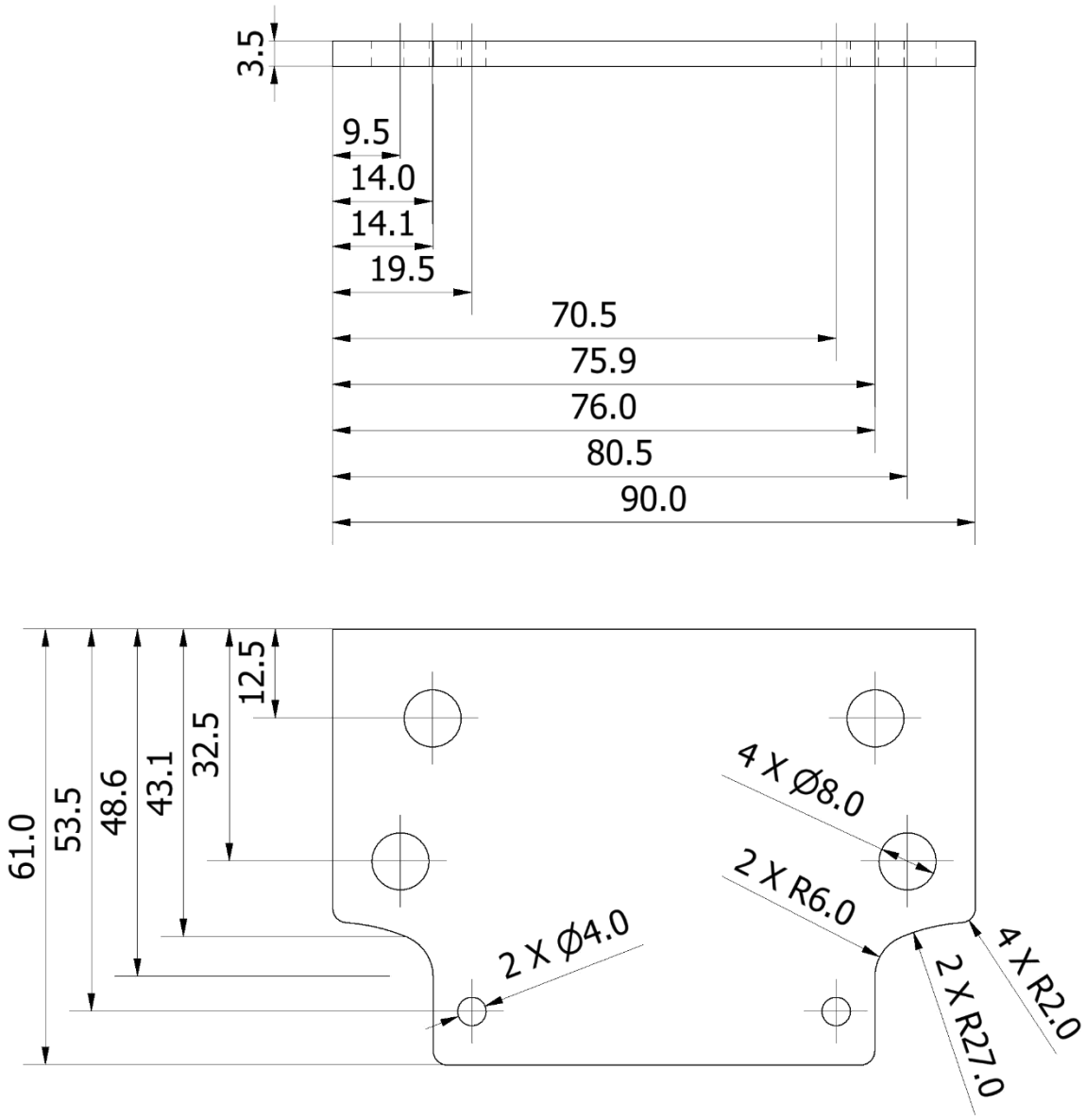
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Scapula Bearing Support		
	J003	Edition	Sheet 48 / 95



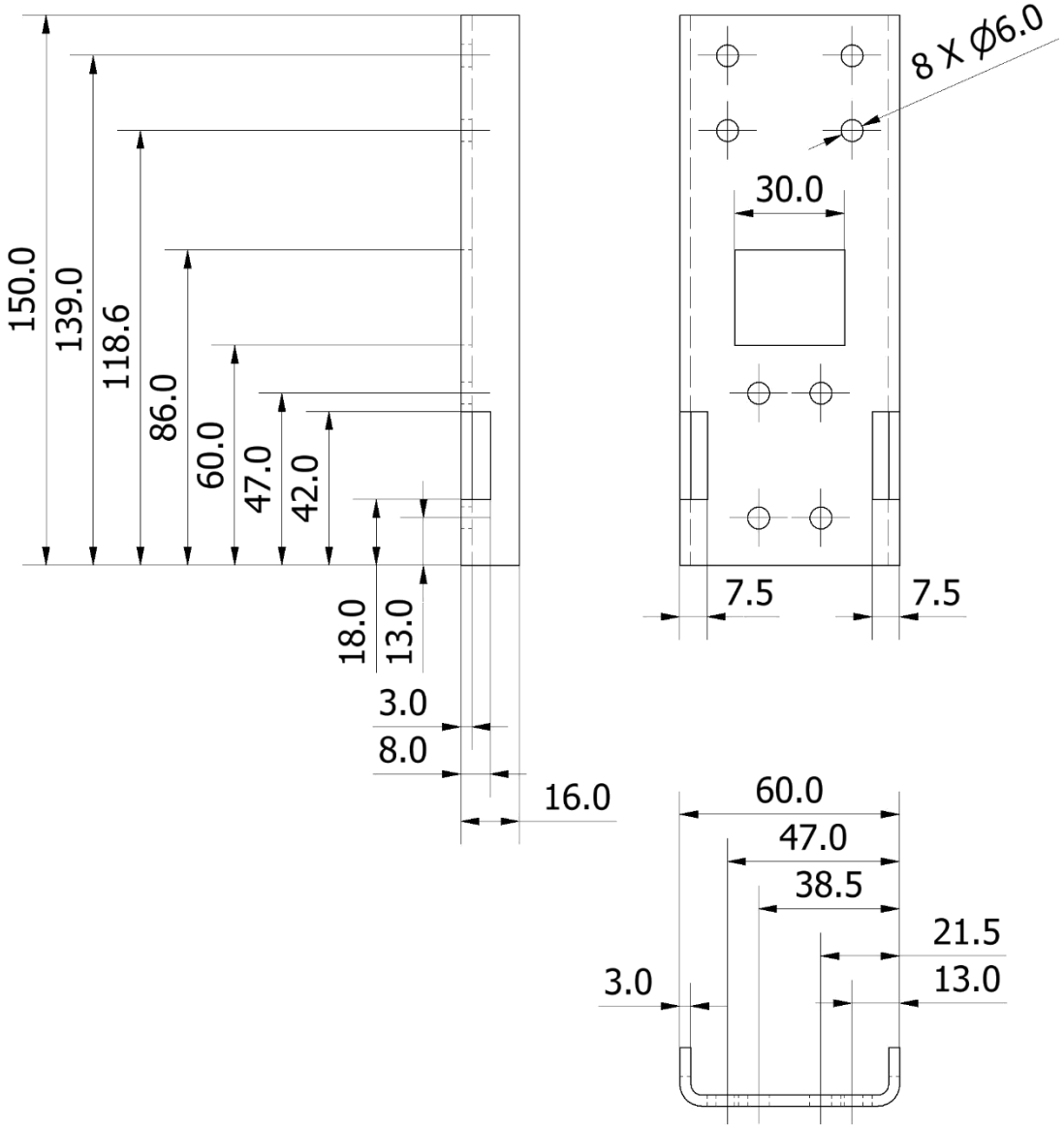
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Shoulder Motor Support		
	J010	Edition	Sheet 49 / 95



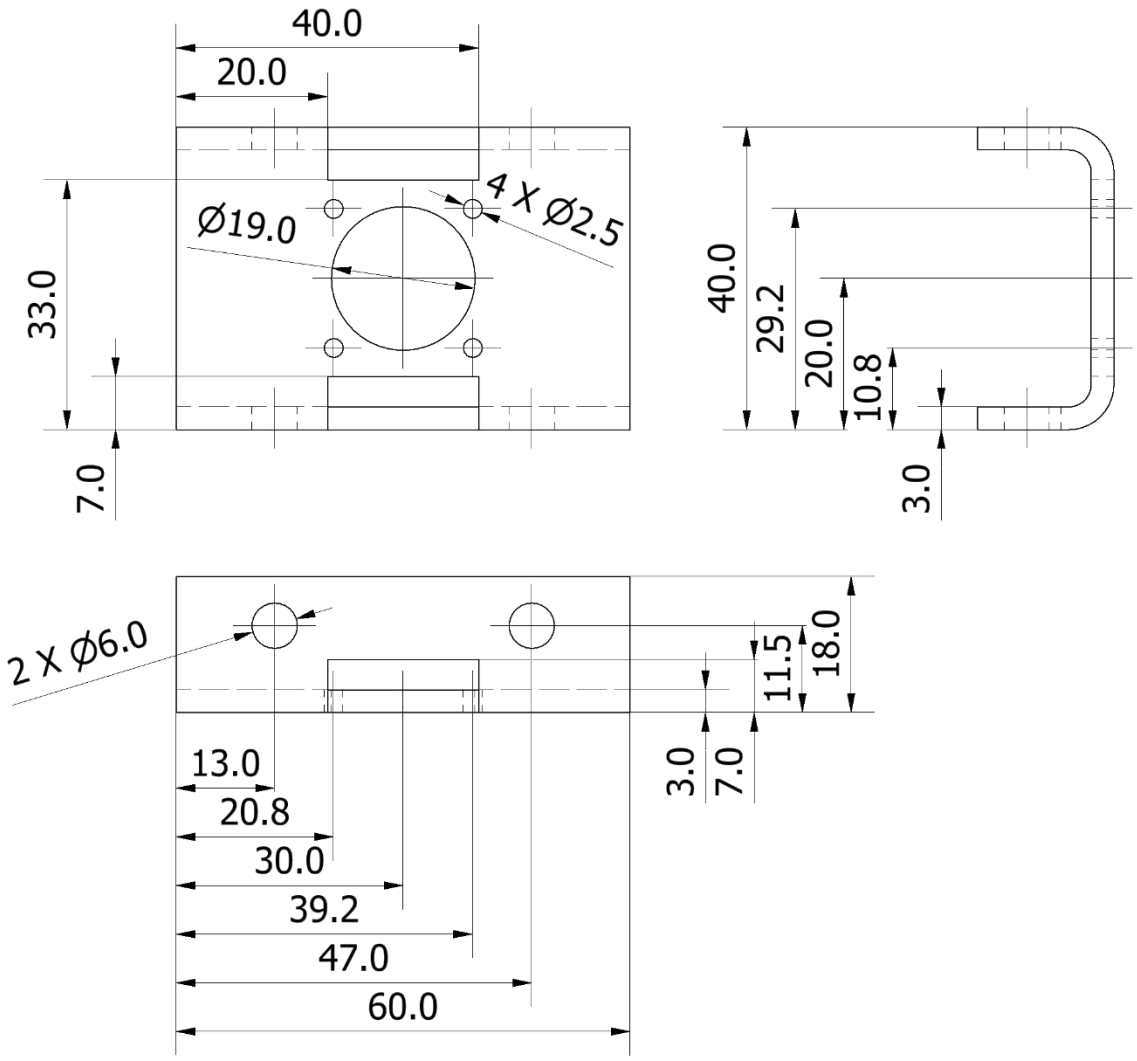
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Shoulder Bearing Support		
	J011	Edition	Sheet 50 / 95



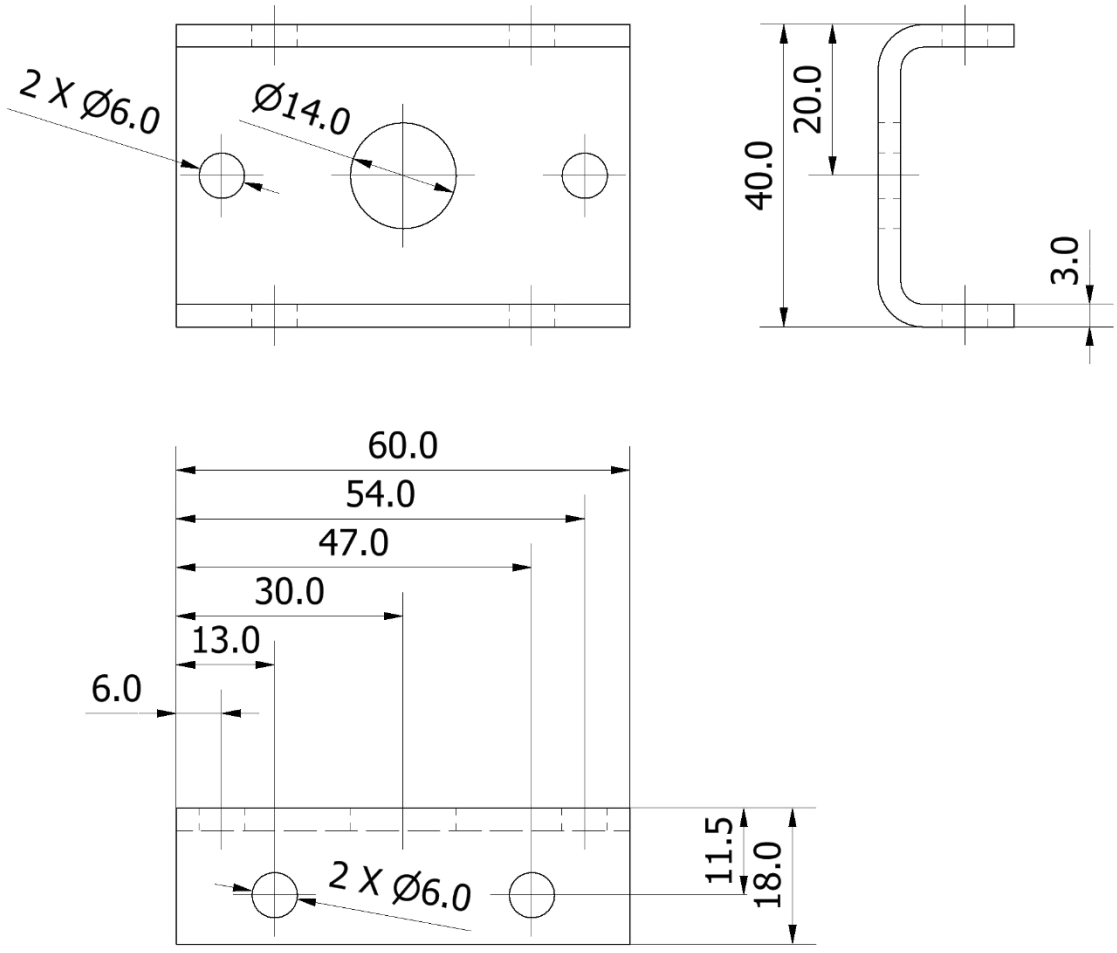
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Proximal Arm Outer Casing		
	J020	Edition	Sheet 51 / 95



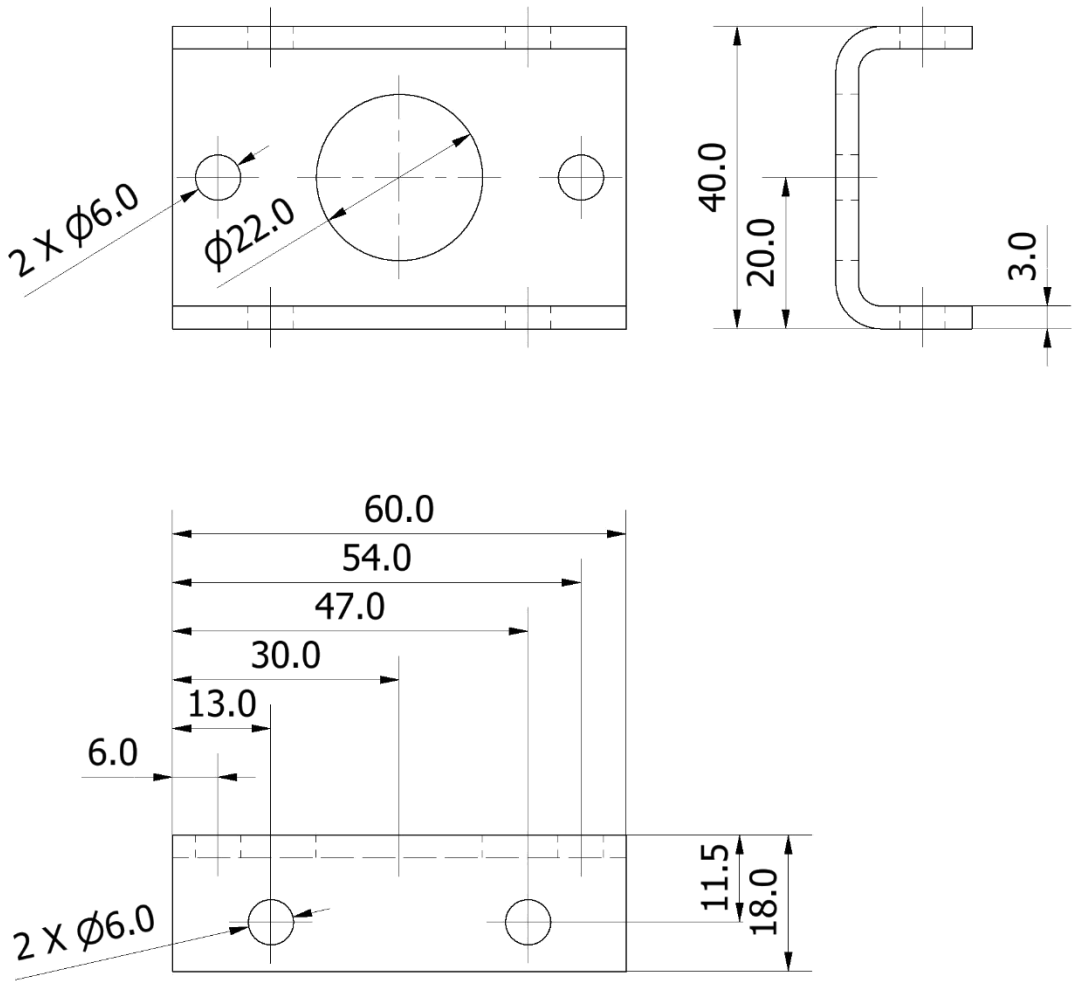
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Arm Motor Support		
	J021	Edition	Sheet 52 / 95



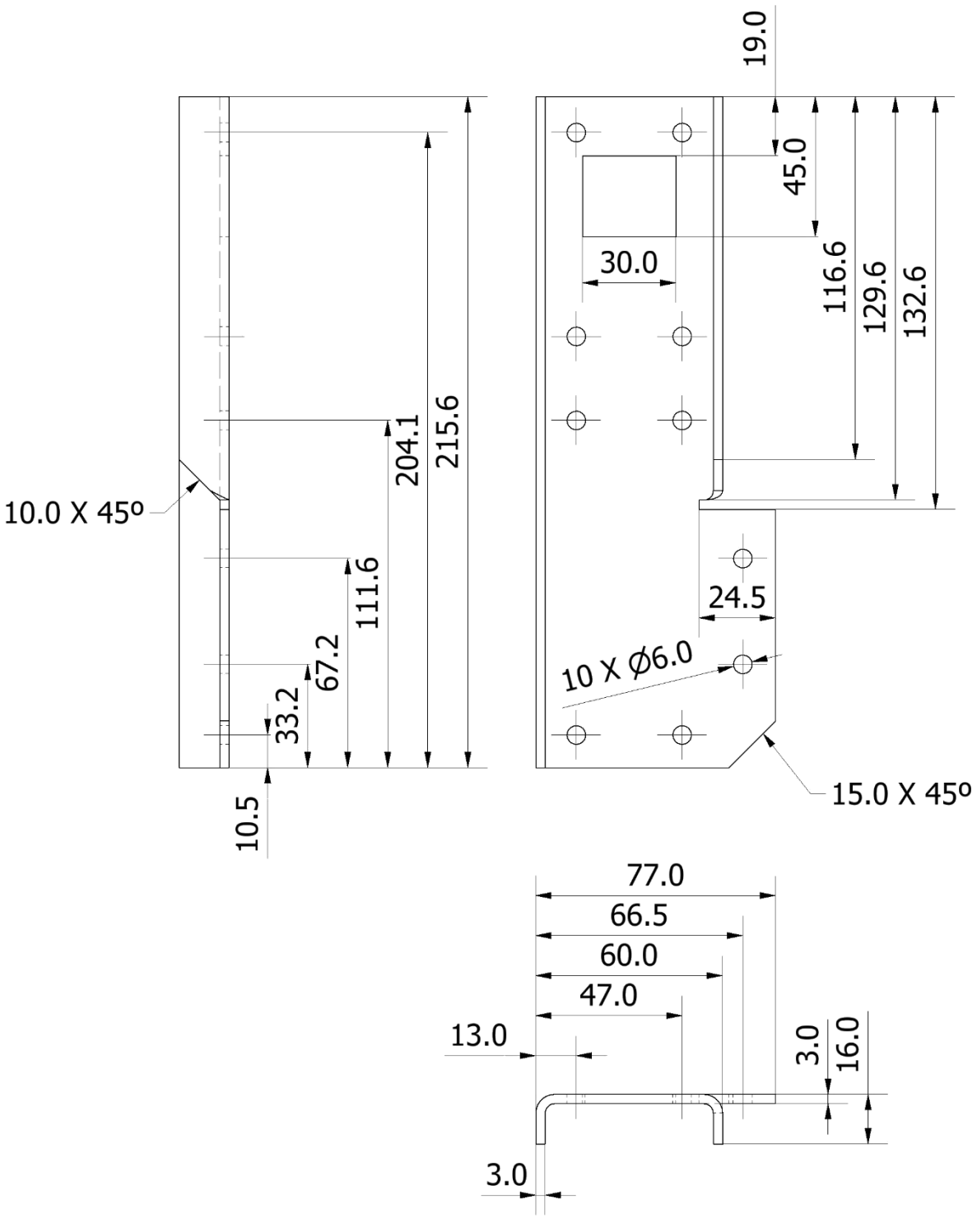
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Arm Bearing Support		
	J022	Edition	Sheet 53 / 95



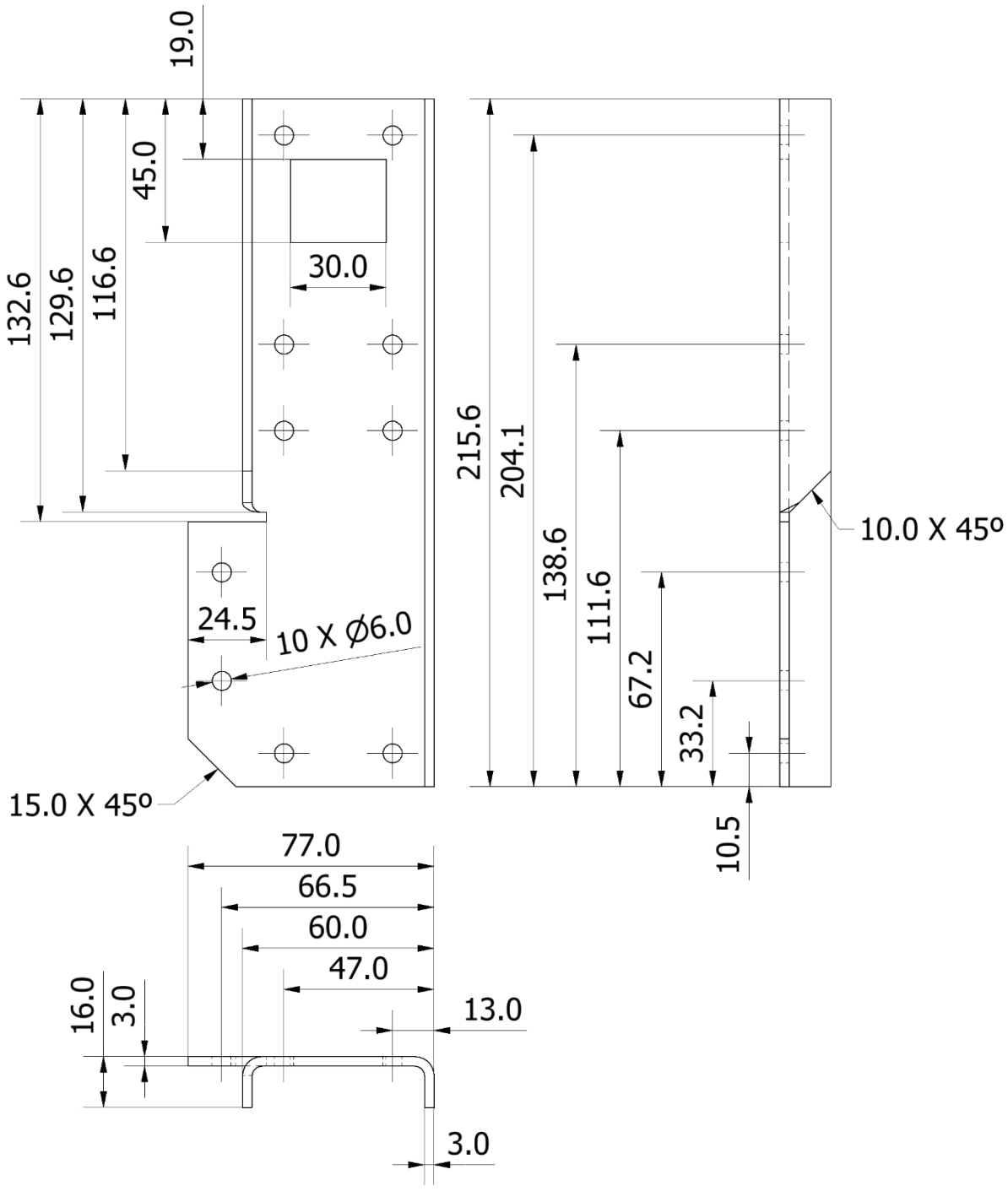
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Arm Shaft Holder Support		
	J023	Edition	Sheet 54 / 95



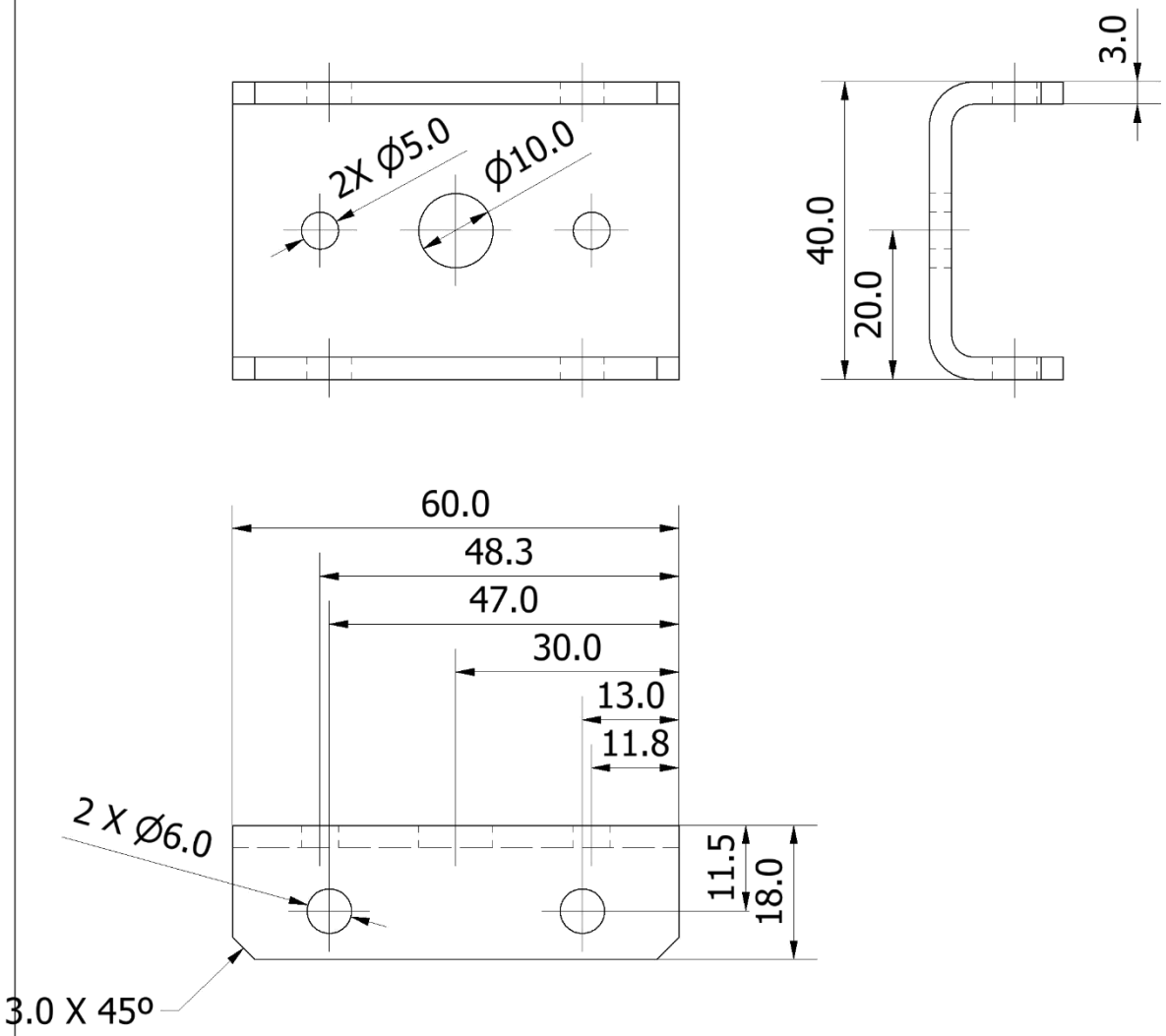
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Distal Arm Outer Casing		
	J030	Edition	Sheet 55 / 95



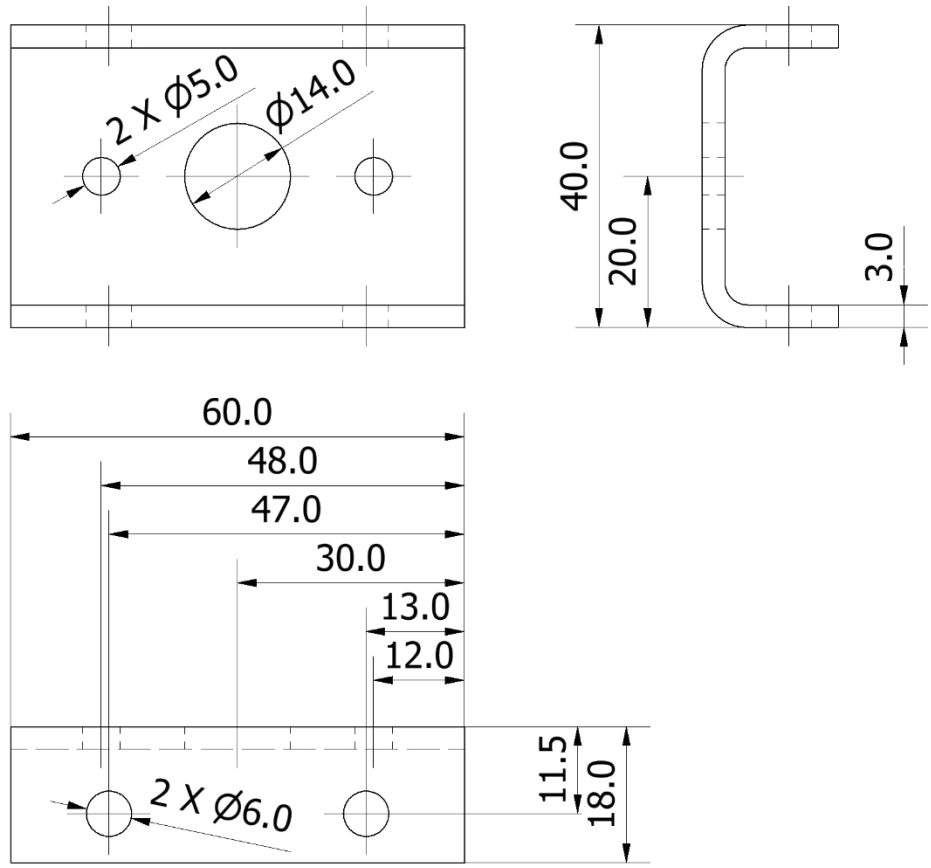
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 2

Universidade do Minho Laboratório de Automação e Robótica	Distal Arm Outer Casing Mirror		
	J031	Edition	Sheet 56 / 95



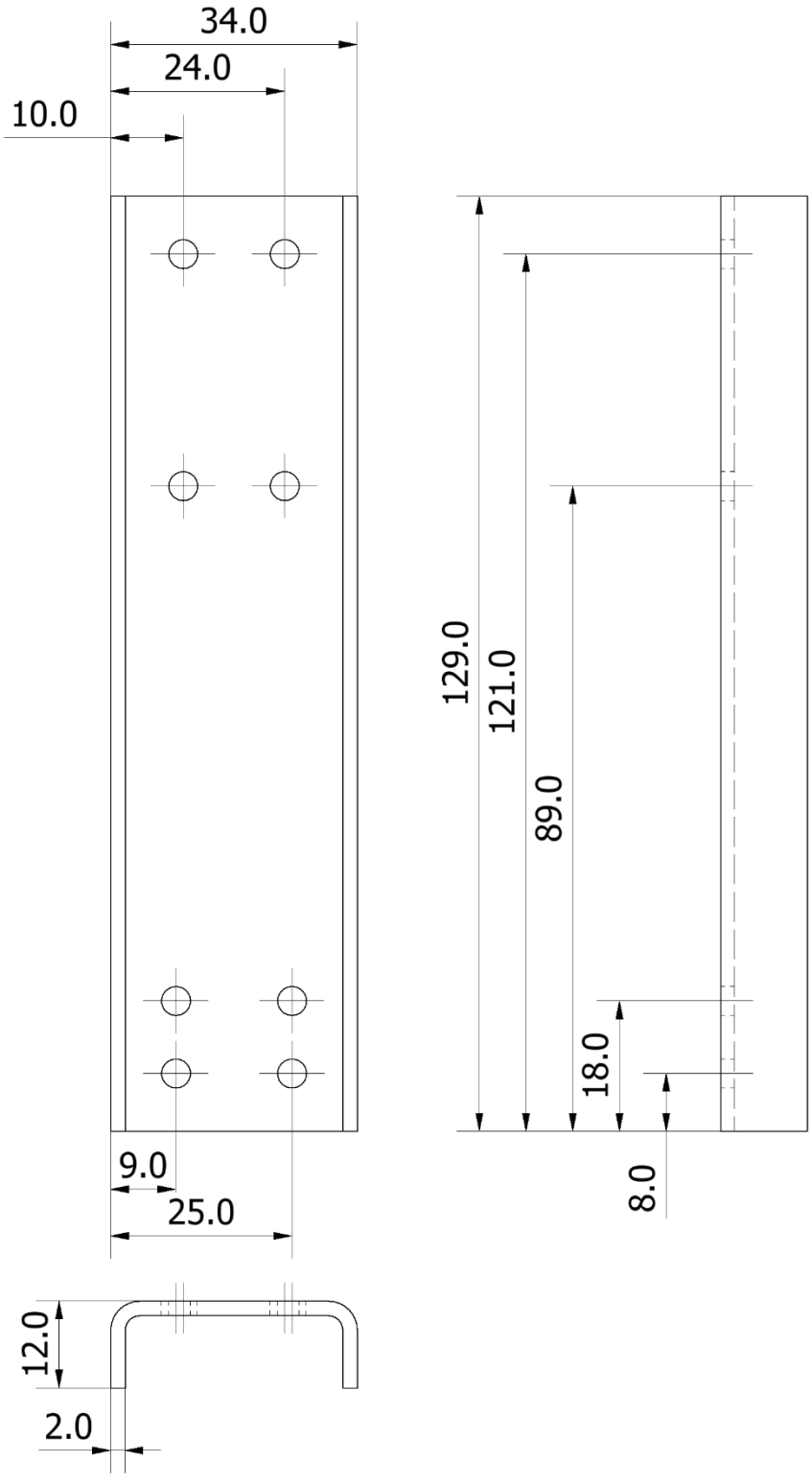
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Arm Bearing Support		
	J032	Edition	Sheet 57 / 95



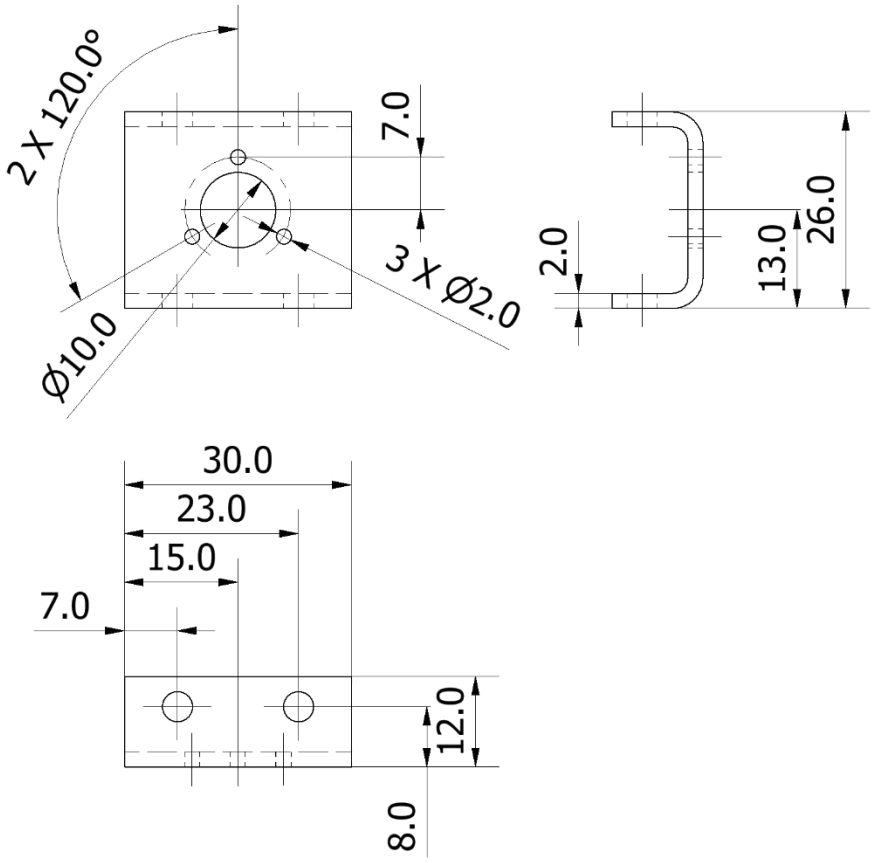
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Arm Shaft Holder Support		
	J033	Edition	Sheet 58 / 95



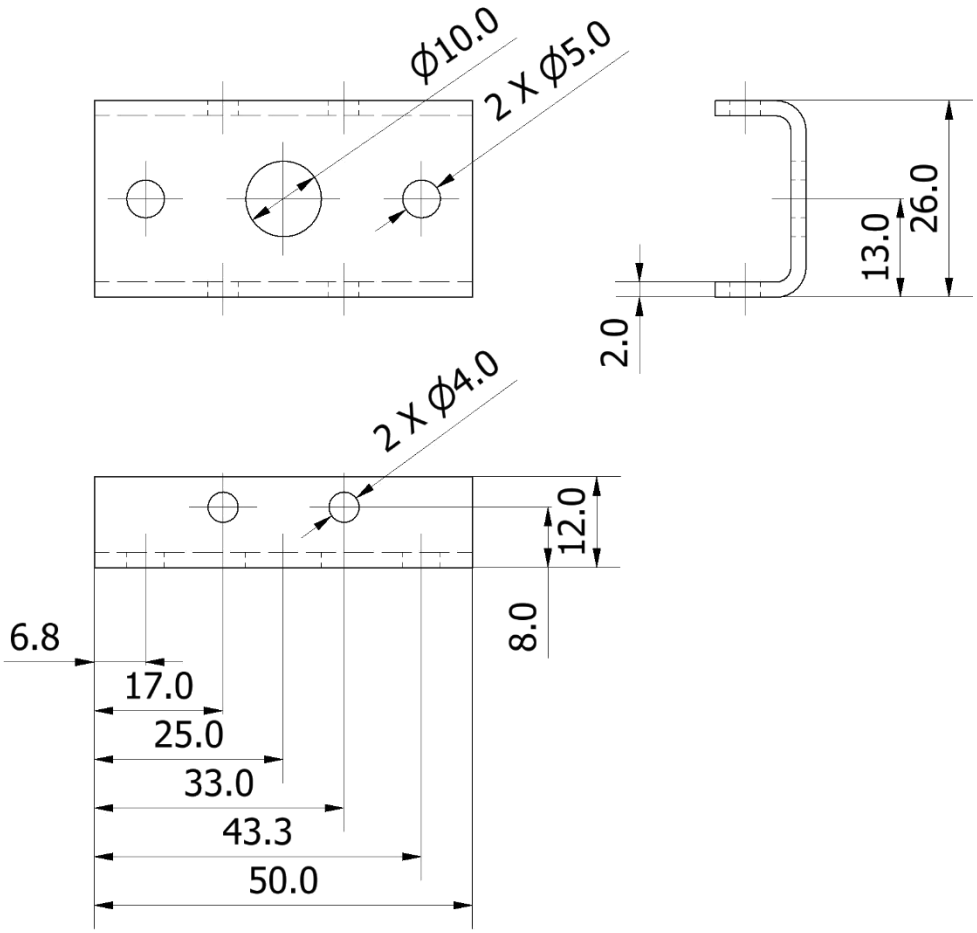
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Outer Casing		
	J040	Edition	Sheet 59 / 95



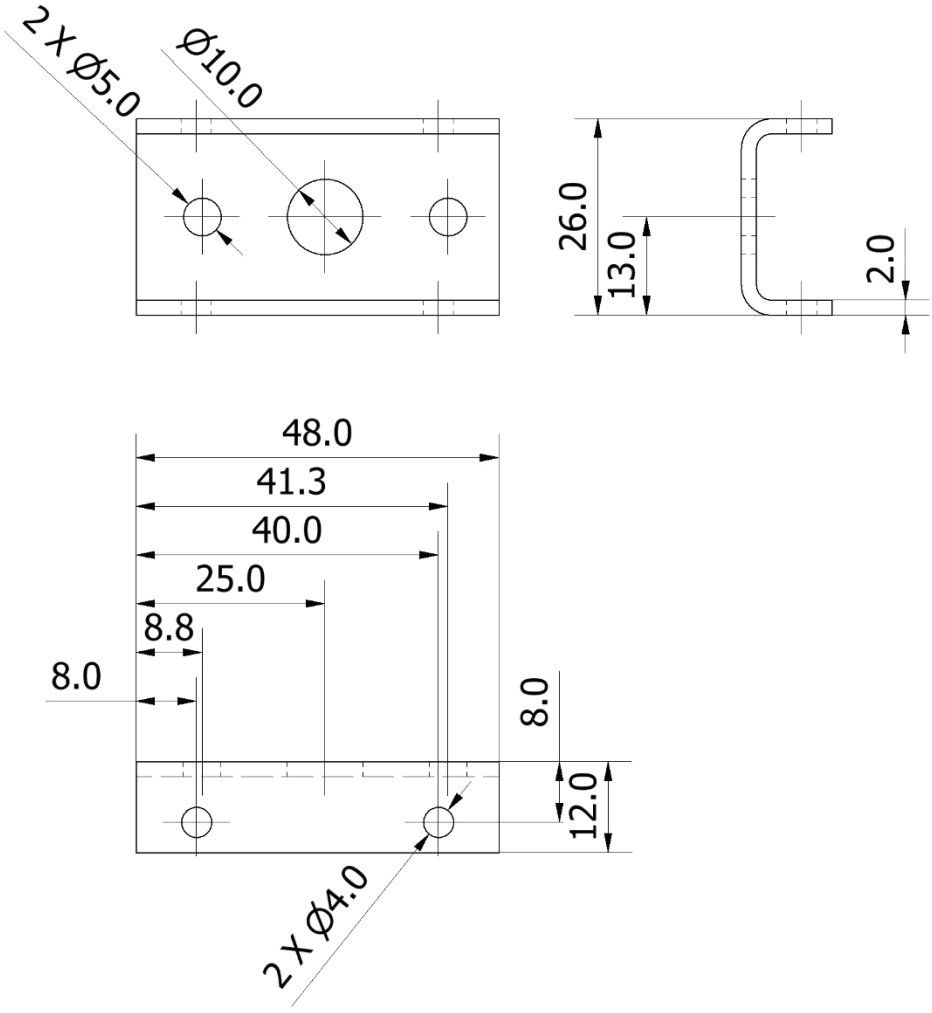
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Motor Support		
	J041	Edition	Sheet 60 / 95



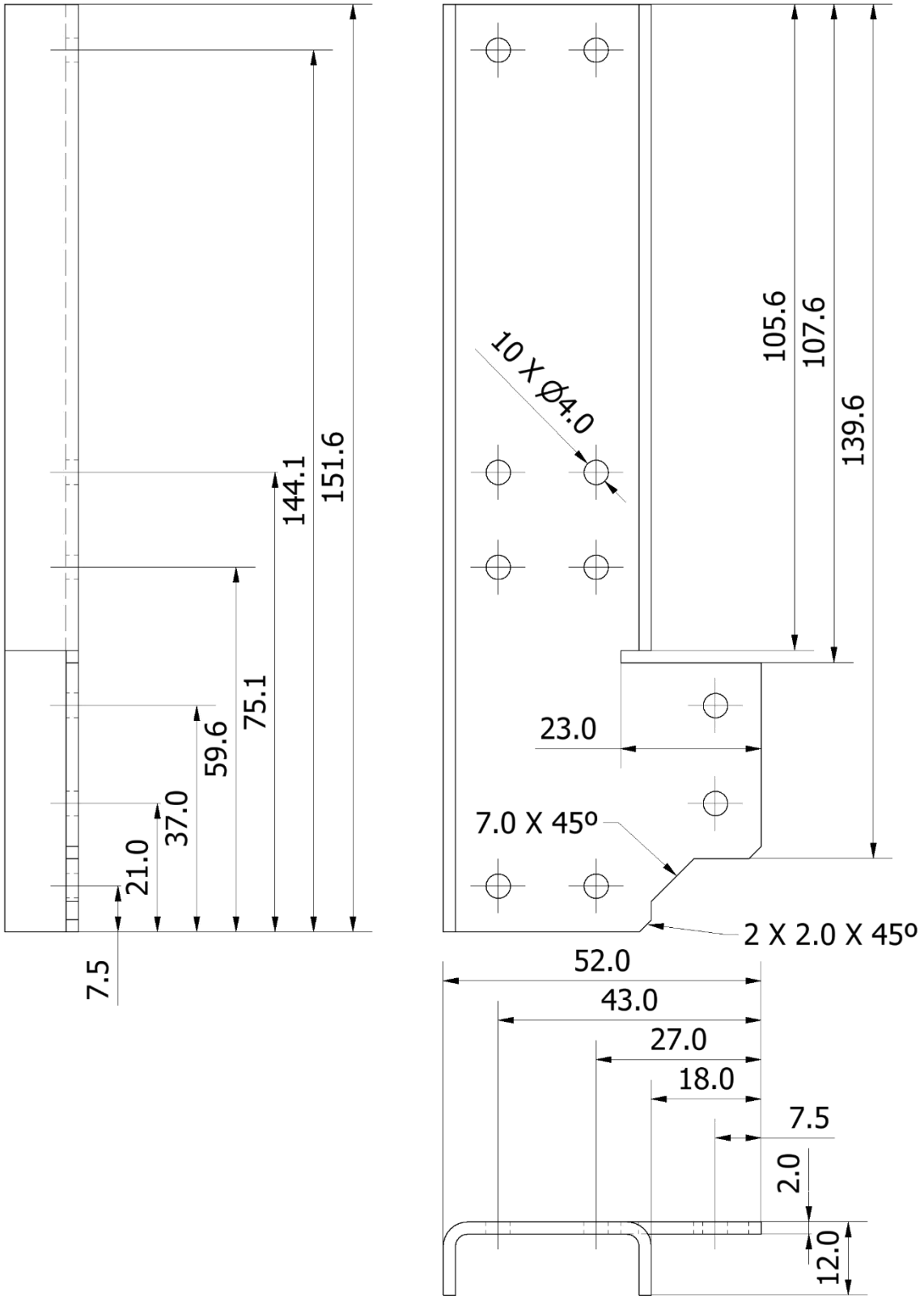
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Bearing Support		
	J042	Edition	Sheet 61 / 95



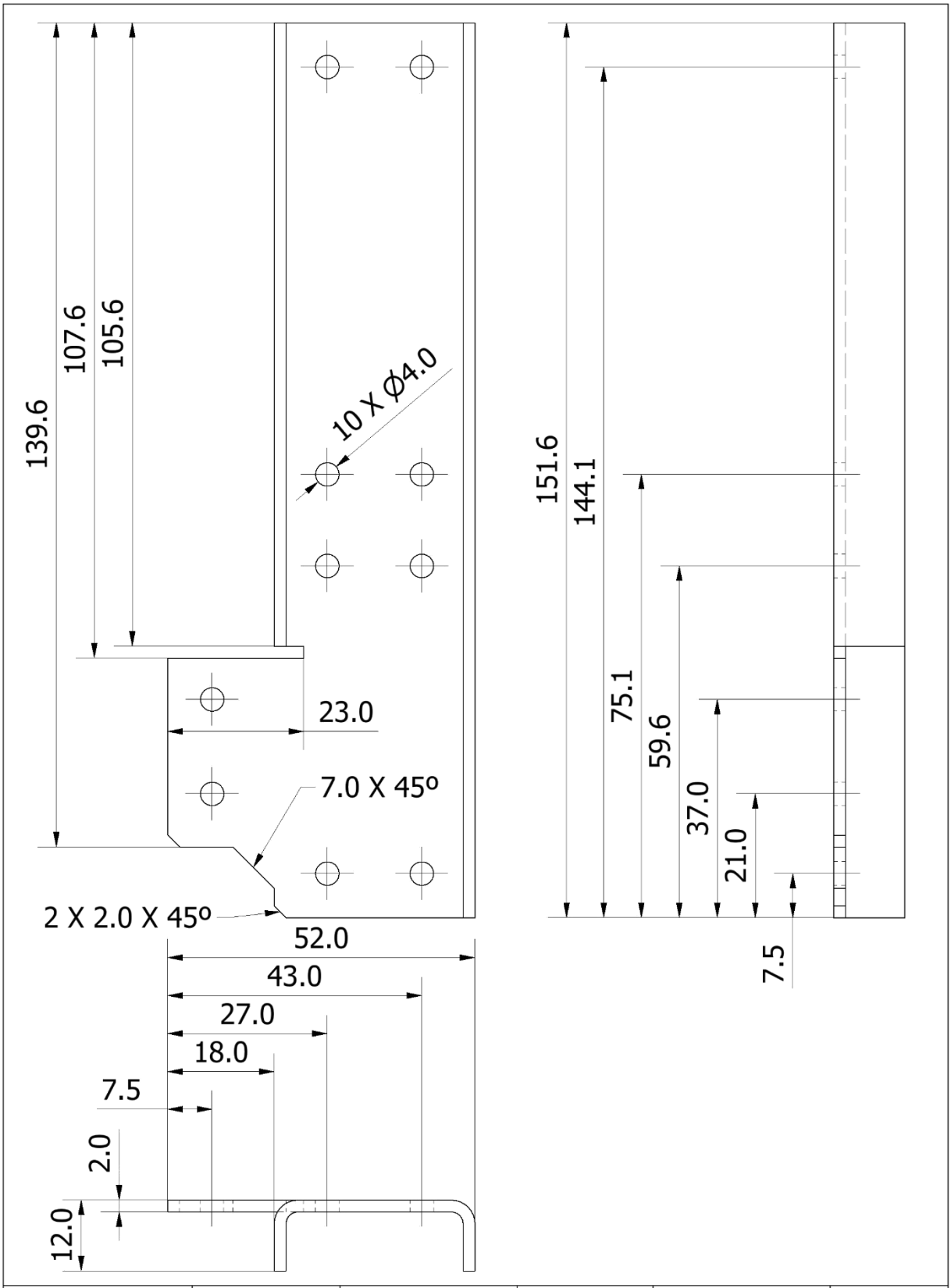
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Shaft Holder Support		
	J043	Edition	Sheet 62 / 95



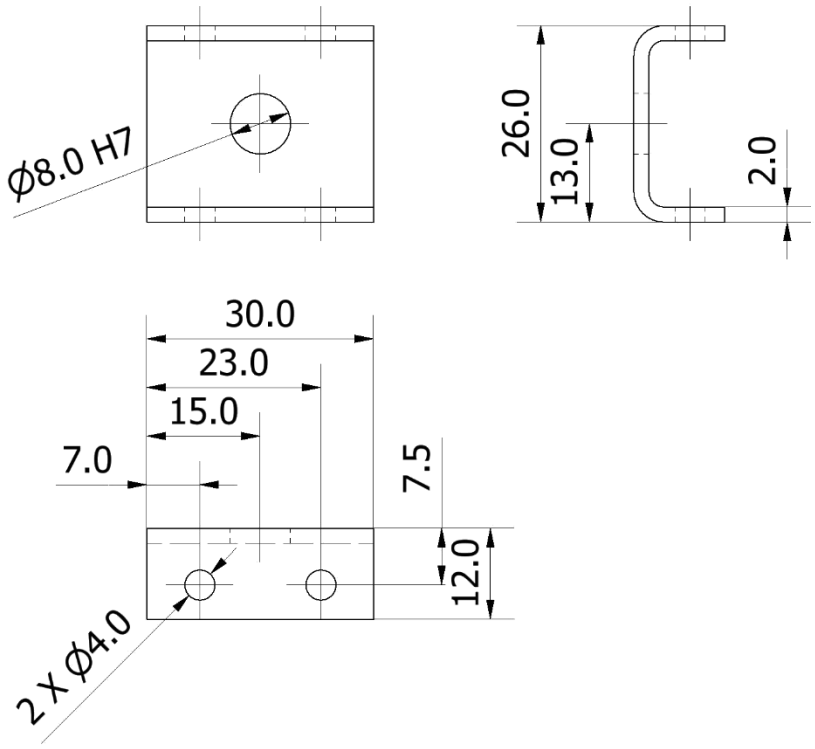
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm Outer Casing		
	J050	Edition	Sheet 63 / 95



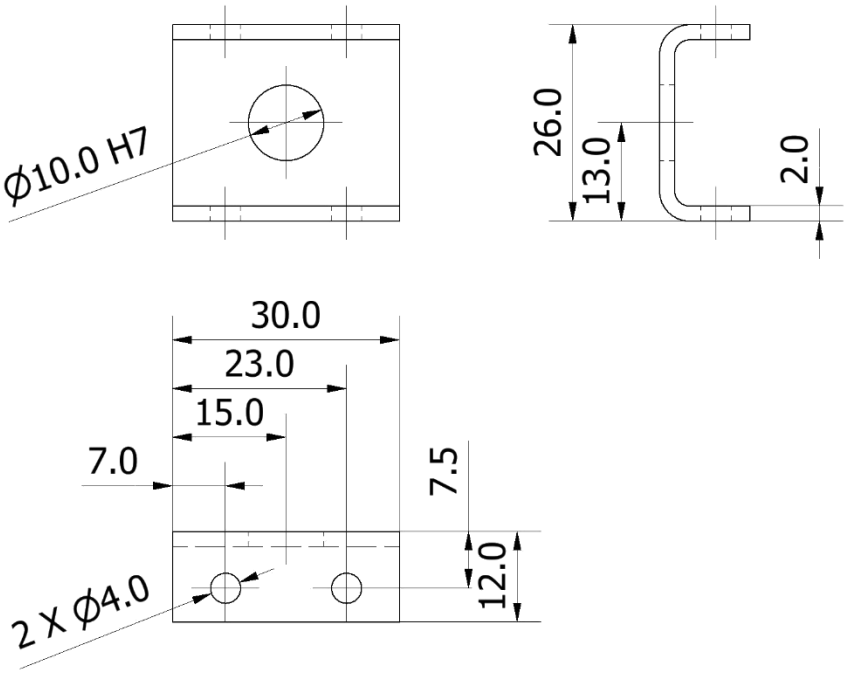
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm Outer Casing Mirror		
	J051	Edition	Sheet 64 / 95



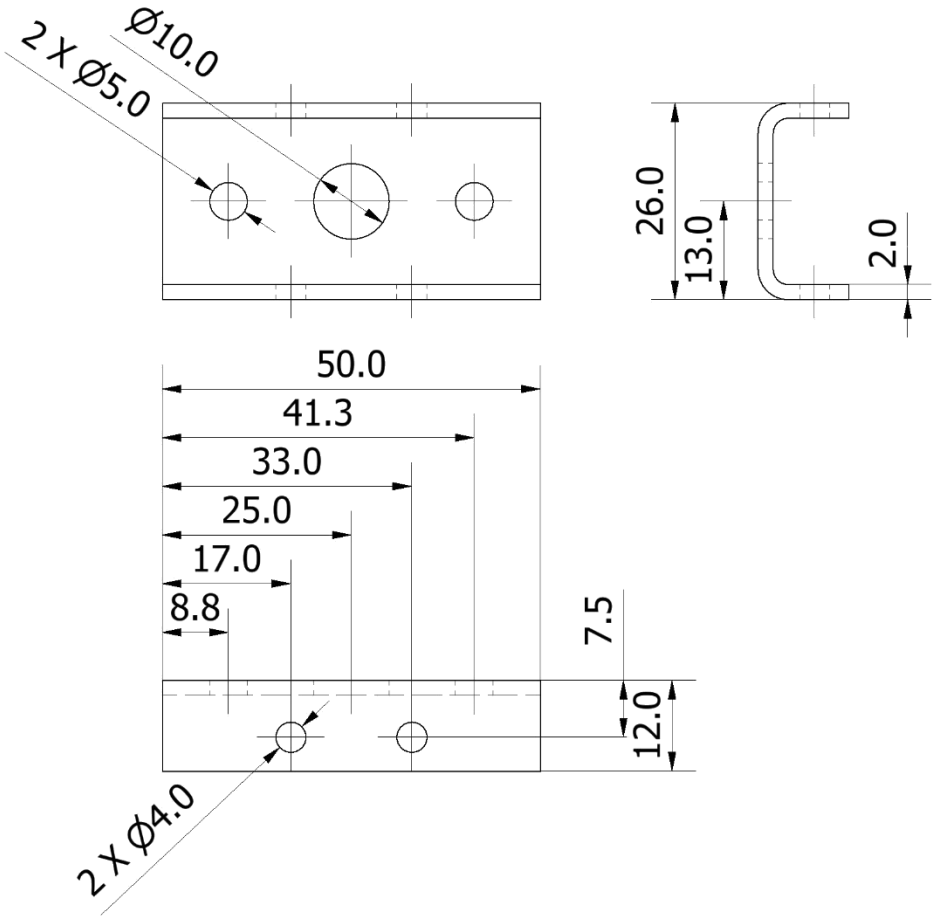
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm 5 mm Bearing Support		
	J052	Edition	Sheet 65 / 95



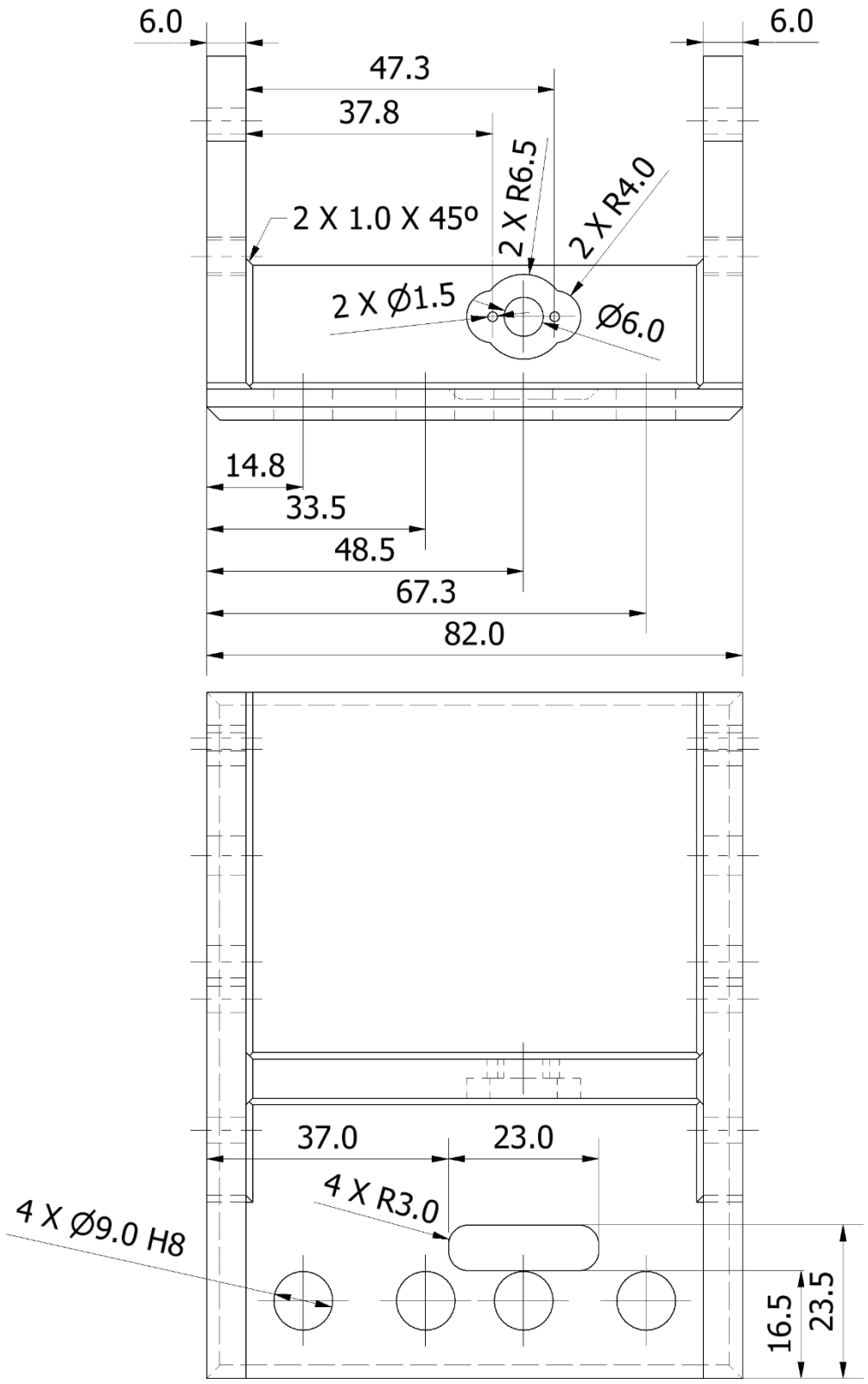
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm 6 mm Bearing Support		
	J053	Edition	Sheet 66 / 95



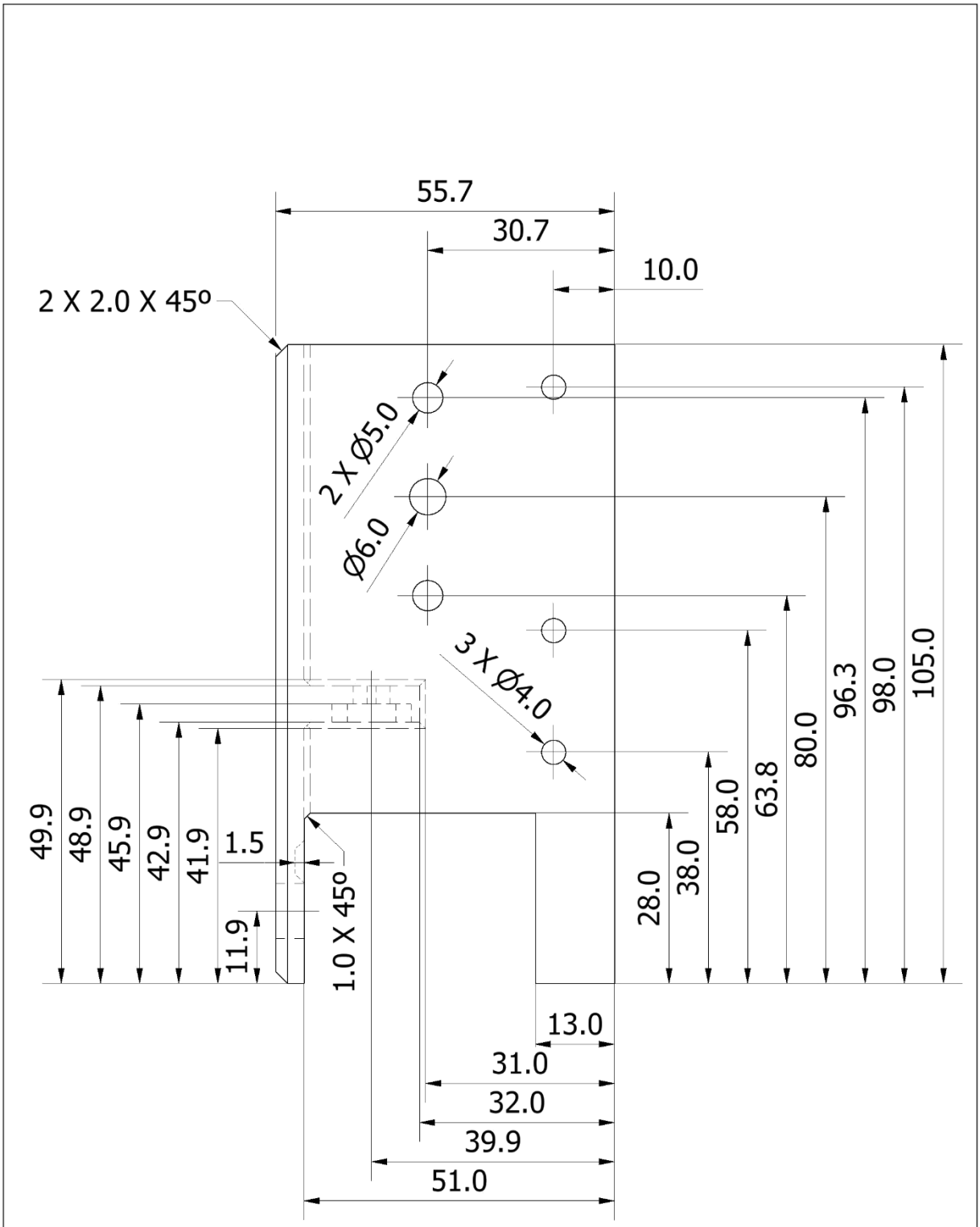
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm Shaft Holder Support		
	J054	Edition	Sheet 67 / 95



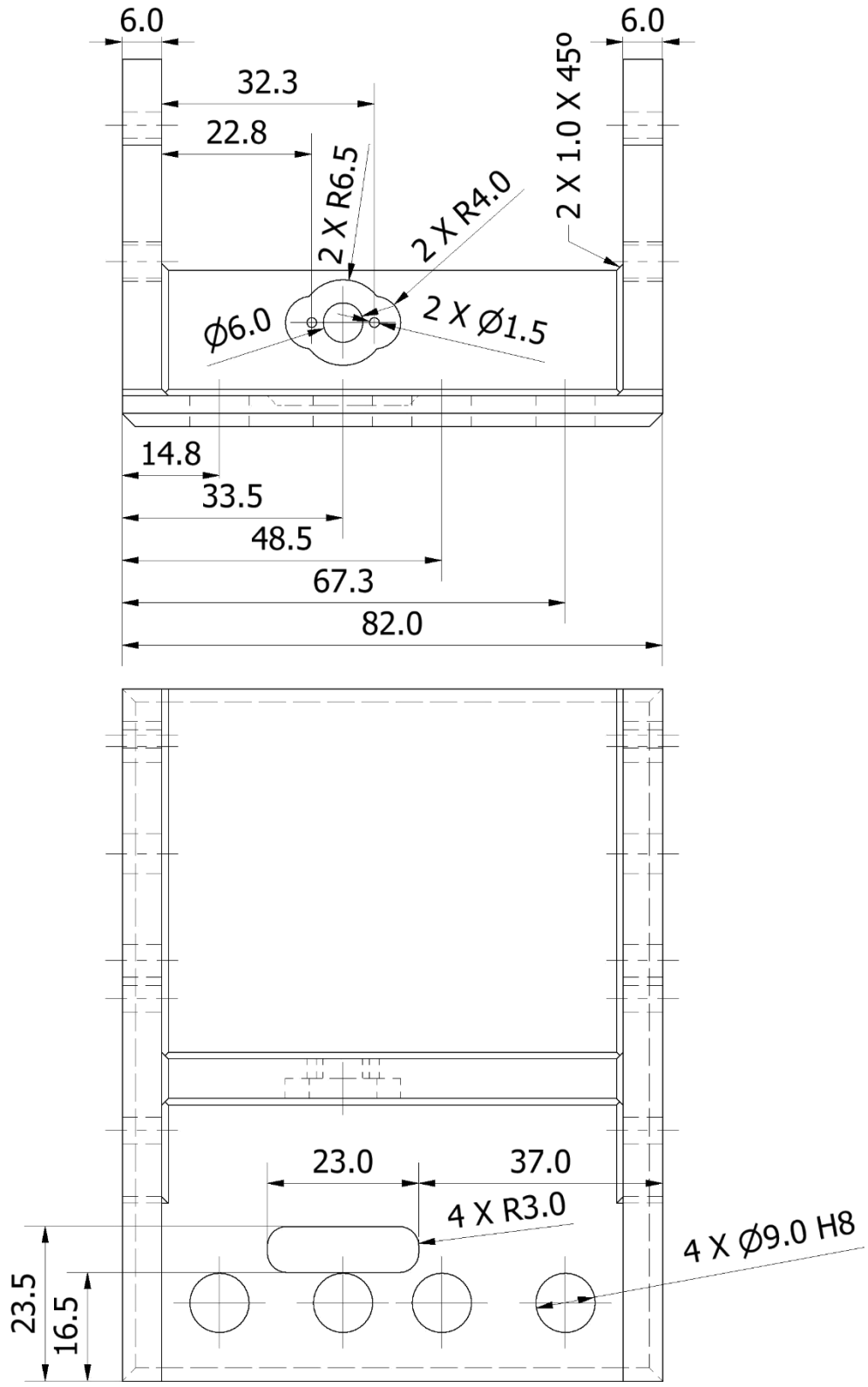
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Lower Half		
	J060 - Sheet 1 / 2	Edition	Sheet 68 / 95



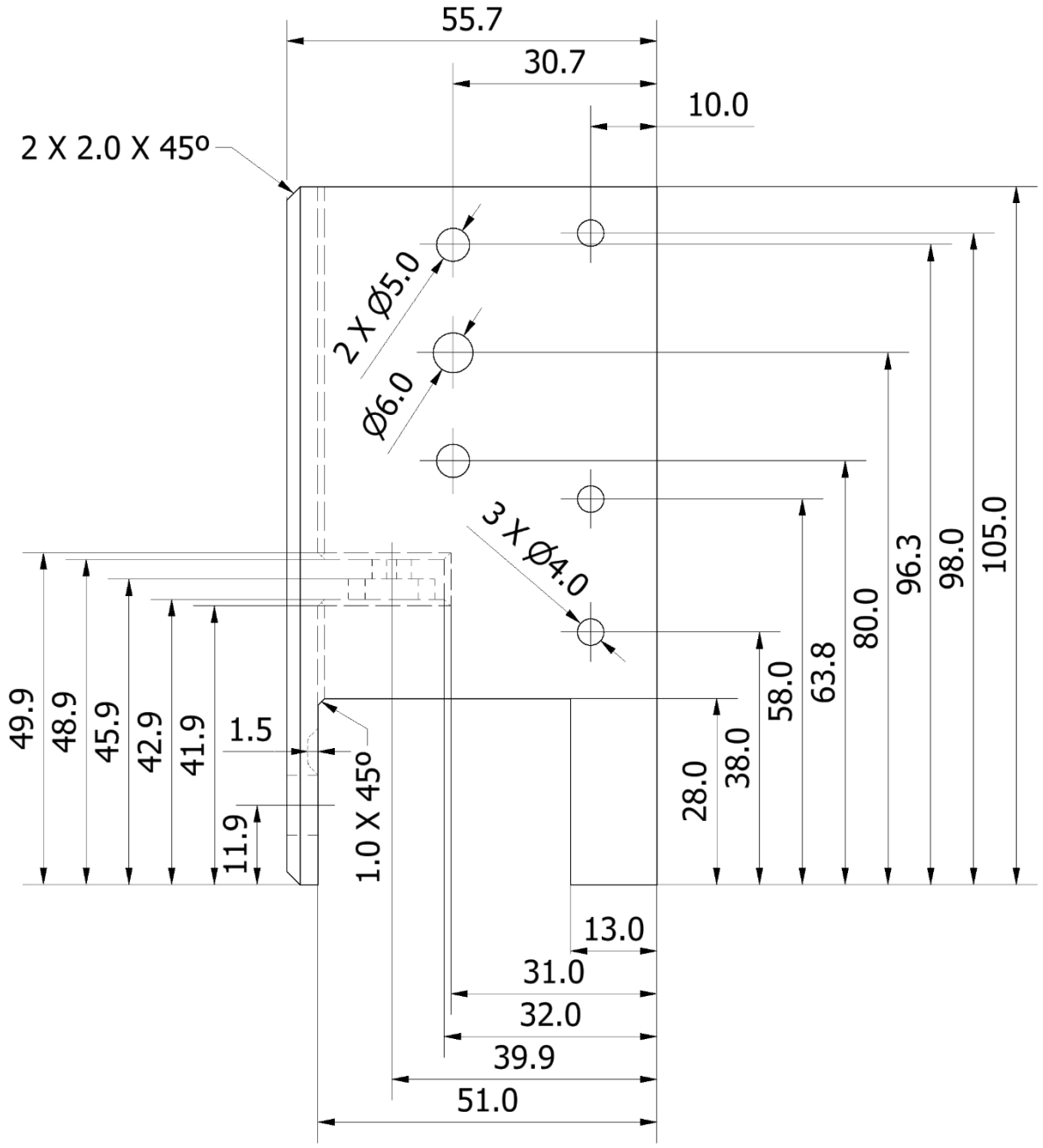
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Lower Half		
	J060 - Sheet 2 / 2	Edition	Sheet 69 / 95



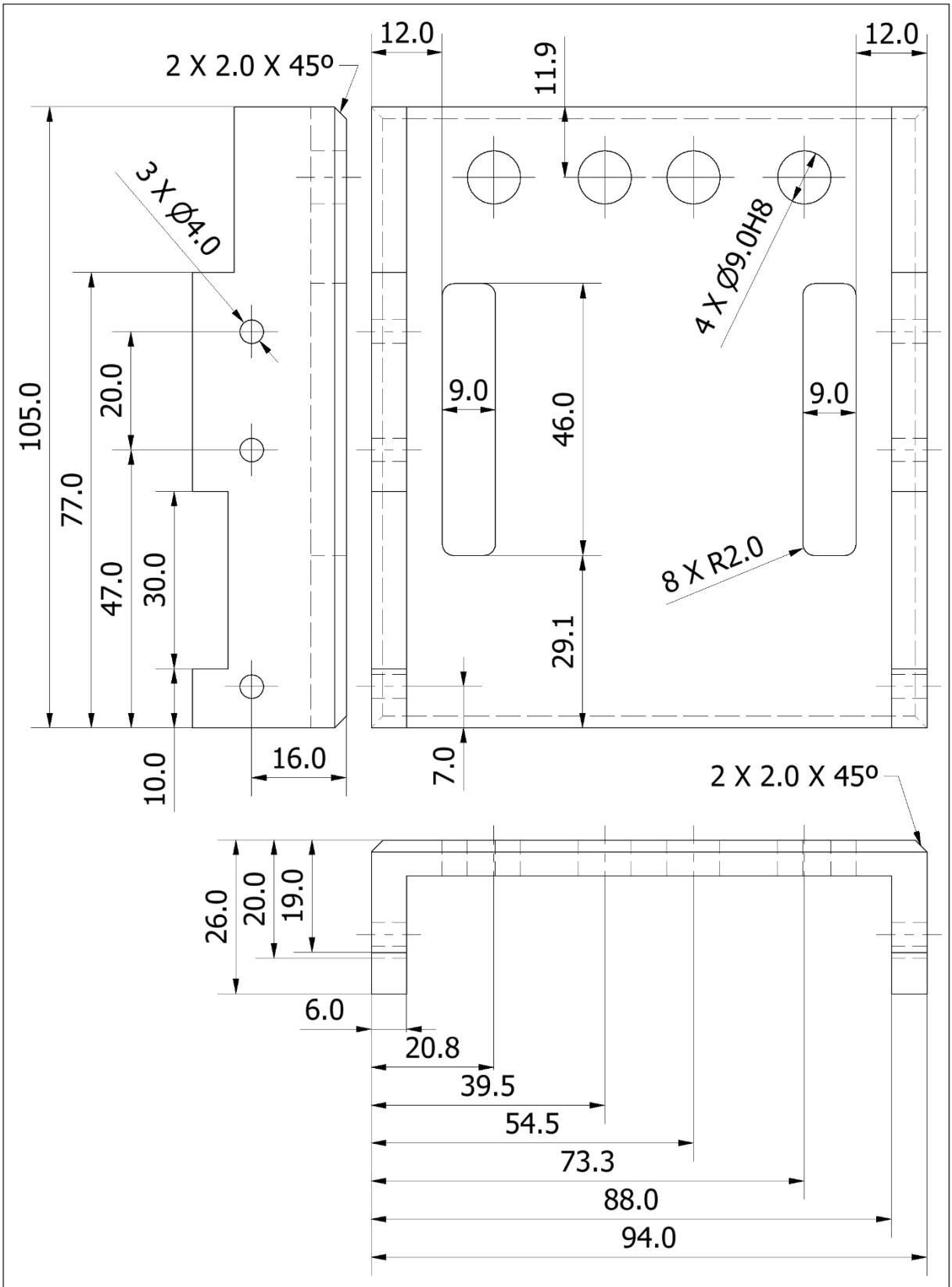
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Lower Half Mirror		
	J061 - Sheet 1 / 2	Edition	Sheet 70 / 95



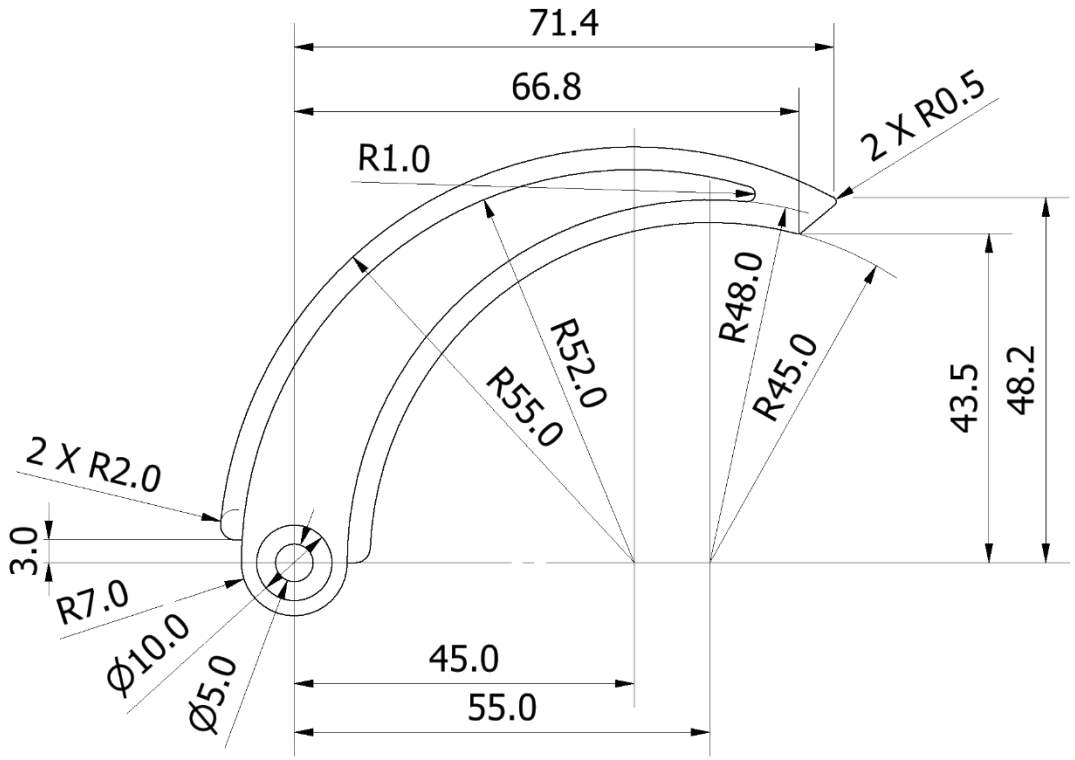
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

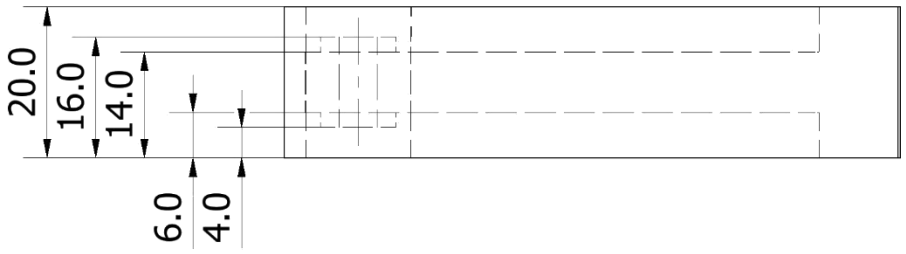
Universidade do Minho Laboratório de Automação e Robótica	Hand Lower Half Mirror		
	J061 - Sheet 2 / 2	Edition	Sheet 71 / 95



Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

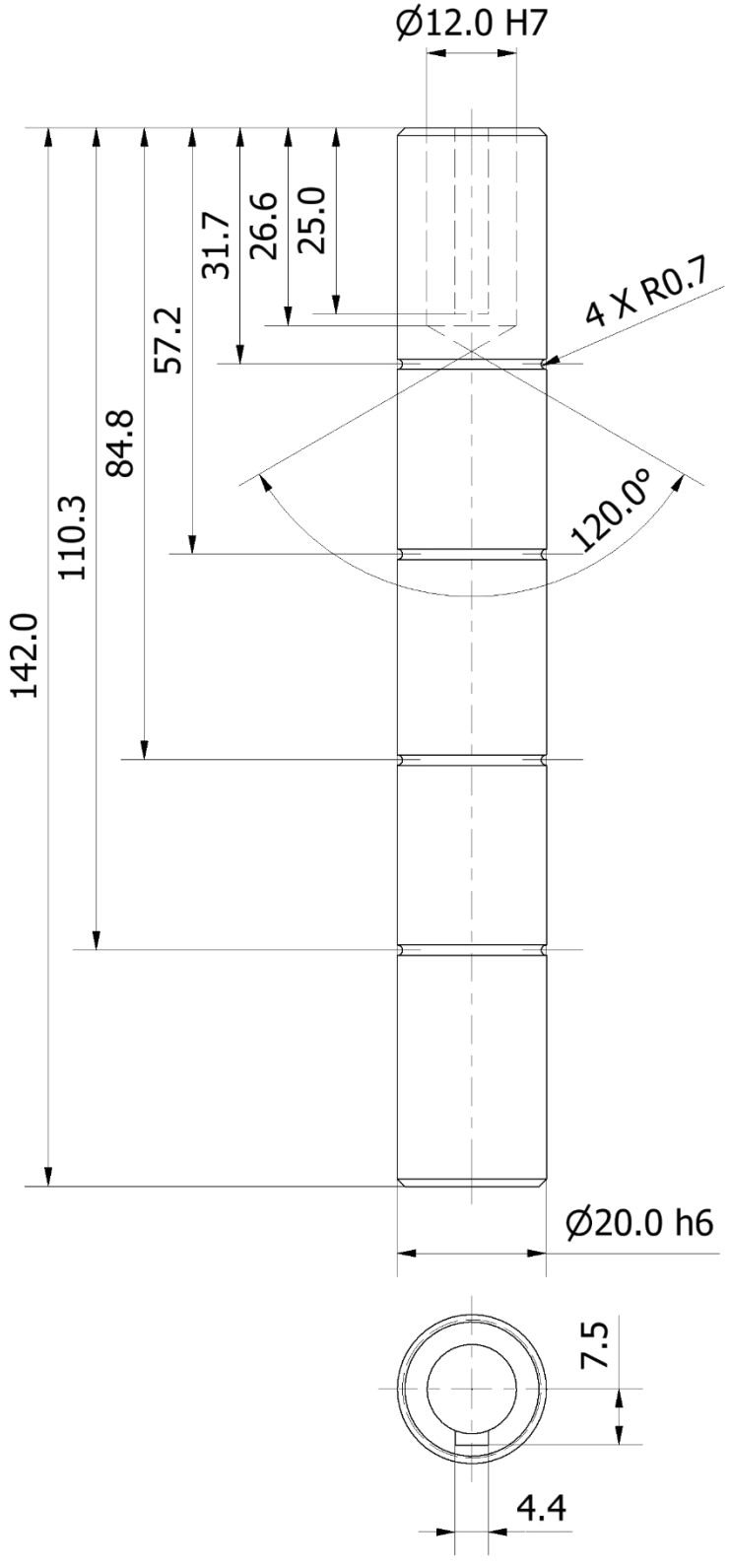
Universidade do Minho Laboratório de Automação e Robótica	Hand Top Half		
	J062	Edition	Sheet 72 / 95





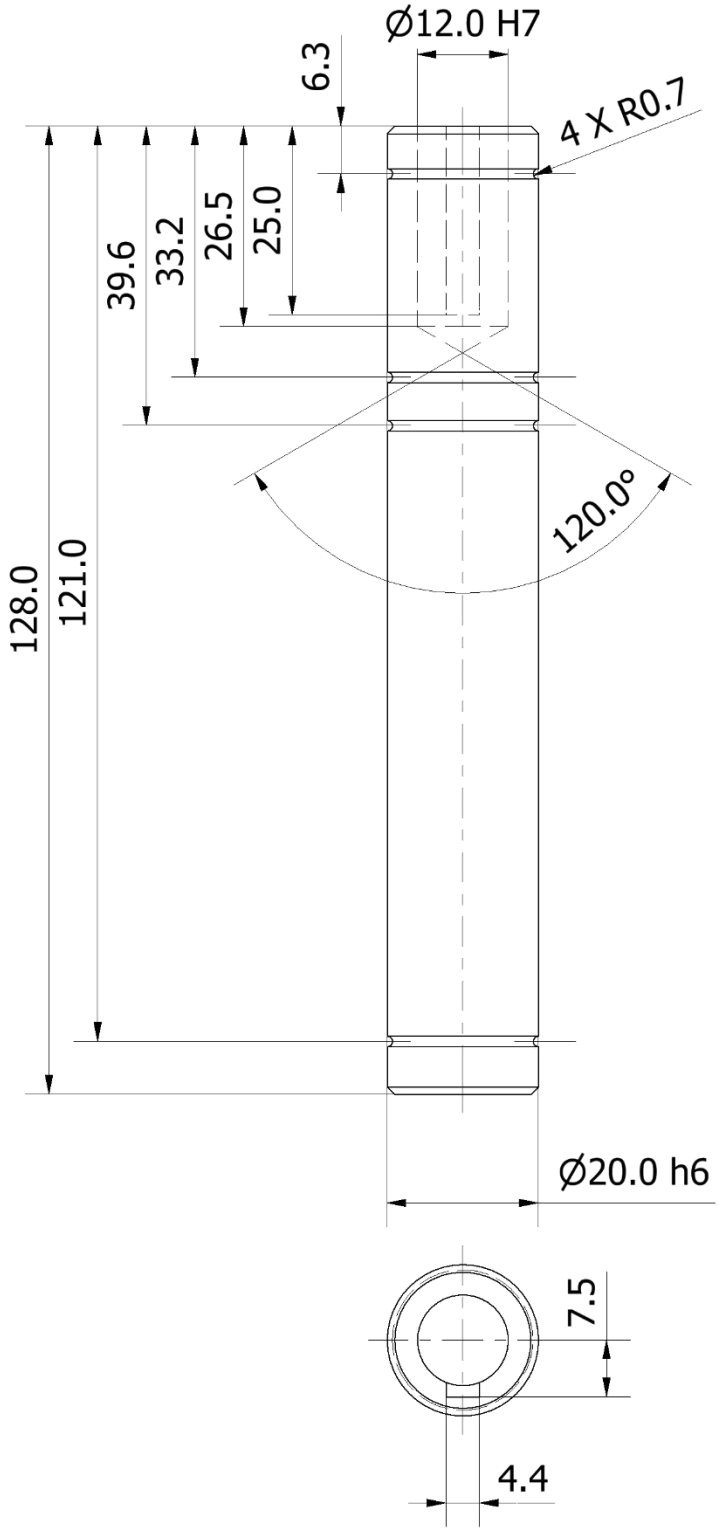
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Claw		
	J070	Edition	Sheet 73 / 95



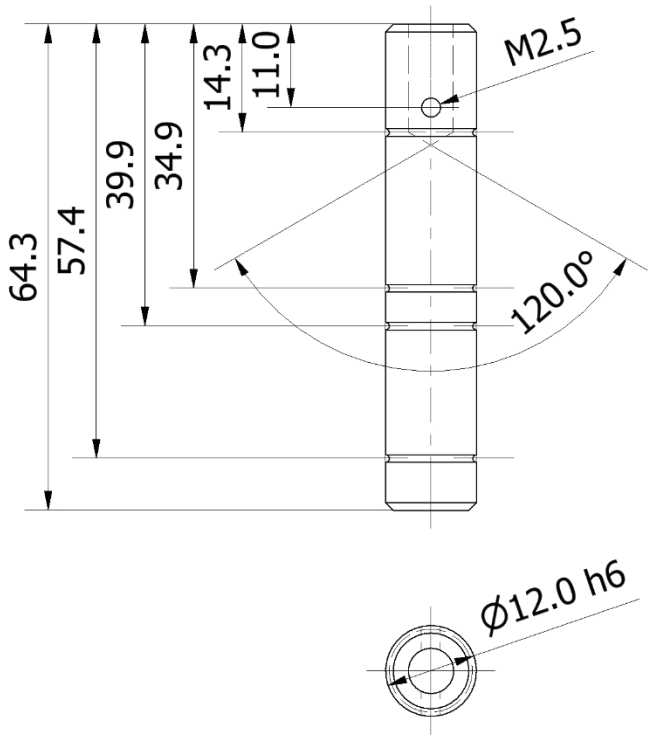
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Scapula Motor Shaft		
	K000	Edition	Sheet 74 / 95



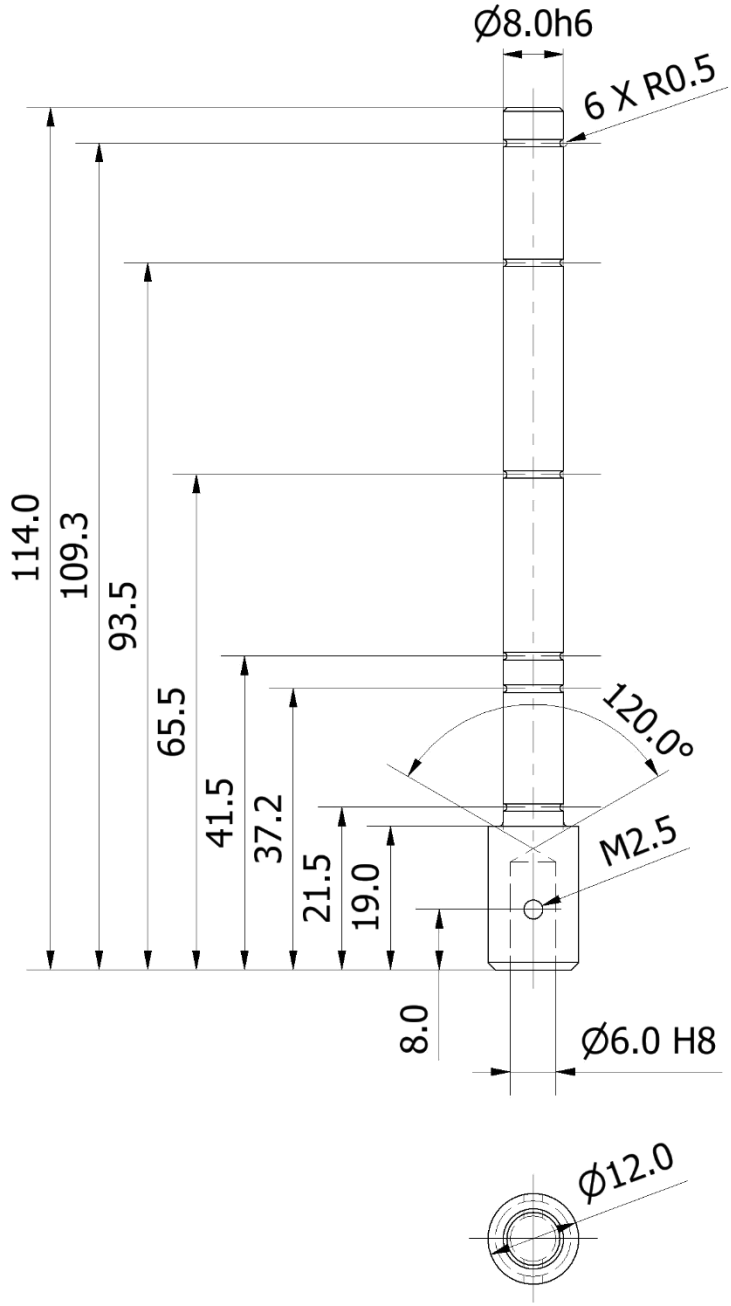
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Shoulder Motor Shaft		
	K001	Edition	Sheet 75 / 95



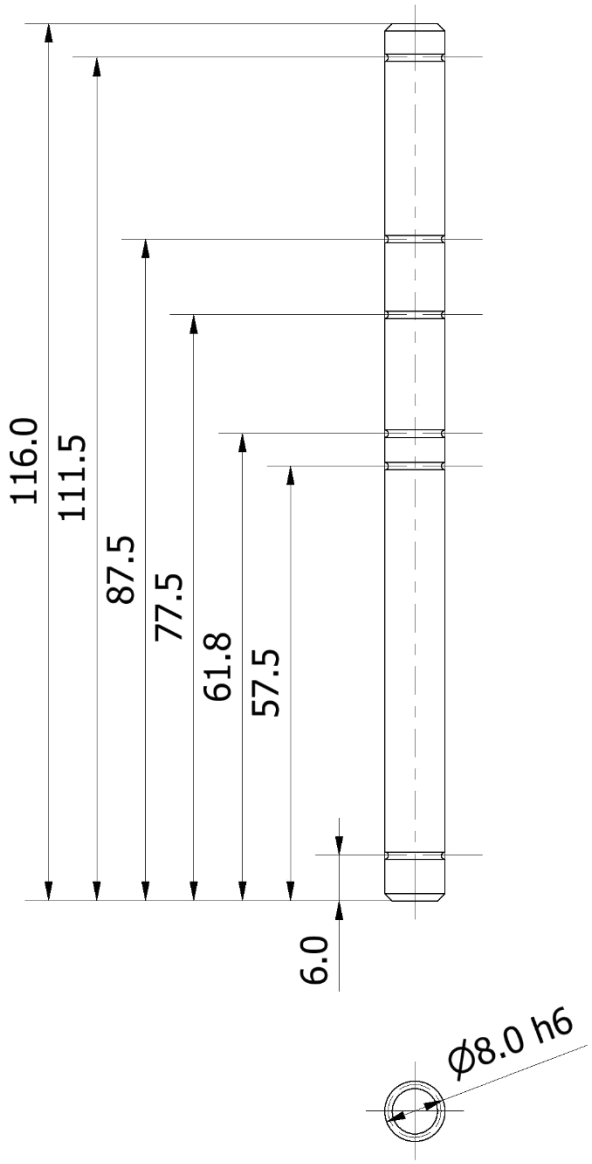
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Arm Motor Shaft		
	K002	Edition	Sheet 76 / 95



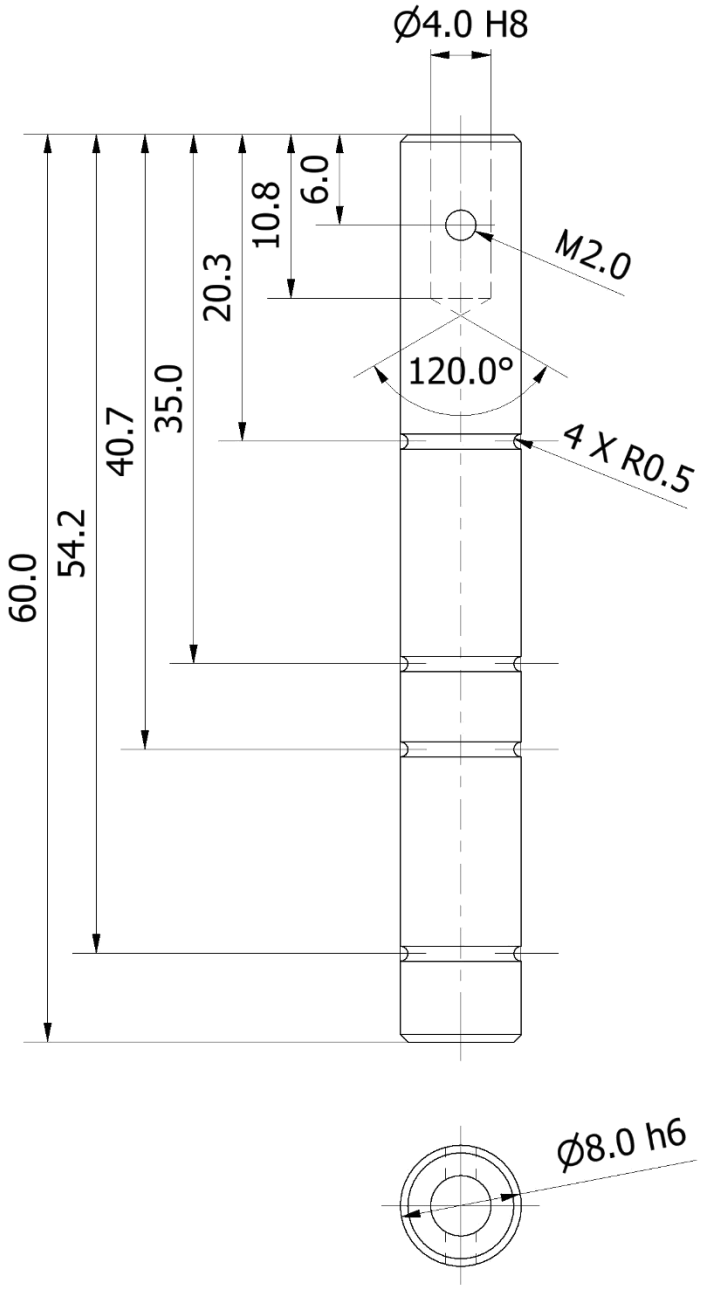
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Arm Motor Shaft		
	K003	Edition	Sheet 77 / 95



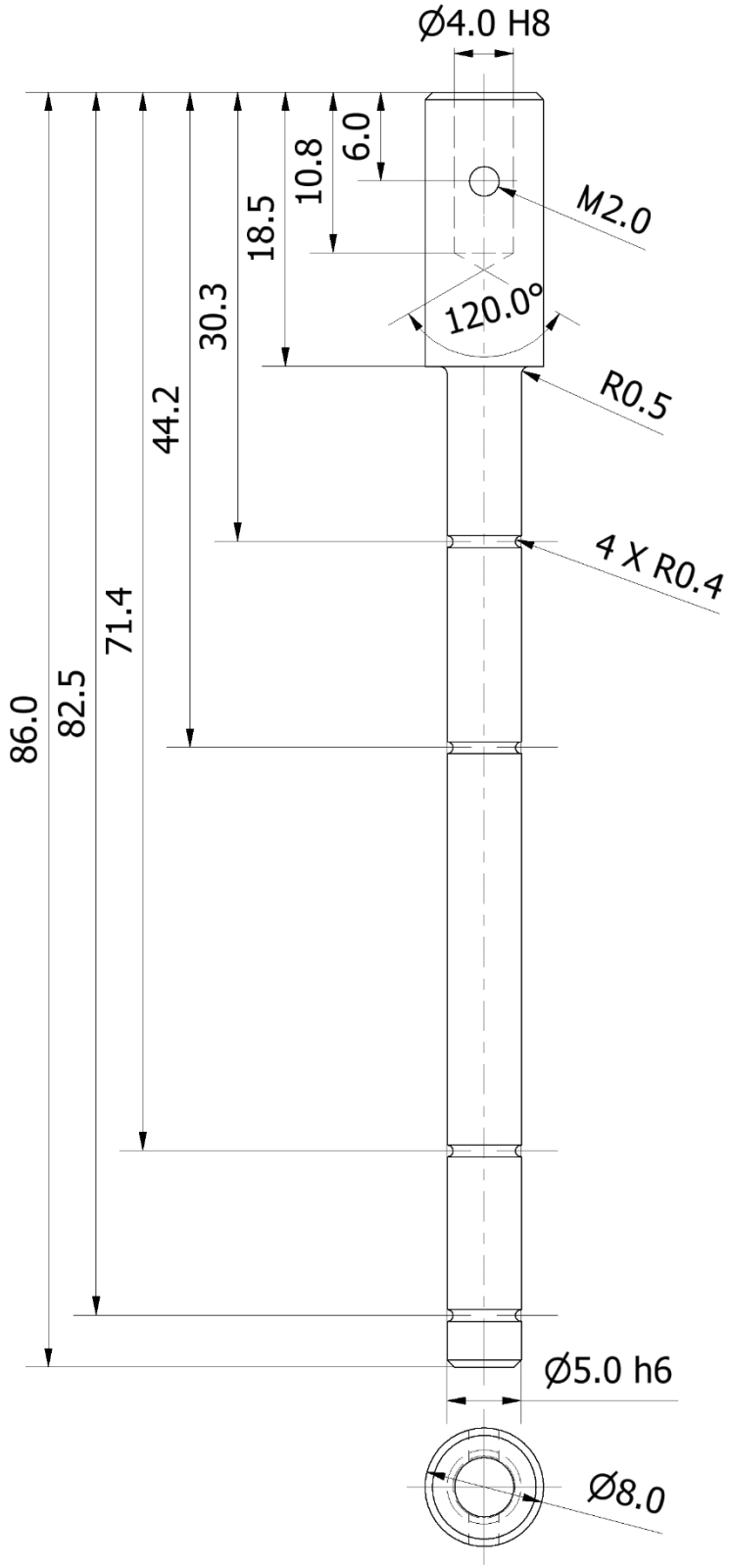
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Input Shaft		
	K004	Edition	Sheet 78 / 95



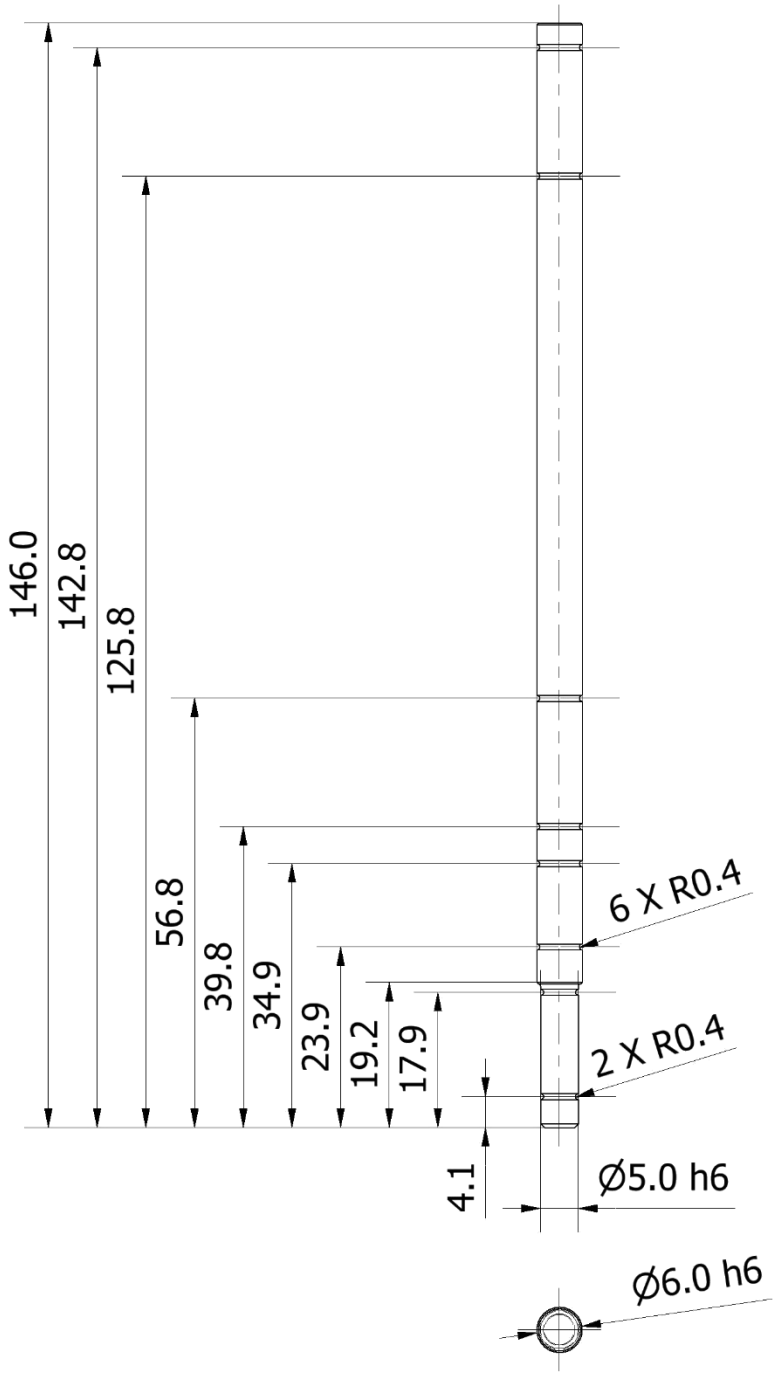
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	2 : 1

Universidade do Minho Laboratório de Automação e Robótica	Proximal Forearm Motor Shaft		
	K005	Edition	Sheet 79 / 95



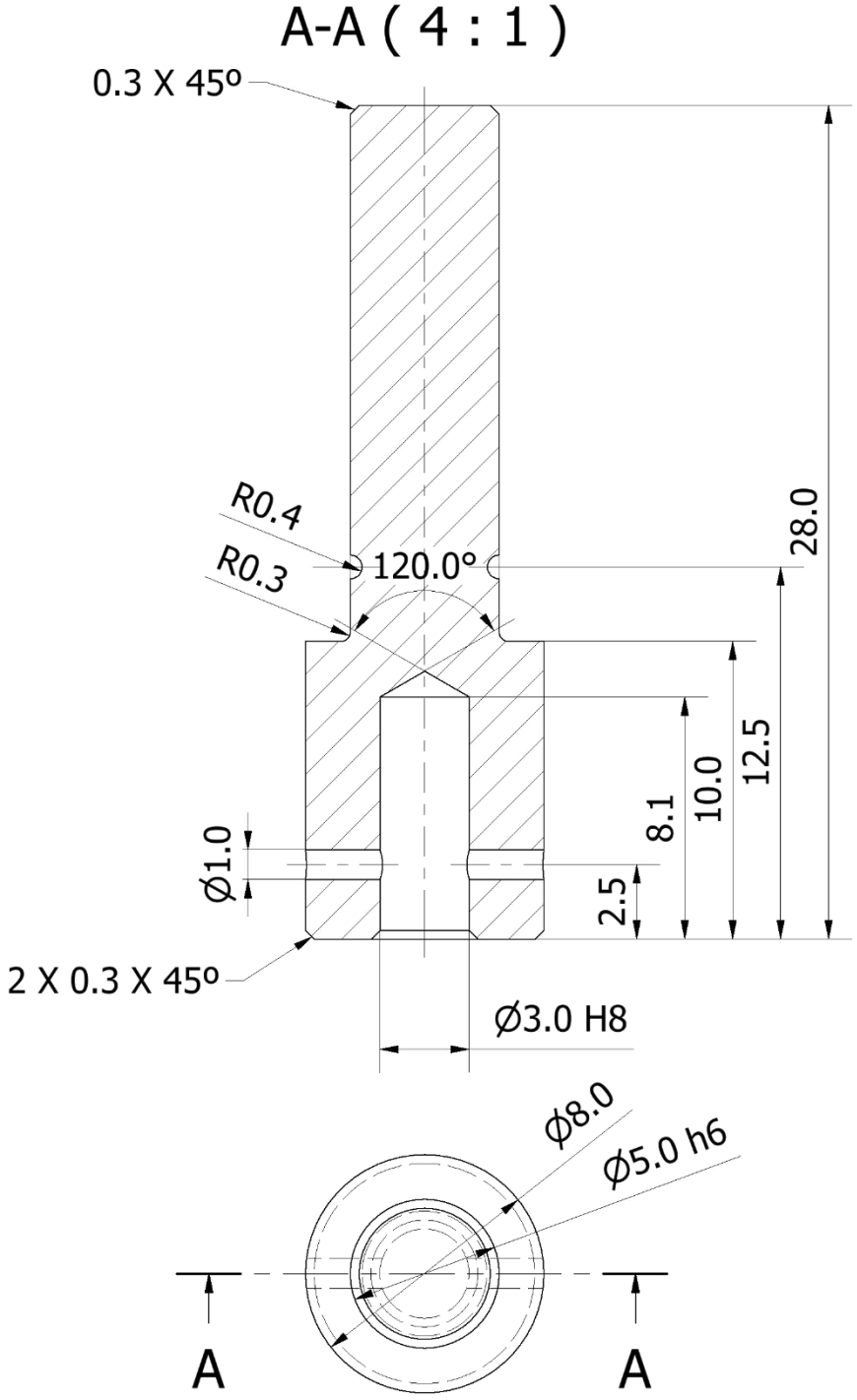
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	2 : 1

Universidade do Minho Laboratório de Automação e Robótica	Distal Forearm Motor Shaft		
	K006	Edition	Sheet 80 / 95



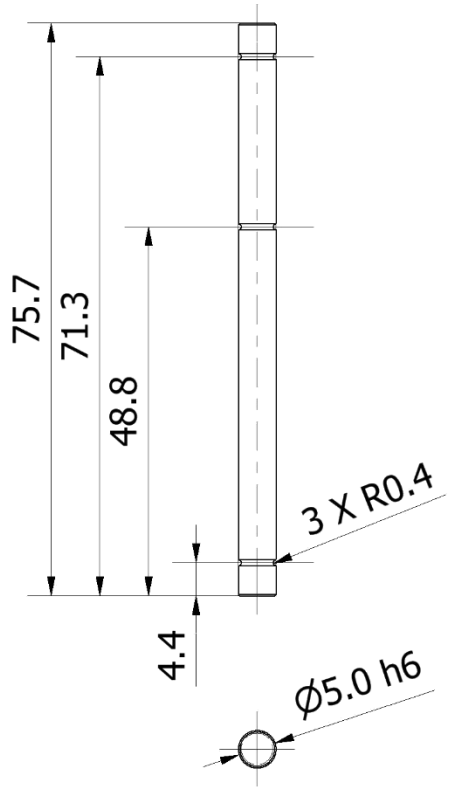
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Input Shaft		
	K007	Edition	Sheet 81 / 95



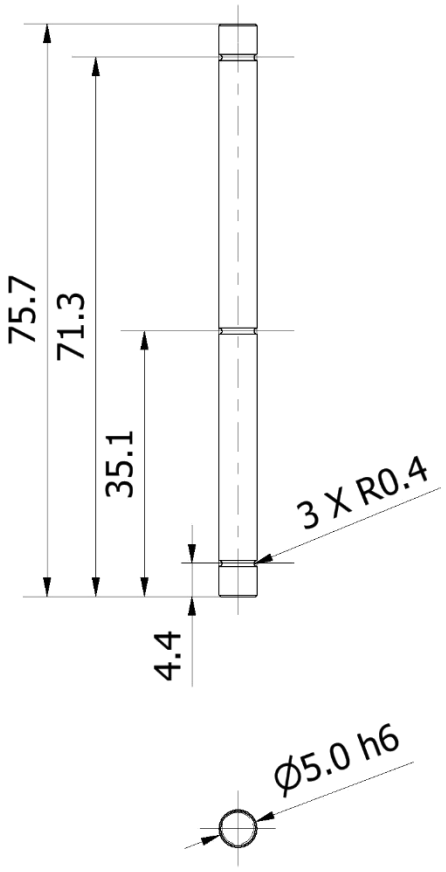
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	4 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Motor Shaft		
	K008	Edition	Sheet 82 / 95



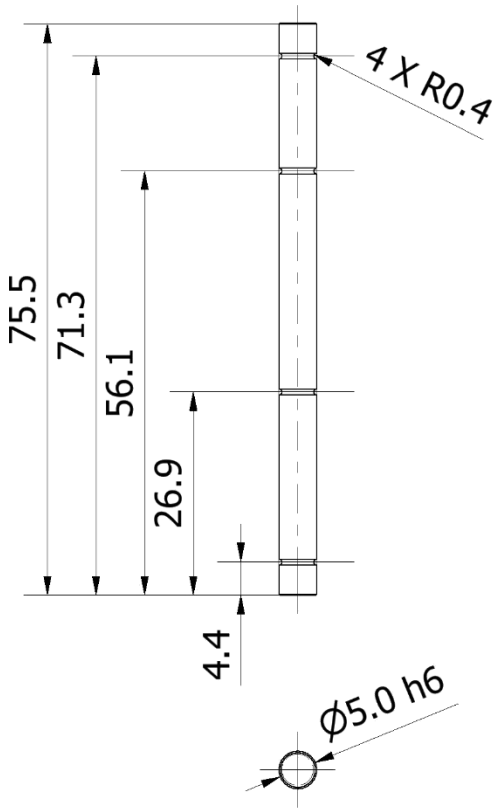
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Motor Gear Shaft		
	K009	Edition	Sheet 83 / 95



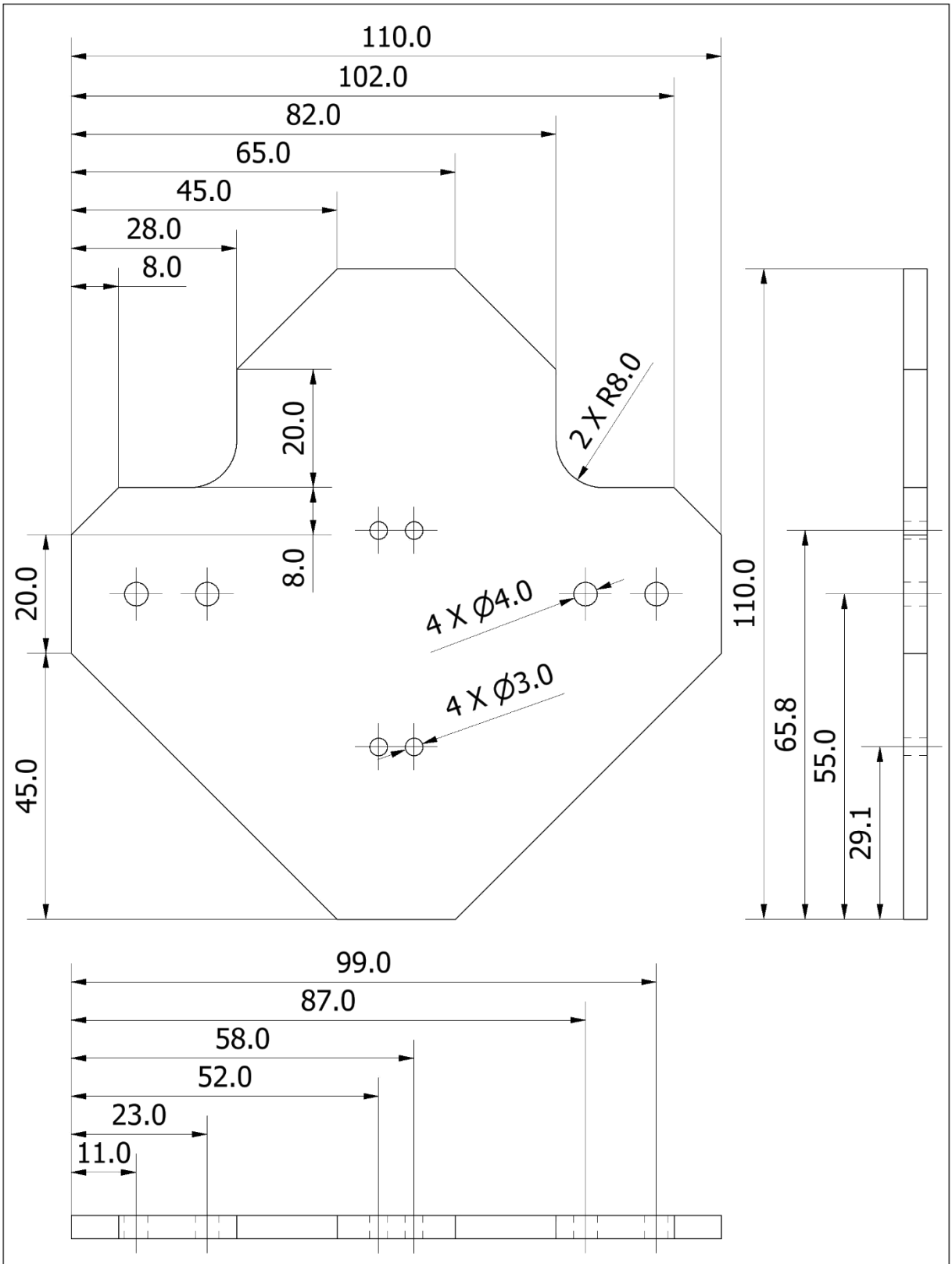
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Hand Auxiliary Gear Shaft		
	K010	Edition	Sheet 84 / 95



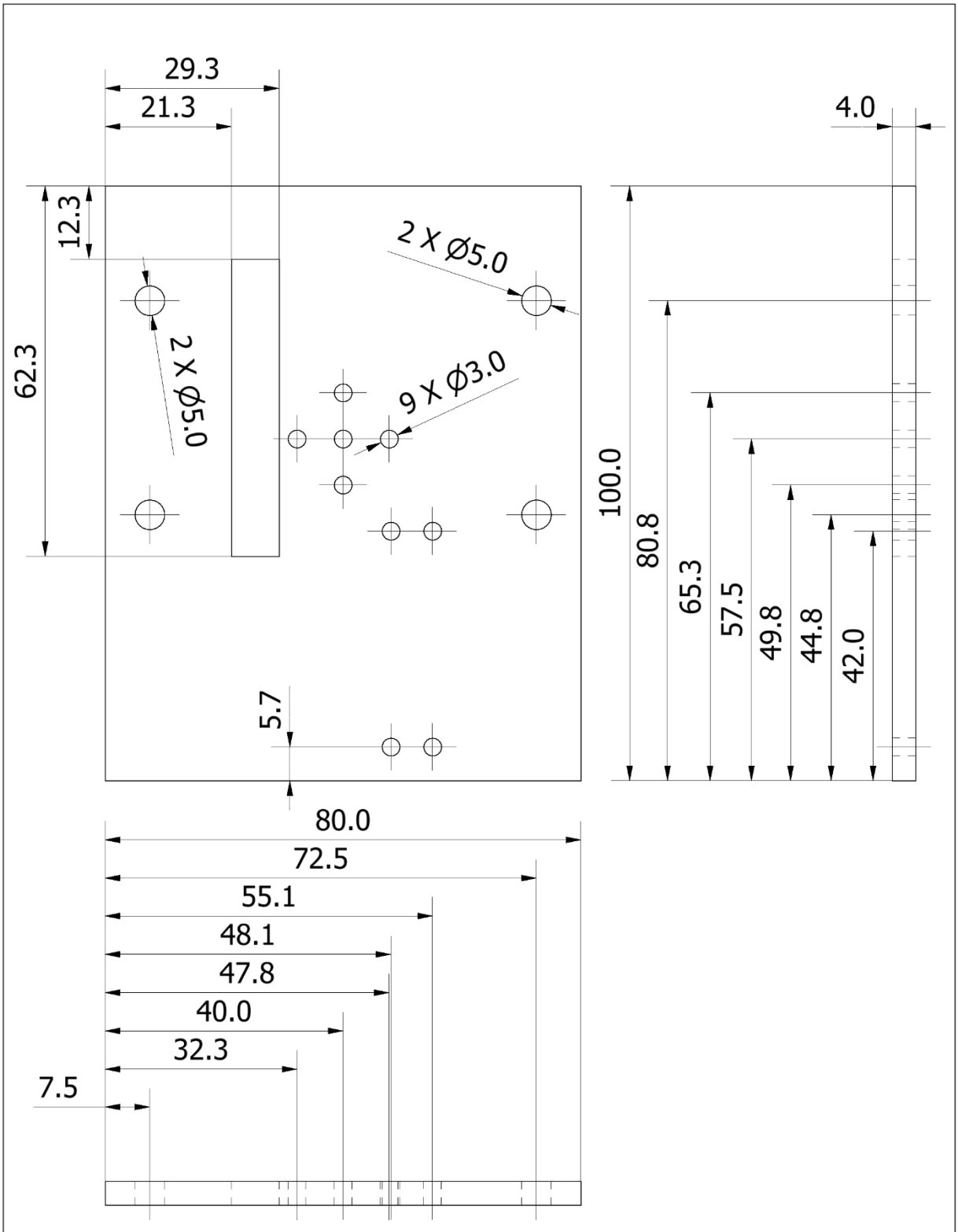
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Claw Input Shaft		
	K011	Edition	Sheet 85 / 95



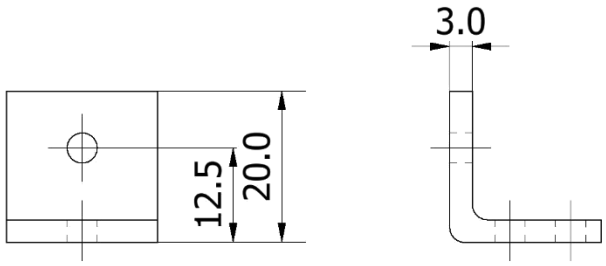
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

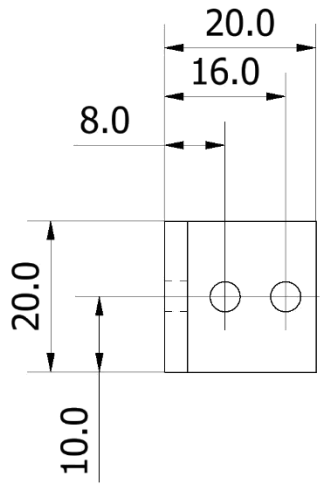
Universidade do Minho Laboratório de Automação e Robótica	Lower Cervical Spine Plate		
	L000	Edition	Sheet 86 / 95



Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

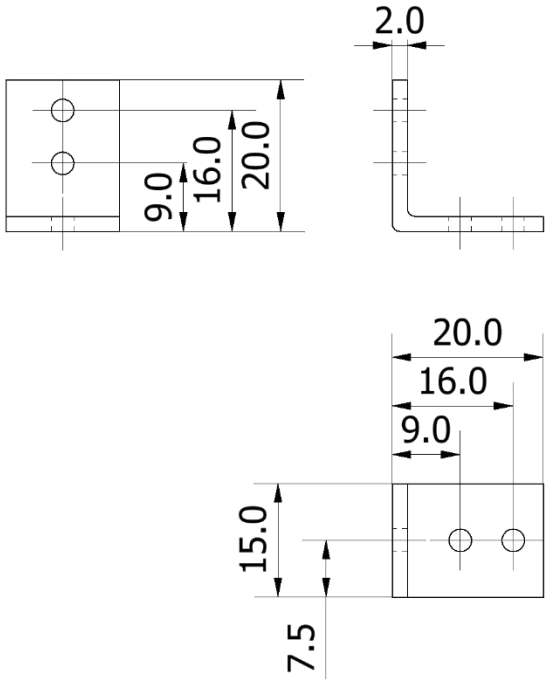
Universidade do Minho Laboratório de Automação e Robótica	Neck Plate		
	L010	Edition	Sheet 87 / 95





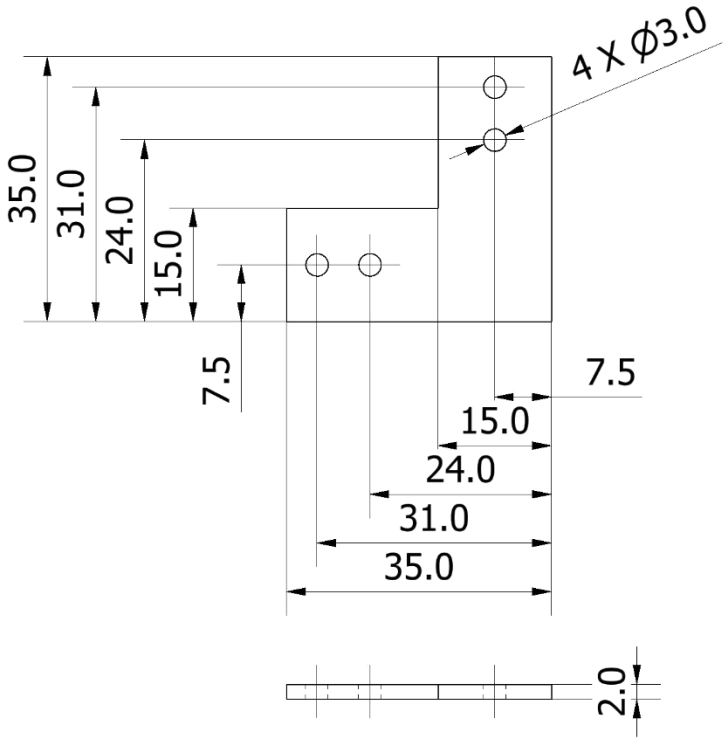
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Neck L Corner		
	L011	Edition	Sheet 88 / 95



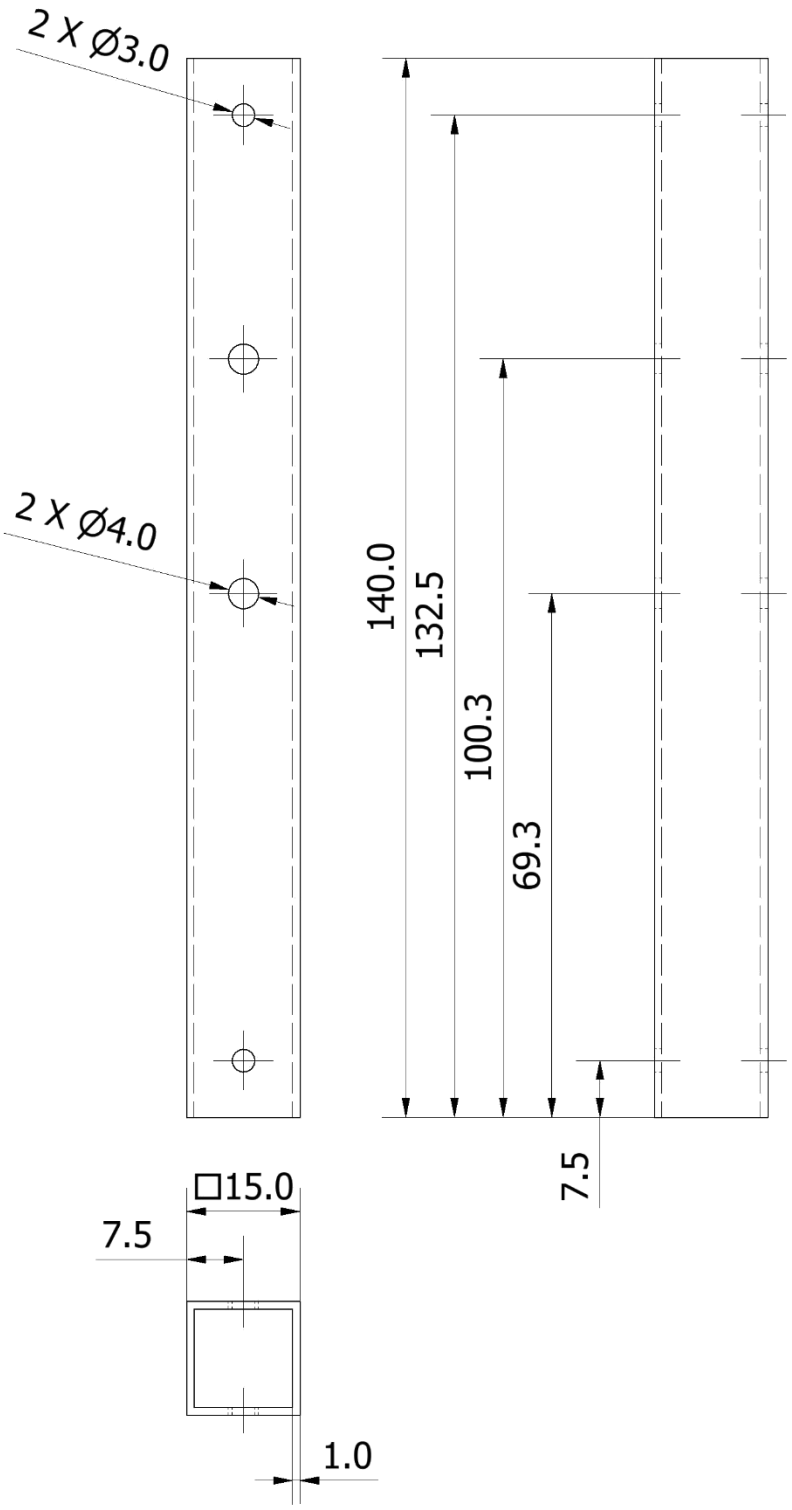
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Head L Corner		
	L020	Edition	Sheet 89 / 95



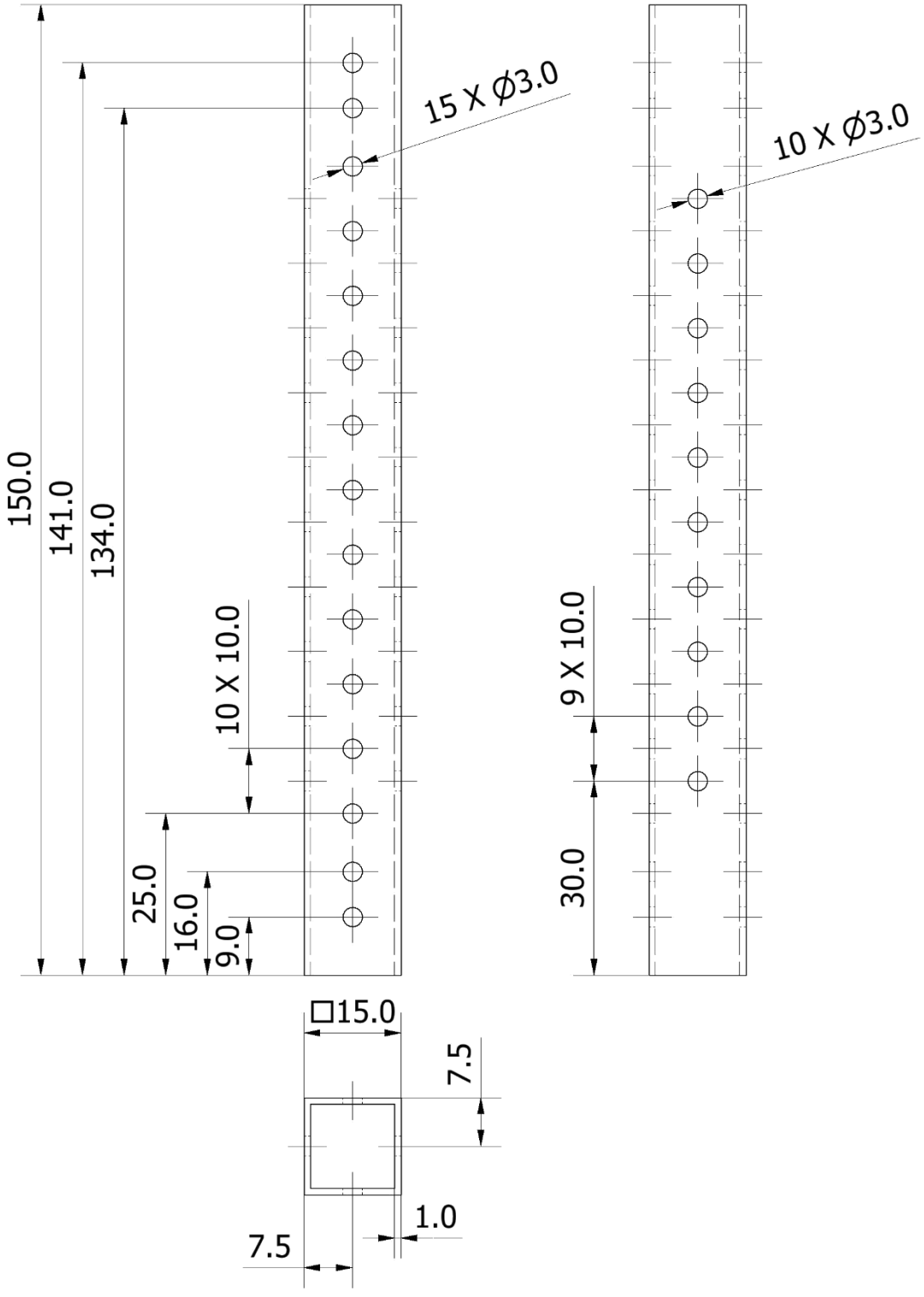
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Head L Plate		
	L021	Edition	Sheet 90 / 95



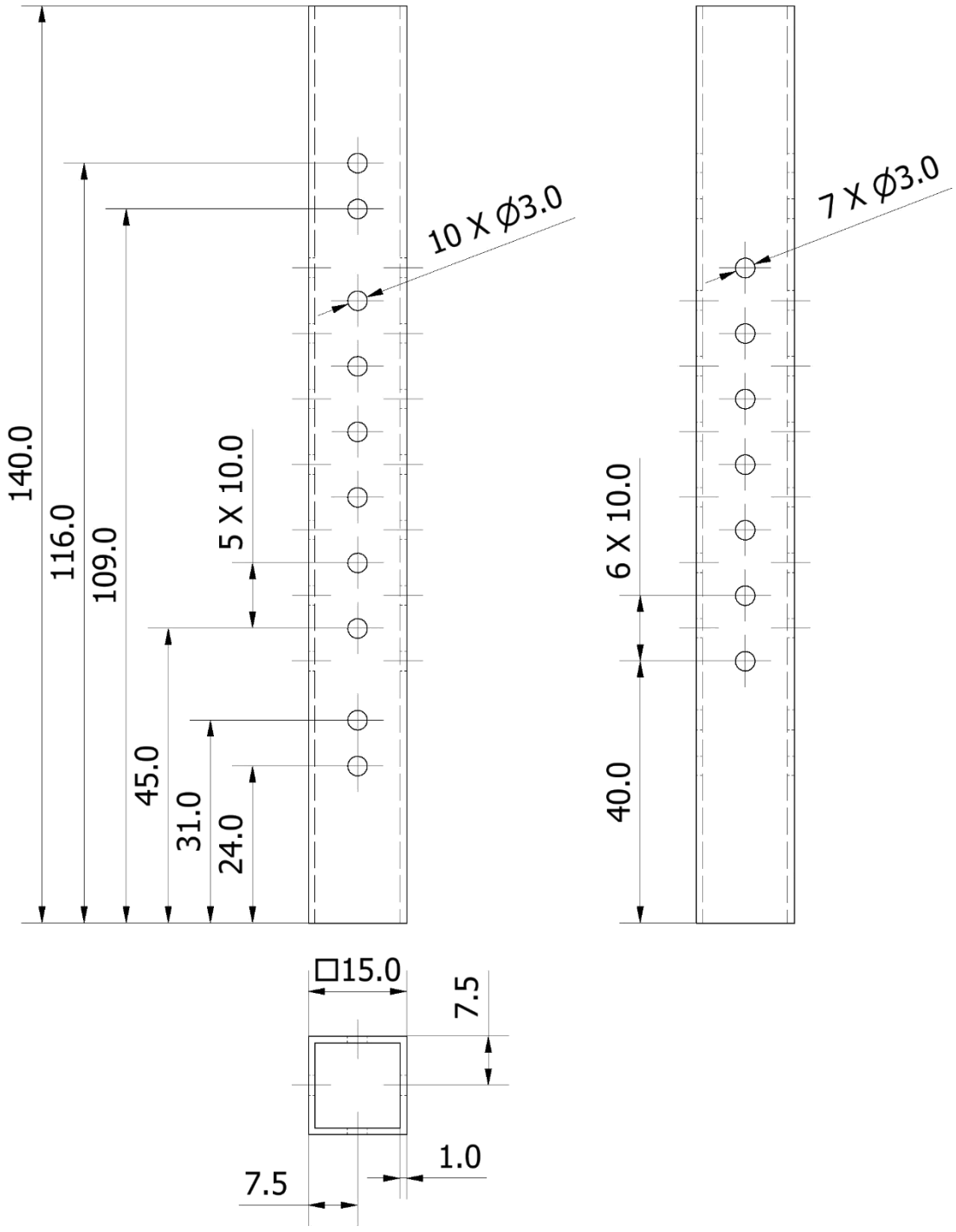
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Head Center Tube		
	L022	Edition	Sheet 91 / 95



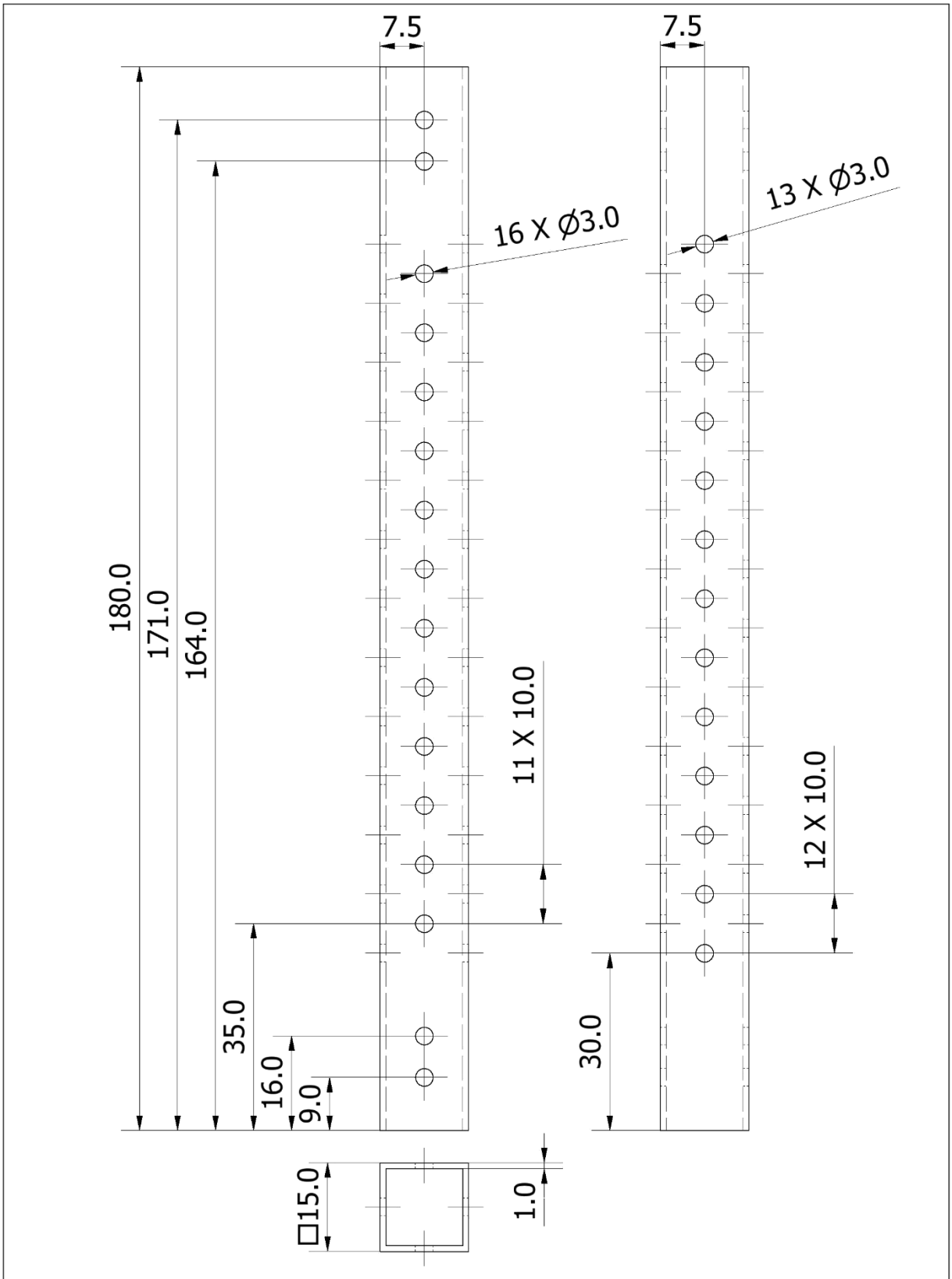
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 :1

Universidade do Minho Laboratório de Automação e Robótica	Head Lateral Tube		
	L023	Edition	Sheet 92 / 95



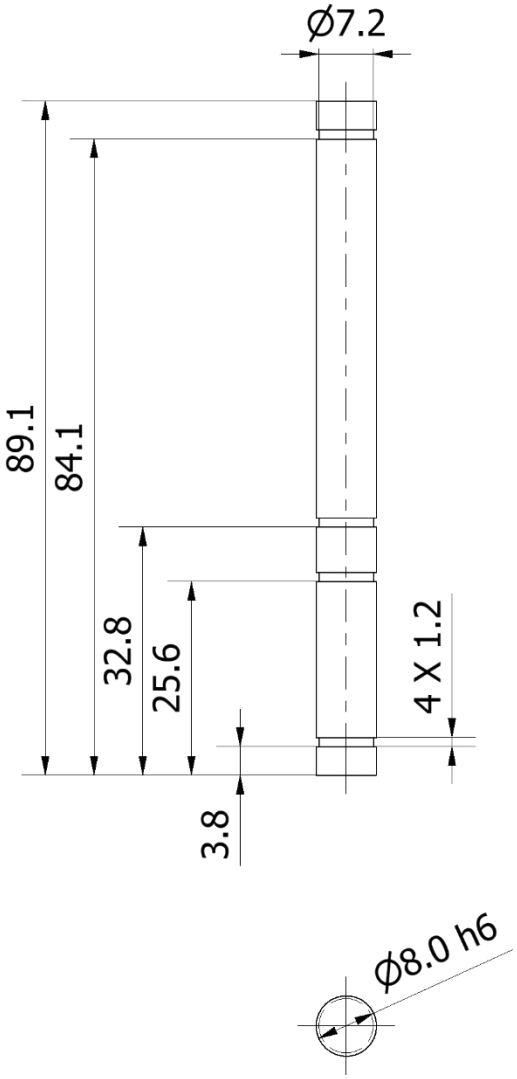
Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Head Front-Back Tube		
	L024	Edition	Sheet 93 / 95



Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

Universidade do Minho Laboratório de Automação e Robótica	Head Vertical Tube		
	L025	Edition	Sheet 94 / 95

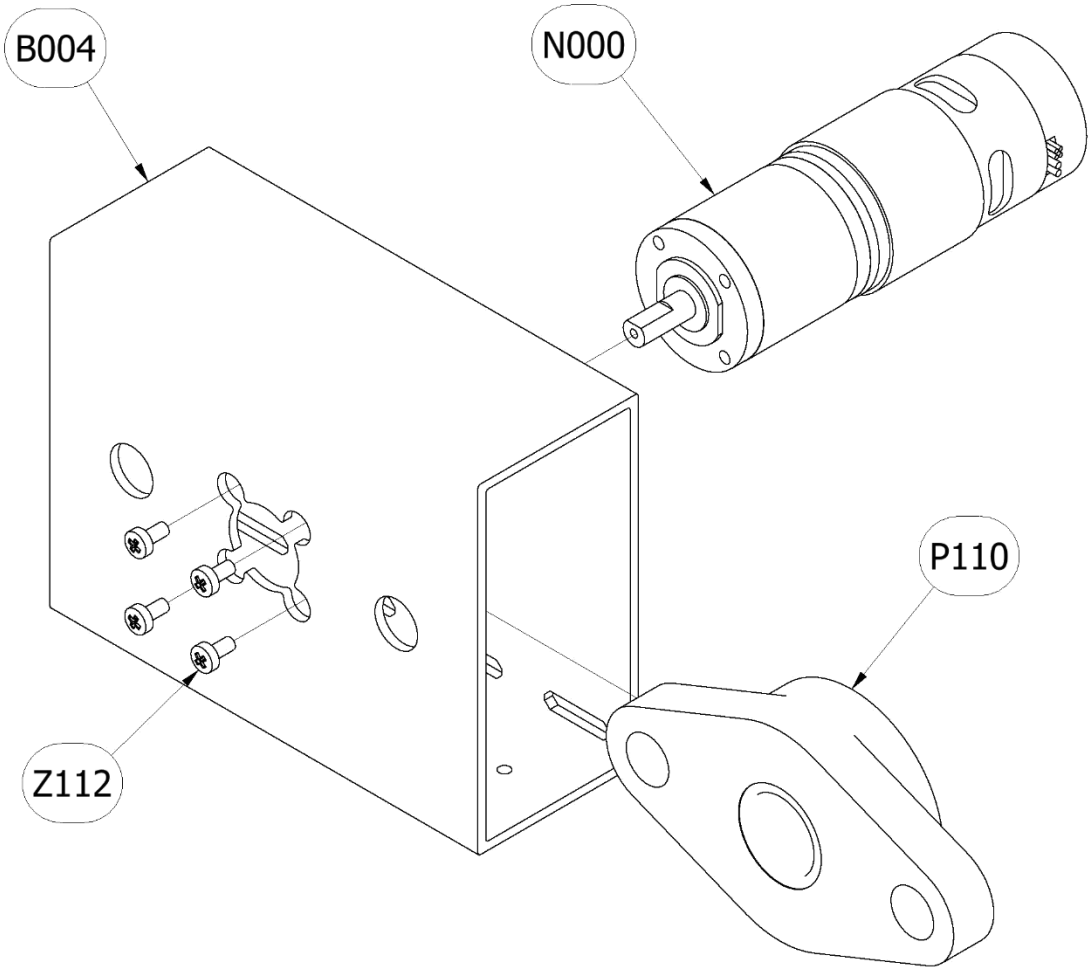


Designed by	Checked by	Approved by	Date	General Tolerance	Date	Scale
Fernando Gonçalves				ISO 2768 medium	30/04/2025	1 : 1

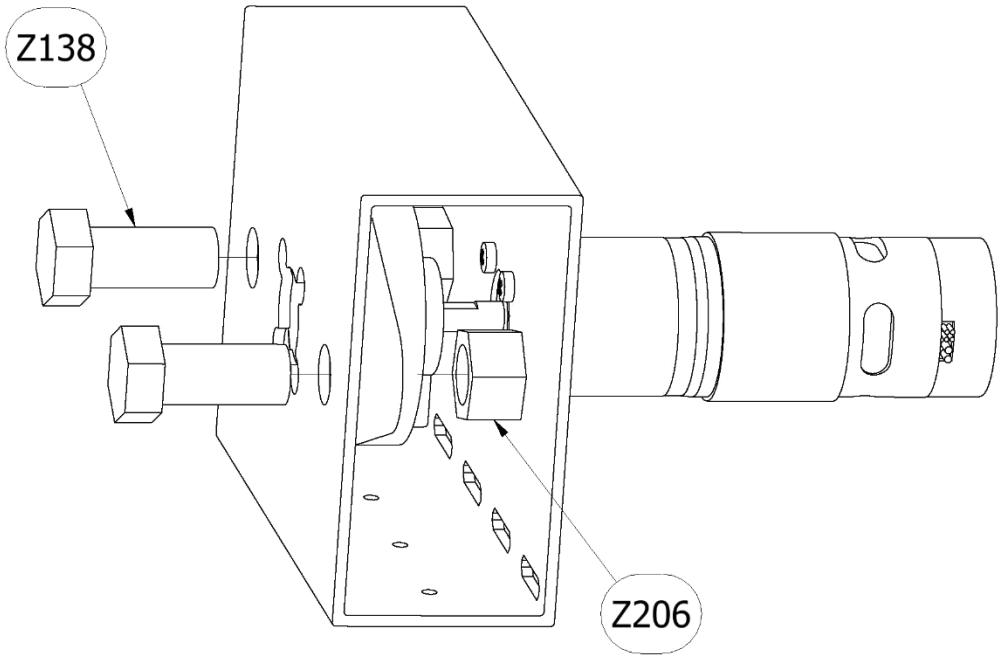
Universidade do Minho Laboratório de Automação e Robótica	Neck Motor Shaft		
	M000	Edition	Sheet 95 / 95

APPENDIX IV - ASSEMBLY GUIDE

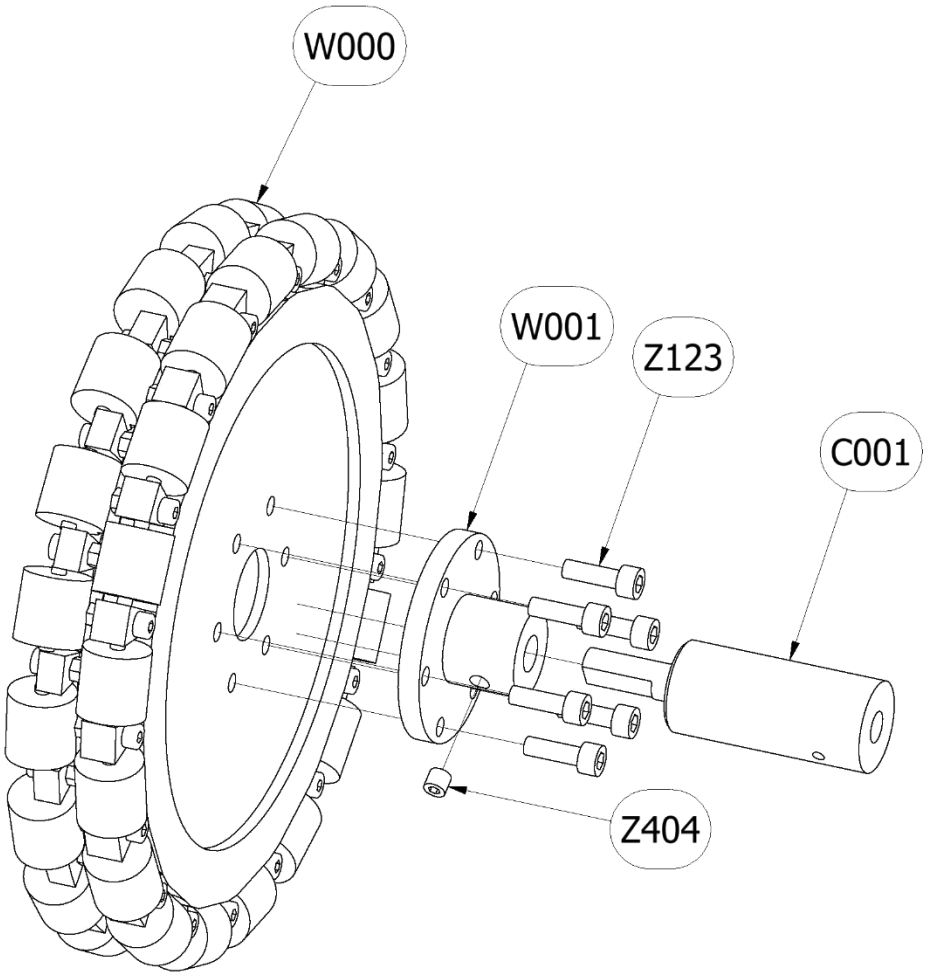
(Refer to the list of parts of the robot in Appendix II to identify components)



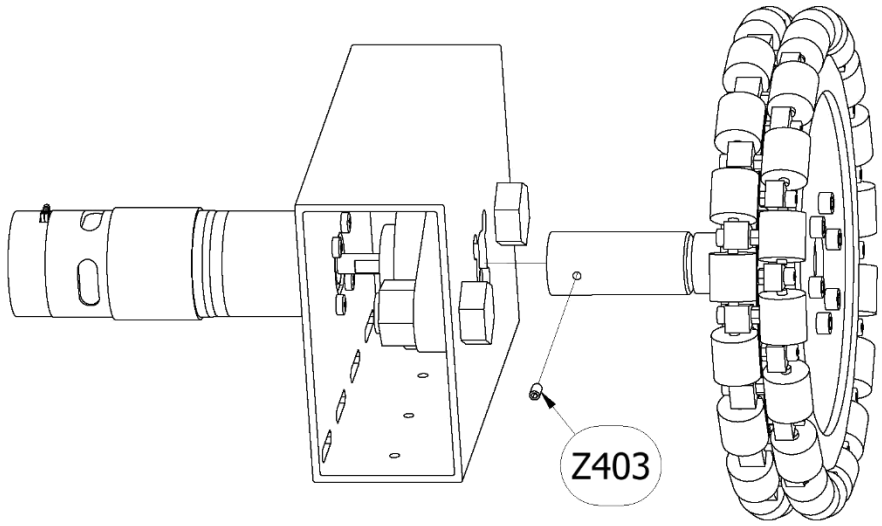
Sheet
1 / 62



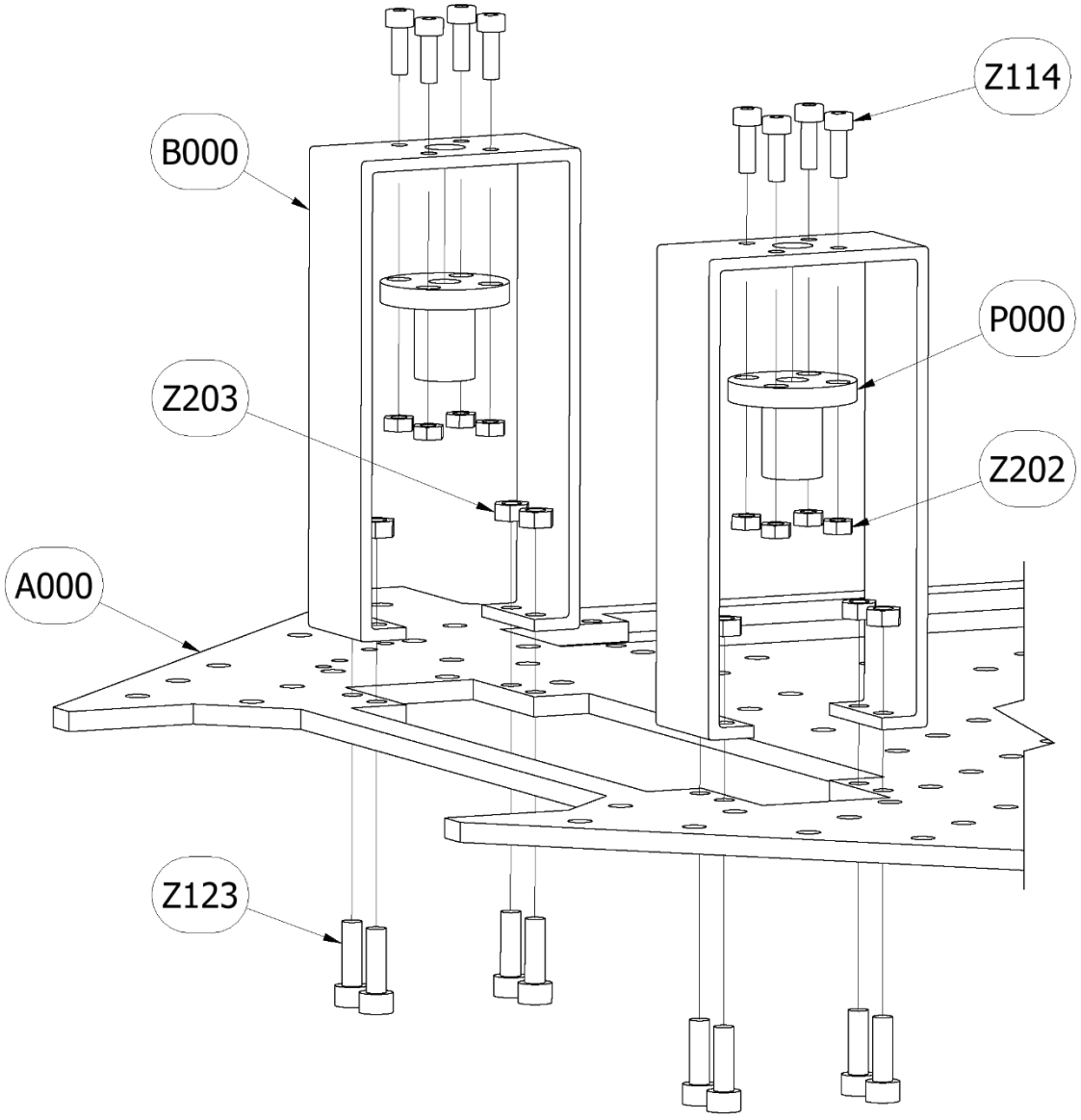
Sheet
2 / 62



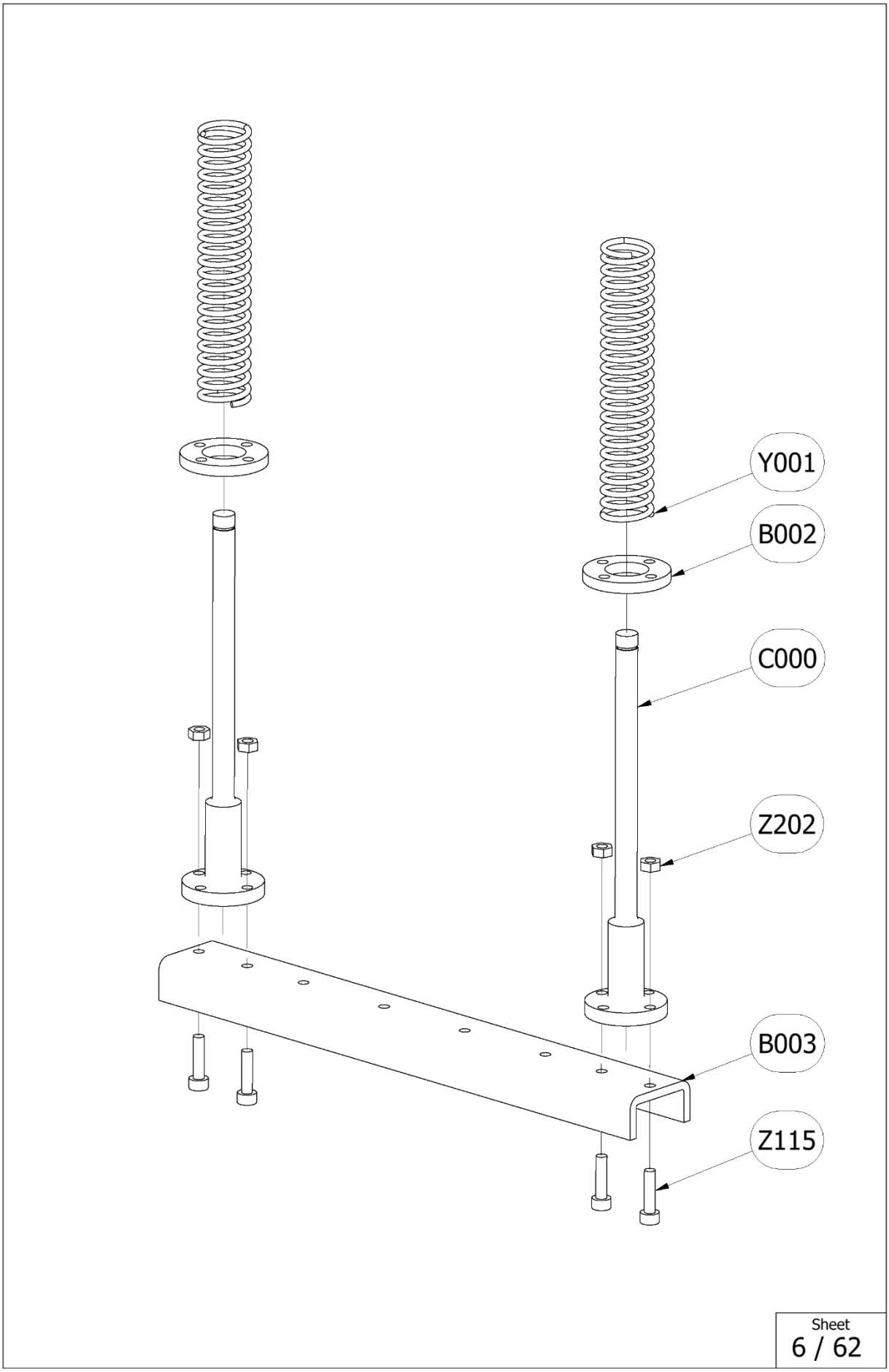
Sheet
3 / 62

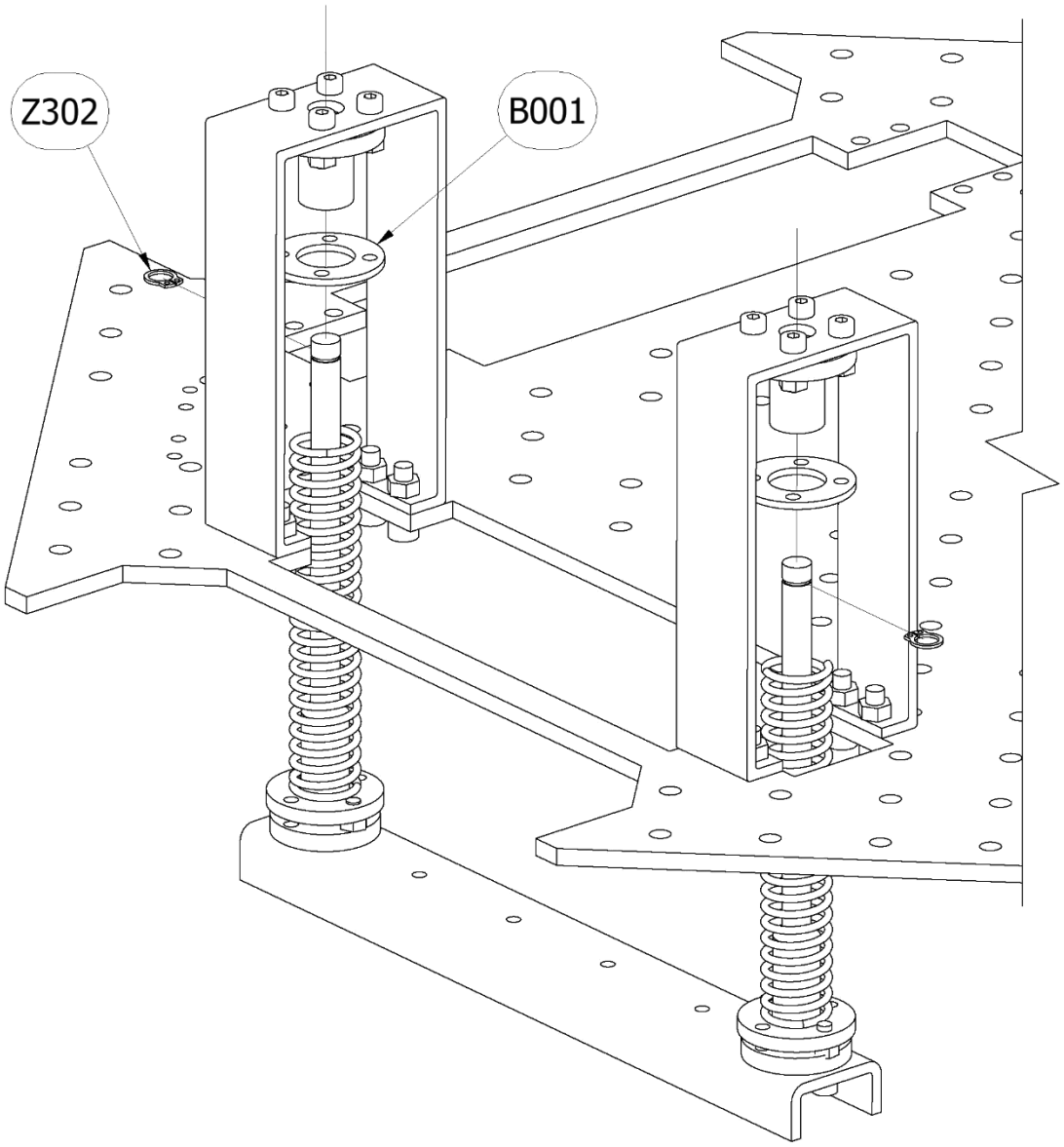


Sheet
4 / 62

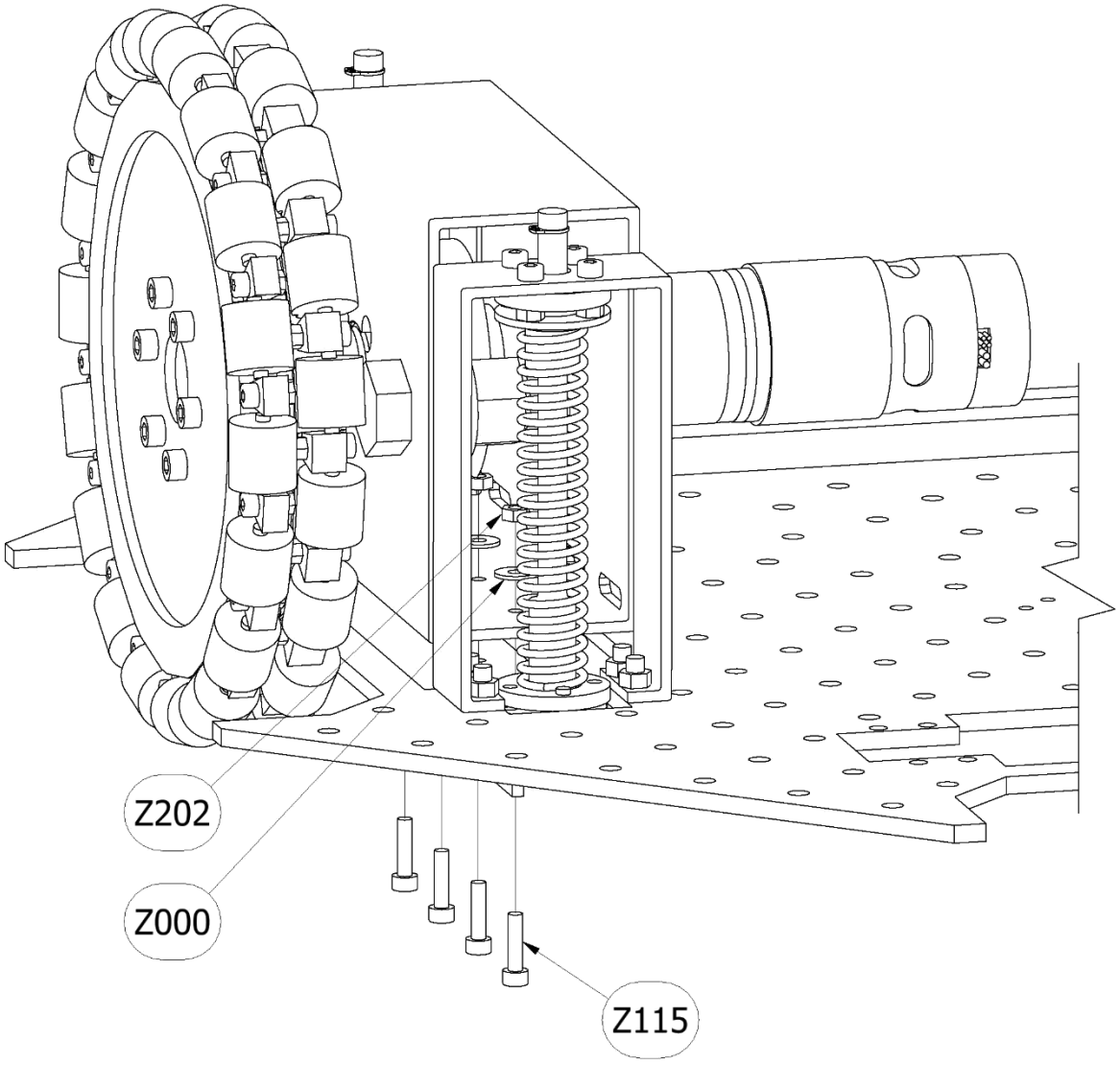


Sheet
5 / 62

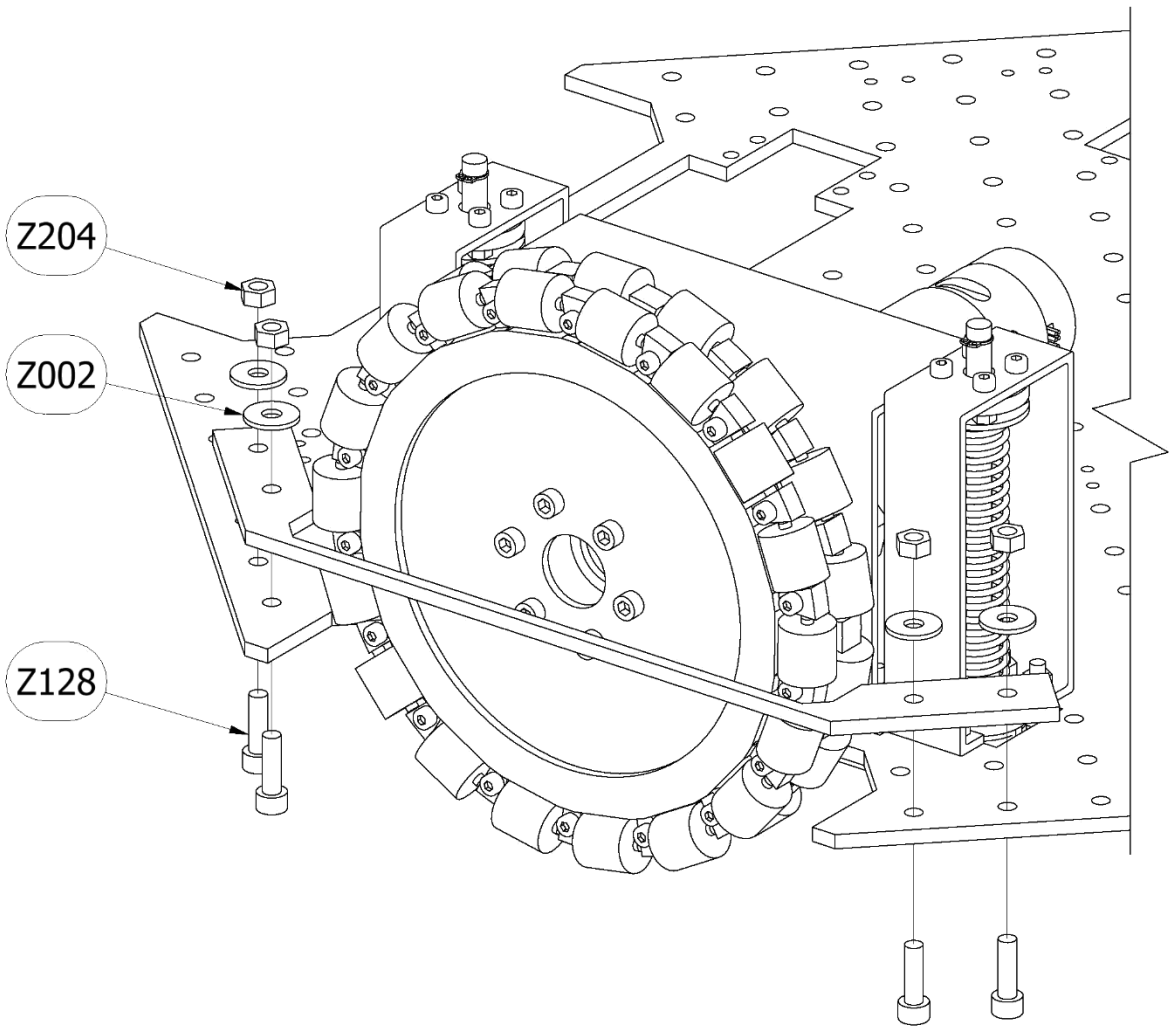

Y001
B002
C000
Z202
B003
Z115
Sheet
6 / 62



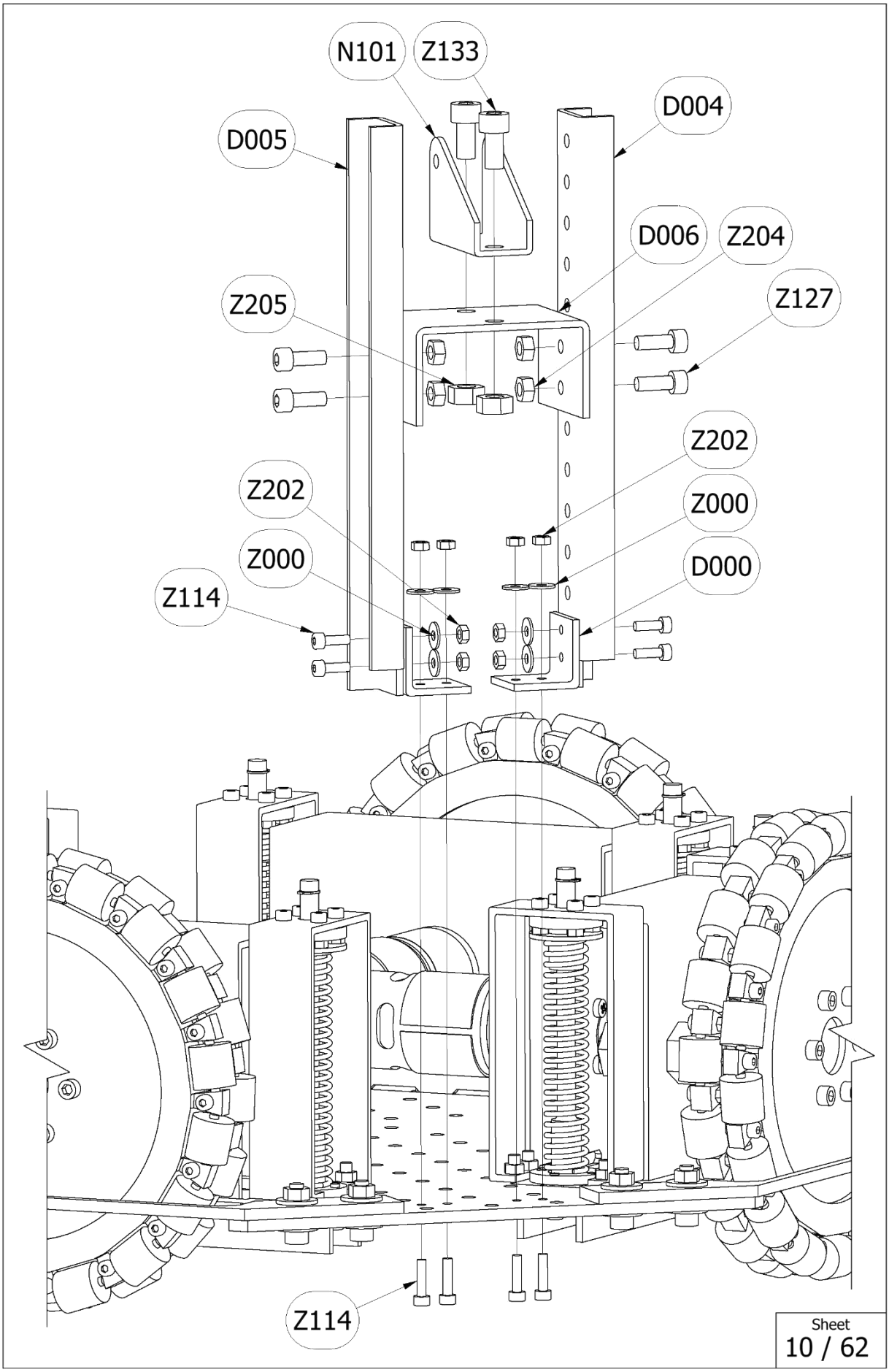
Sheet
7 / 62

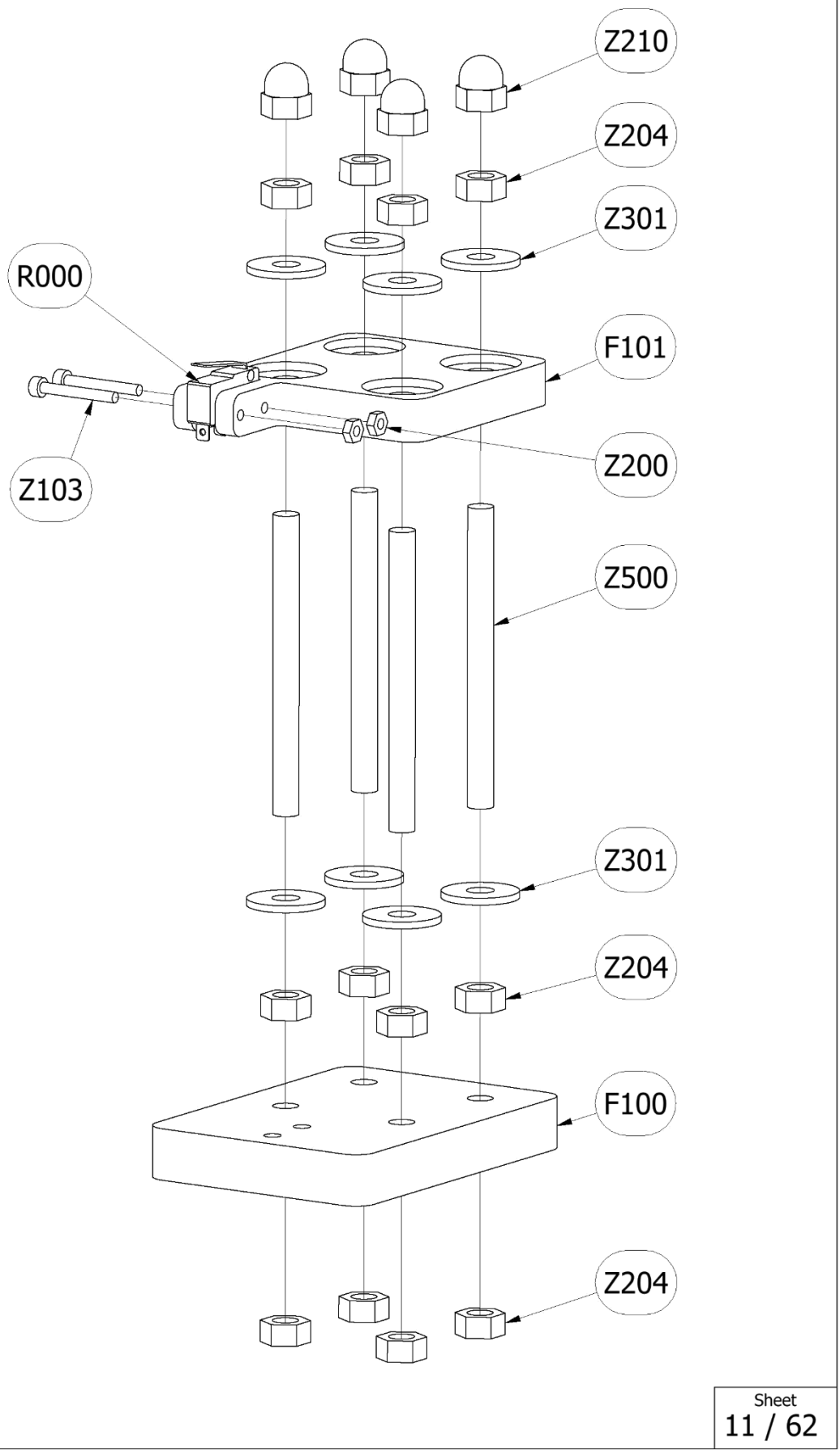


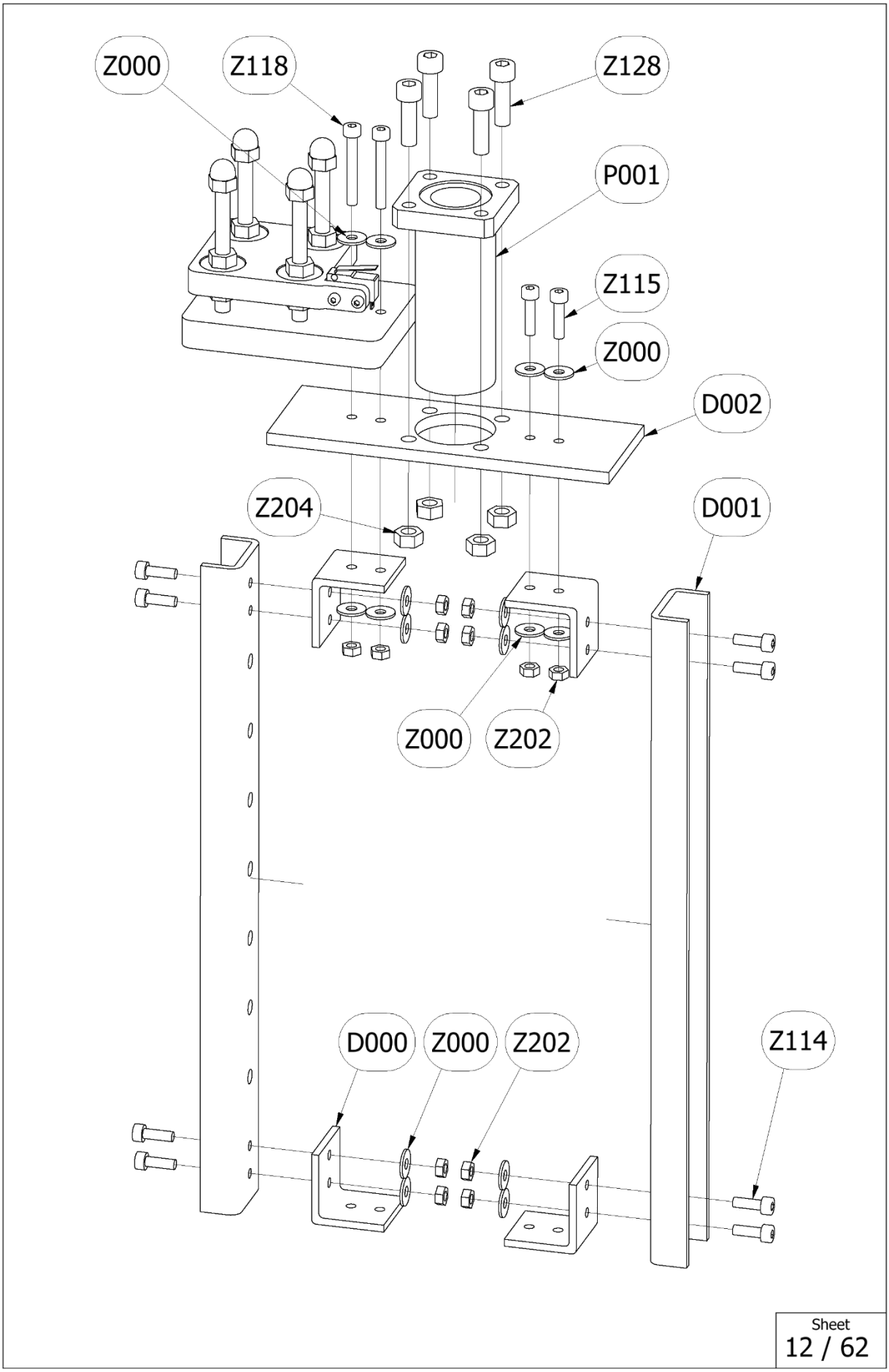
Sheet
8 / 62

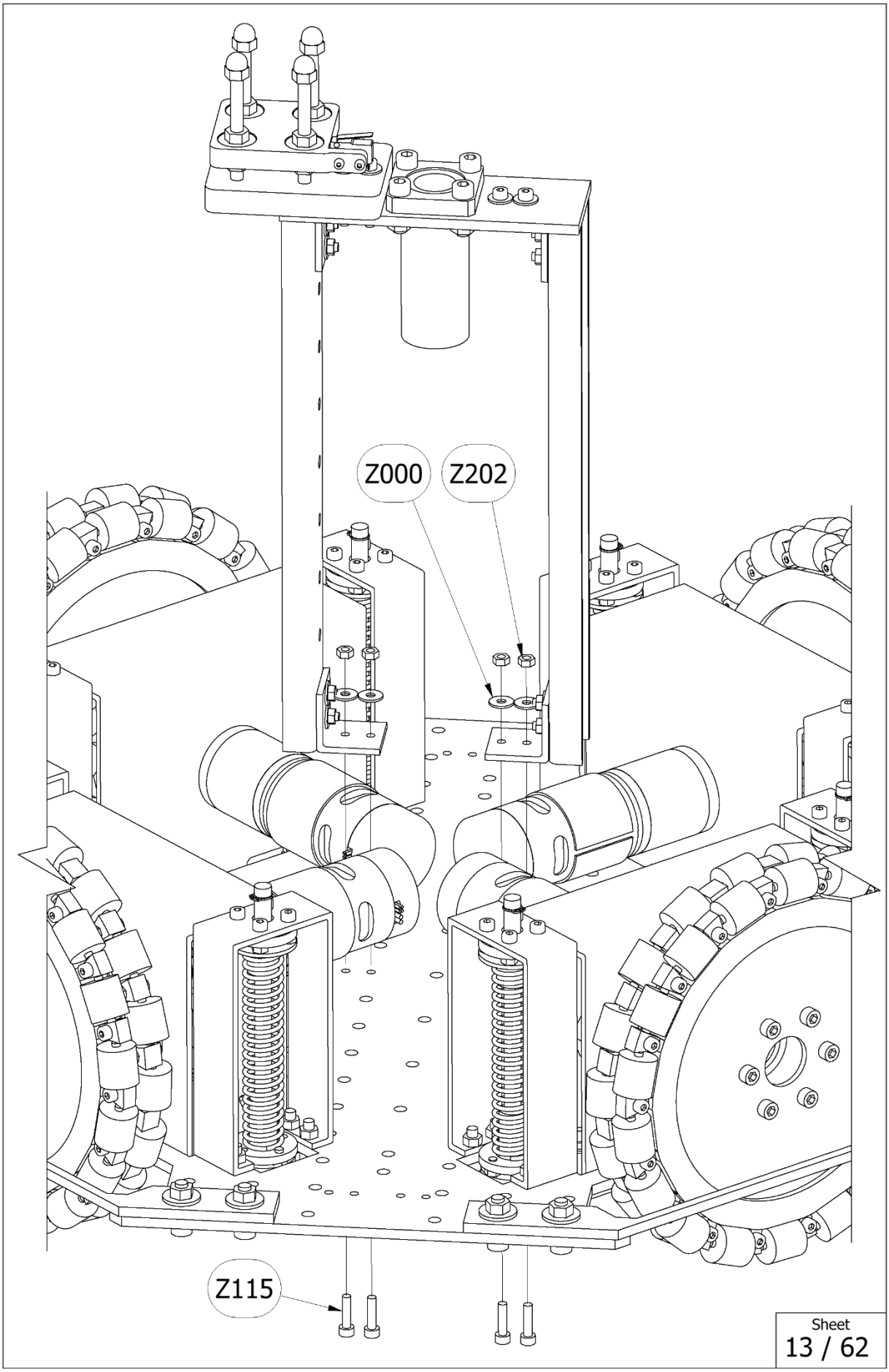


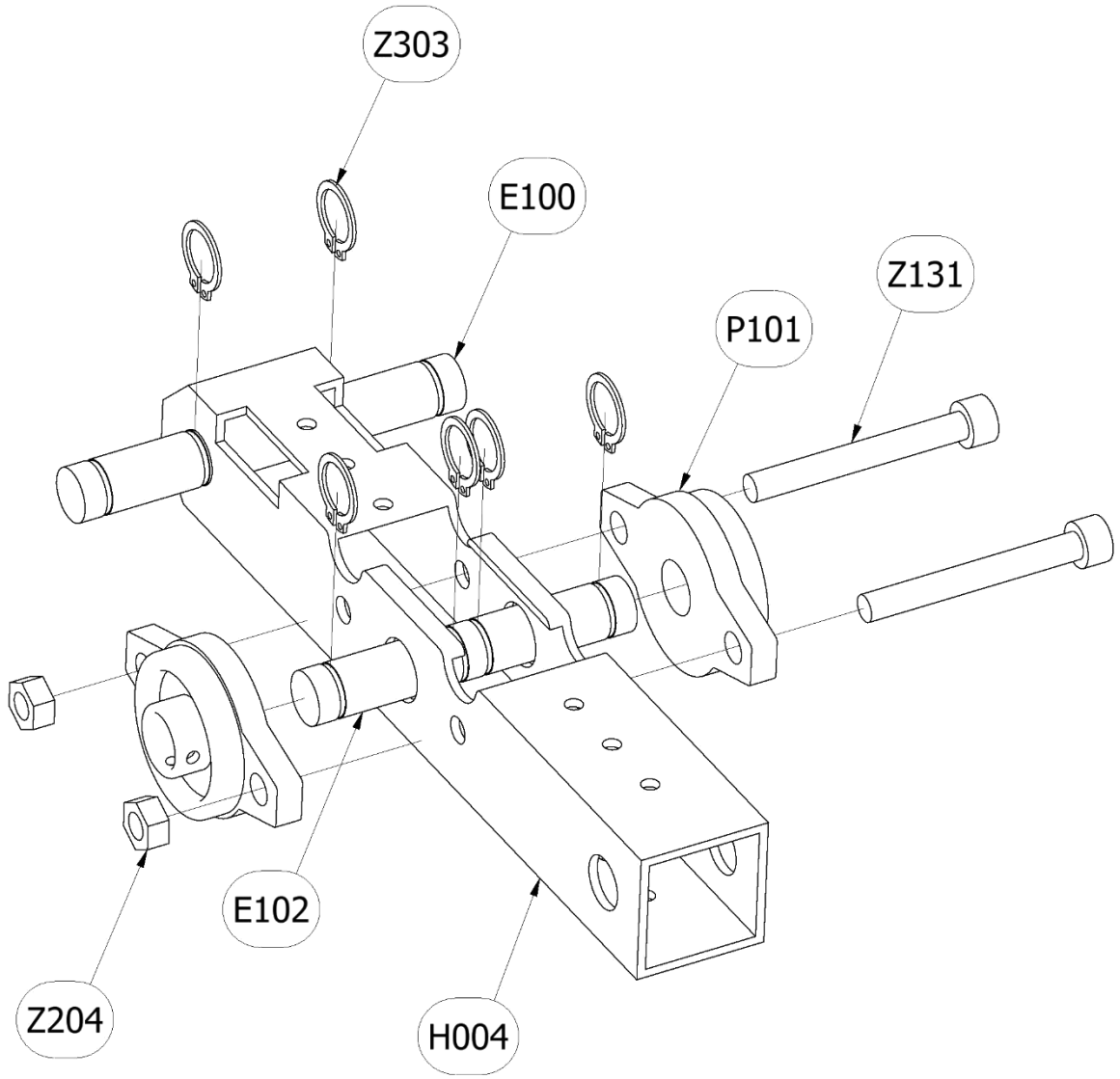
Sheet
9 / 62


N101
Z133
D004
D005
D006
Z204
Z205
Z127
Z202
Z202
Z000
Z000
D000
Z114
Z114
Sheet
10 / 62

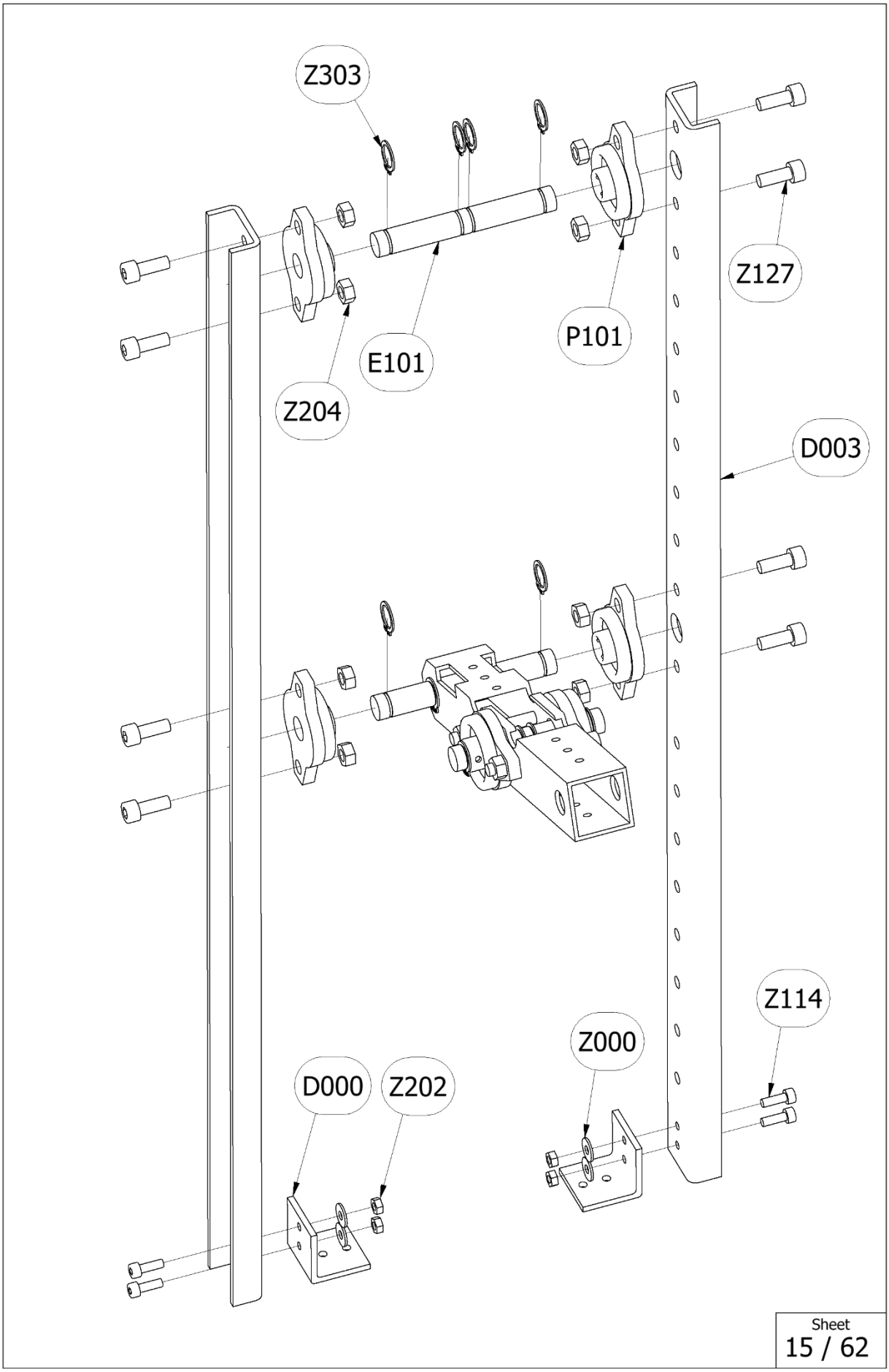

Z210
Z204
Z301
R000
F101
Z200
Z103
Z500
Z301
Z204
F100
Z204
Sheet
11 / 62

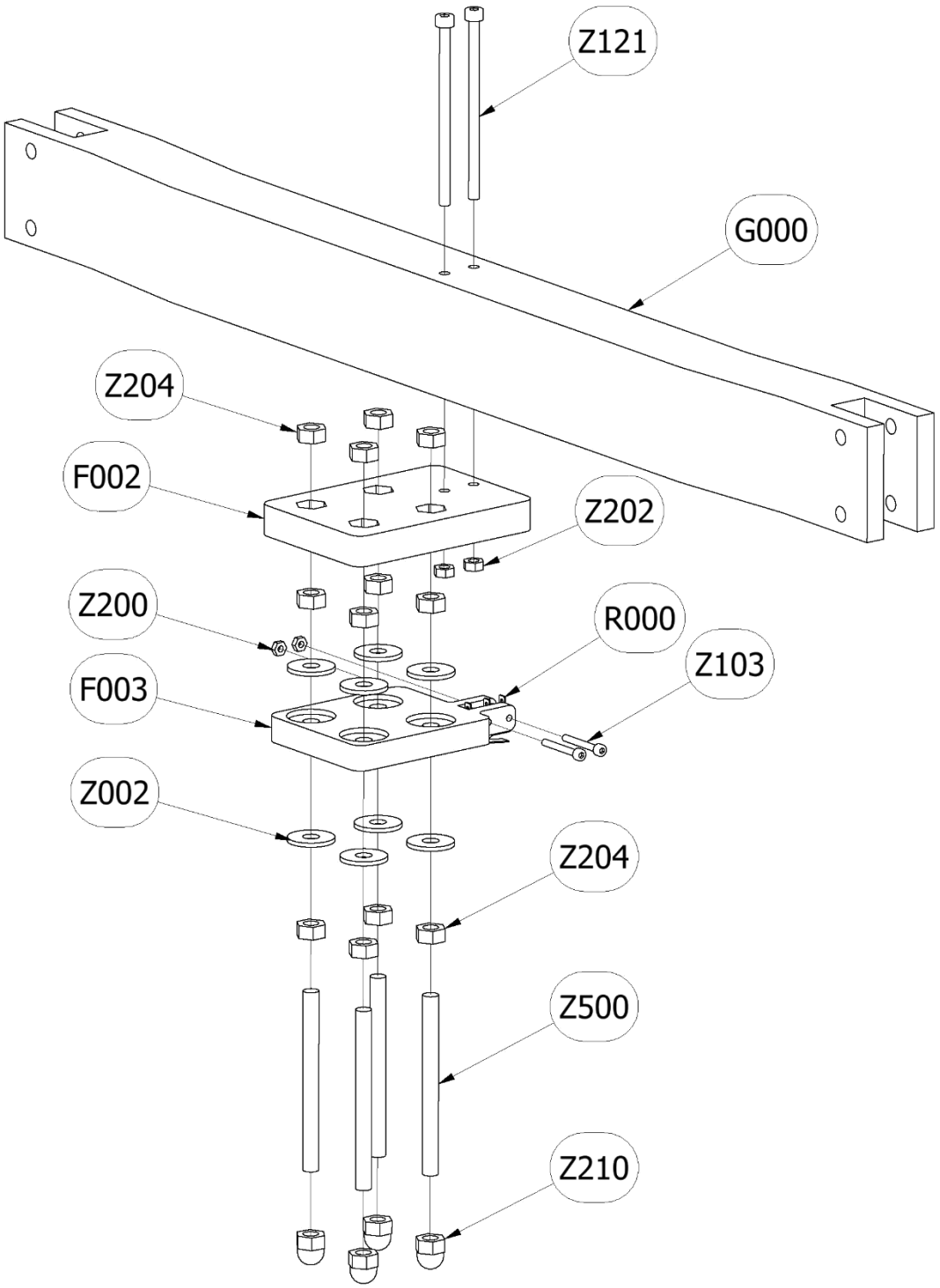

Z000
Z118
Z128
P001
Z115
Z000
D002
Z204
D001
Z000
Z202
D000
Z000
Z202
Z114
Sheet
12 / 62


Z000
Z202
Z115
Sheet
13 / 62

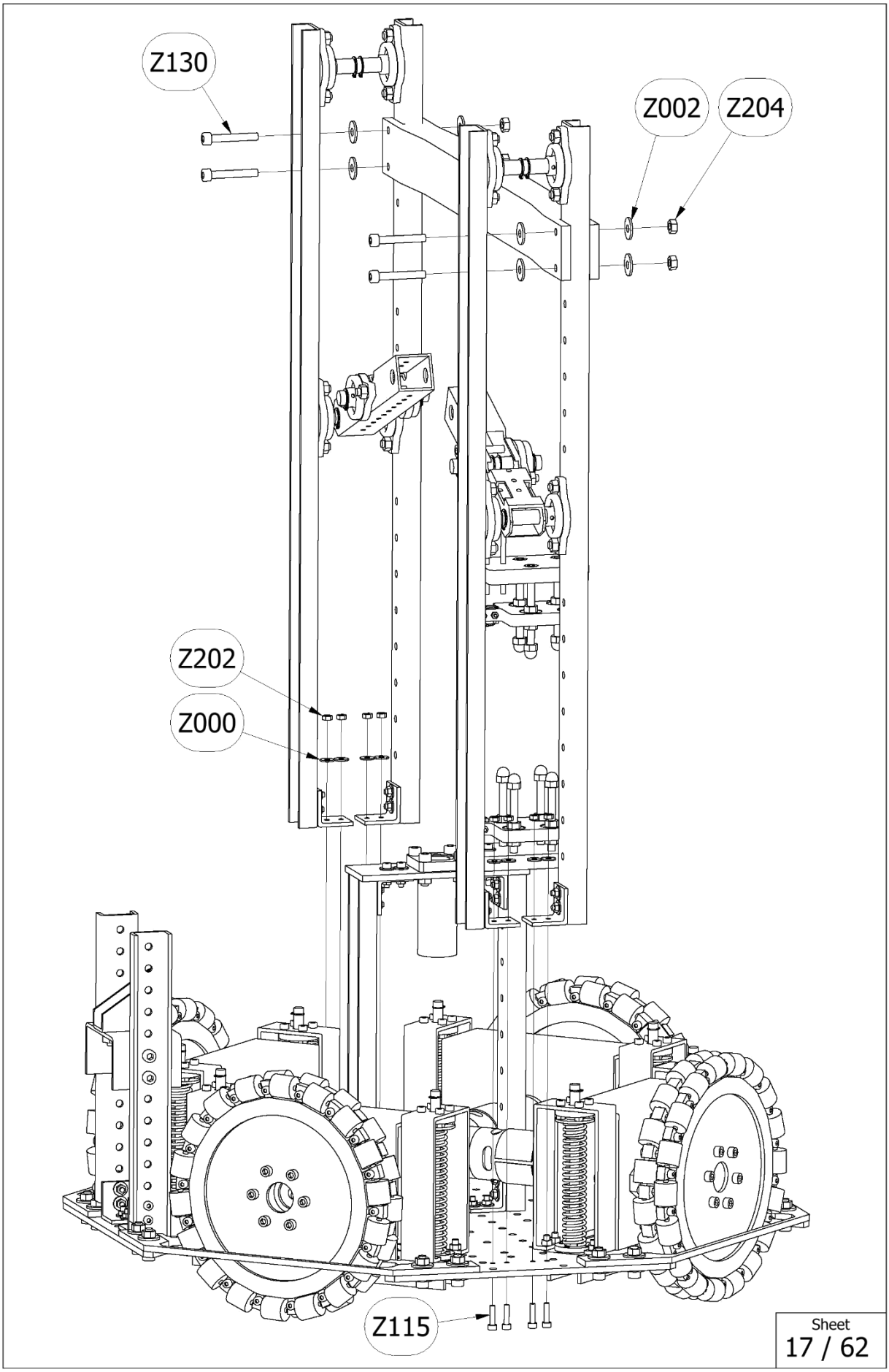


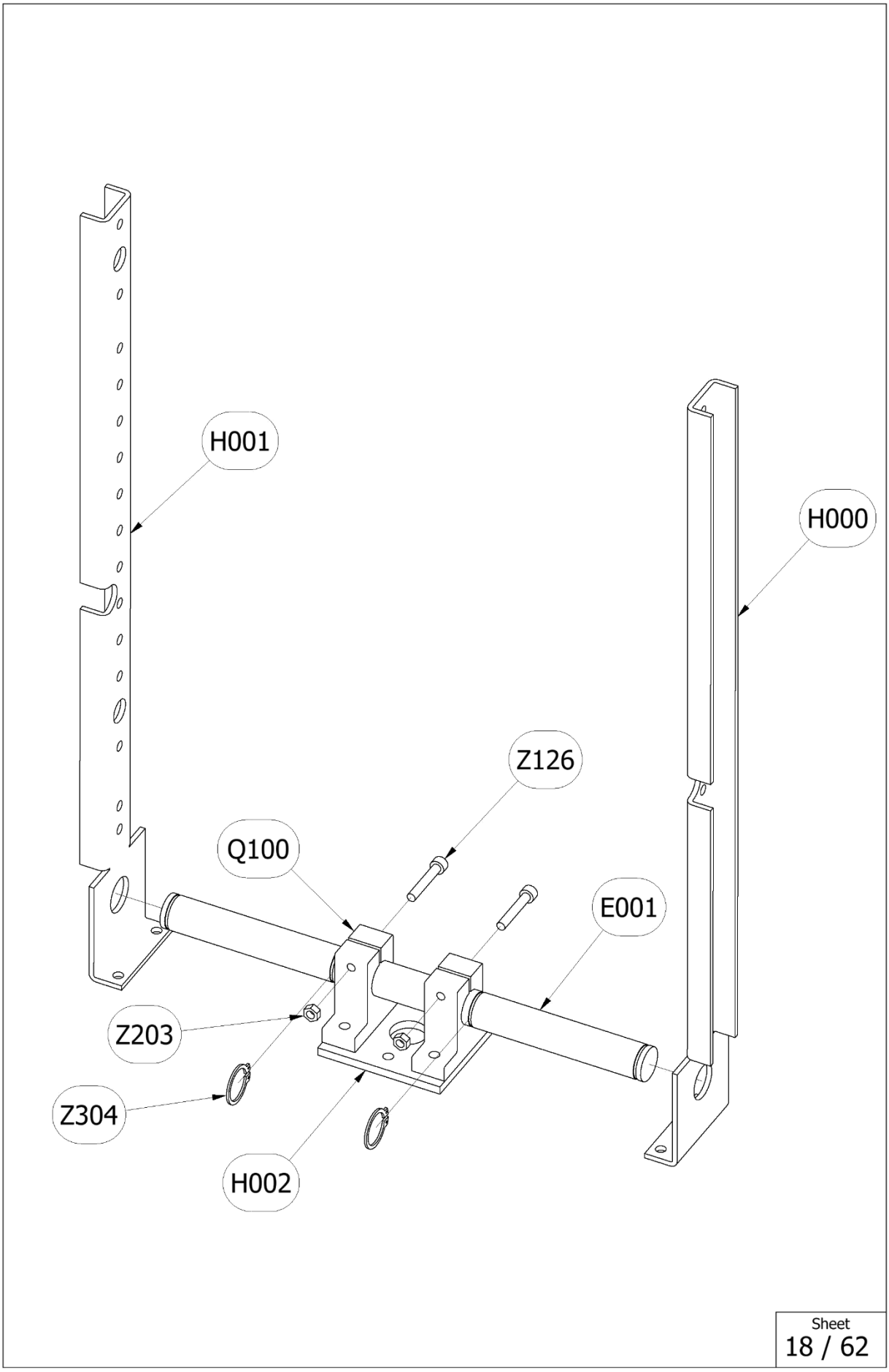
Sheet
14 / 62

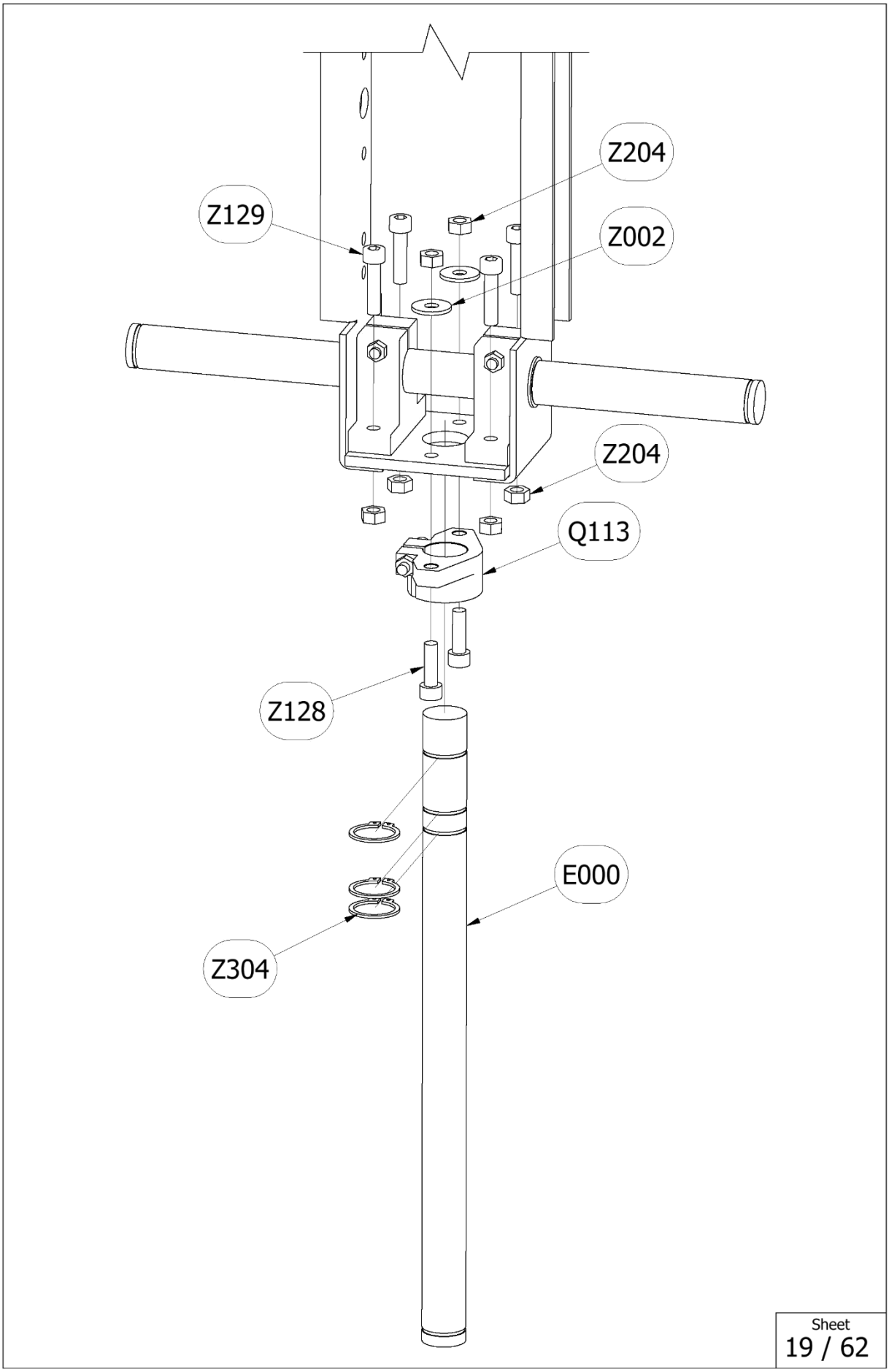

Z303
Z127
P101
E101
Z204
D003
Z114
Z000
D000
Z202
Sheet
15 / 62

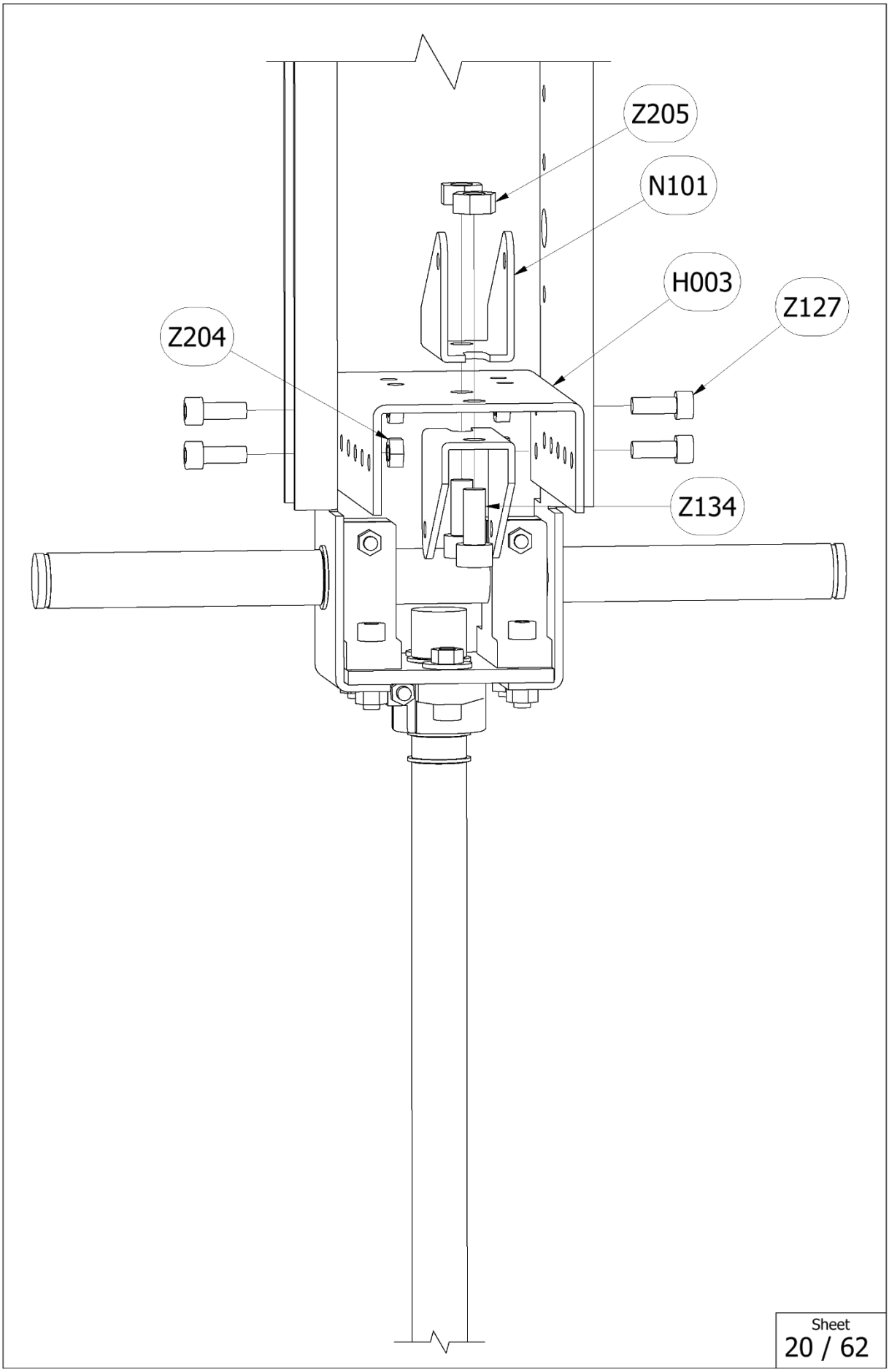

Z121
G000
Z204
F002
Z202
Z200
R000
Z103
F003
Z002
Z204
Z500
Z210

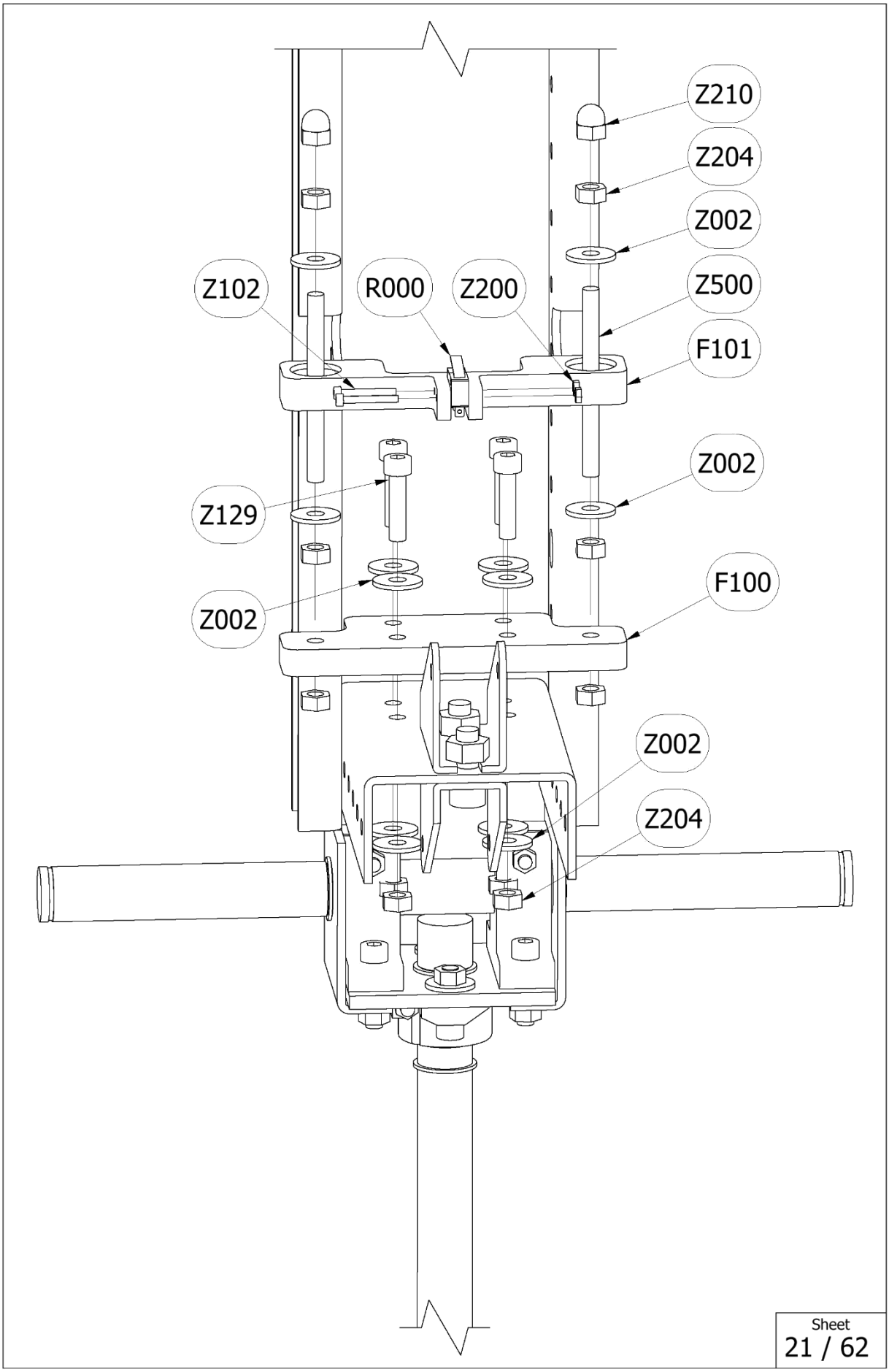
Sheet
16 / 62

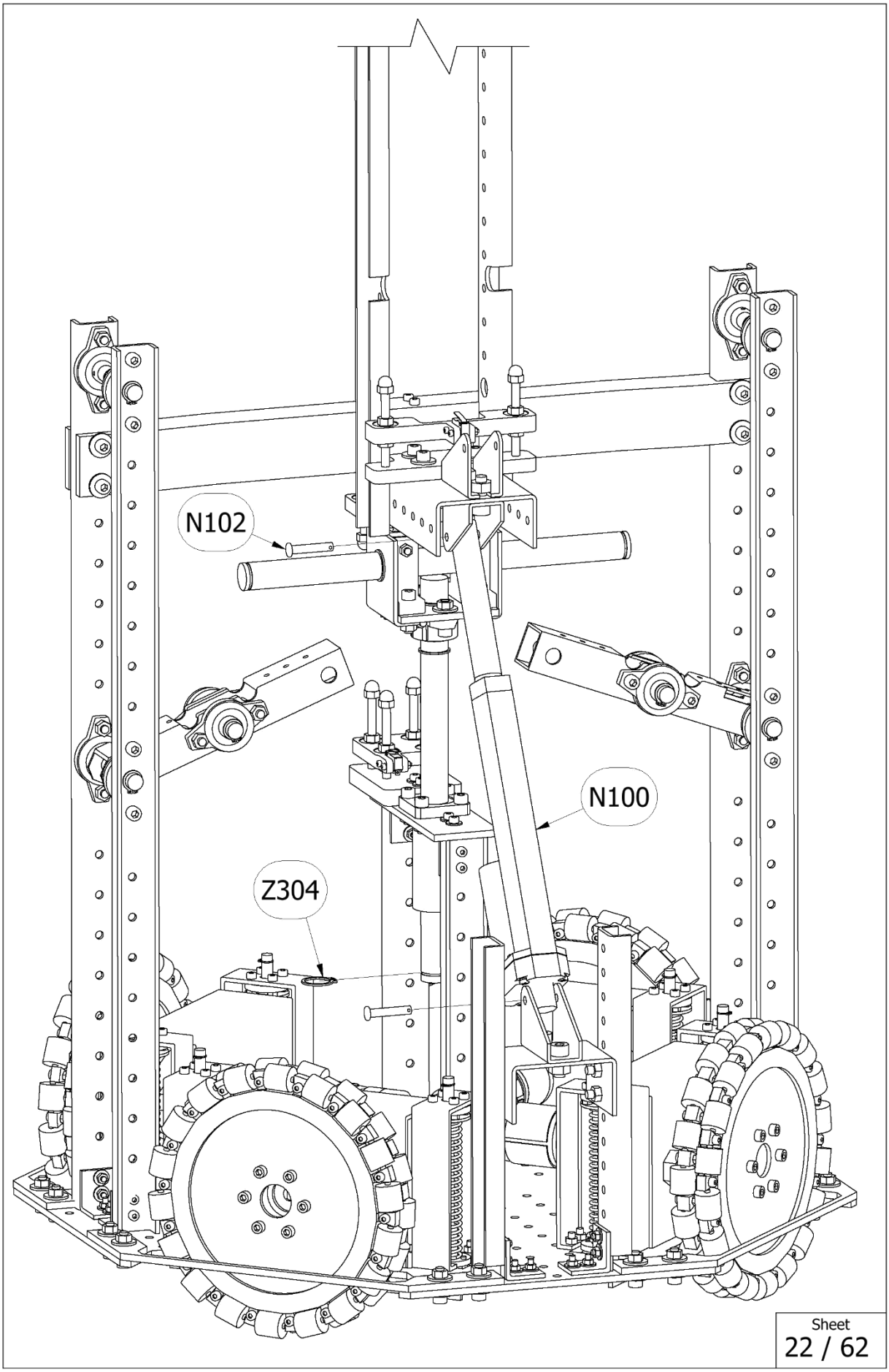

Z130
Z002
Z204
Z202
Z000
Z115
Sheet
17 / 62

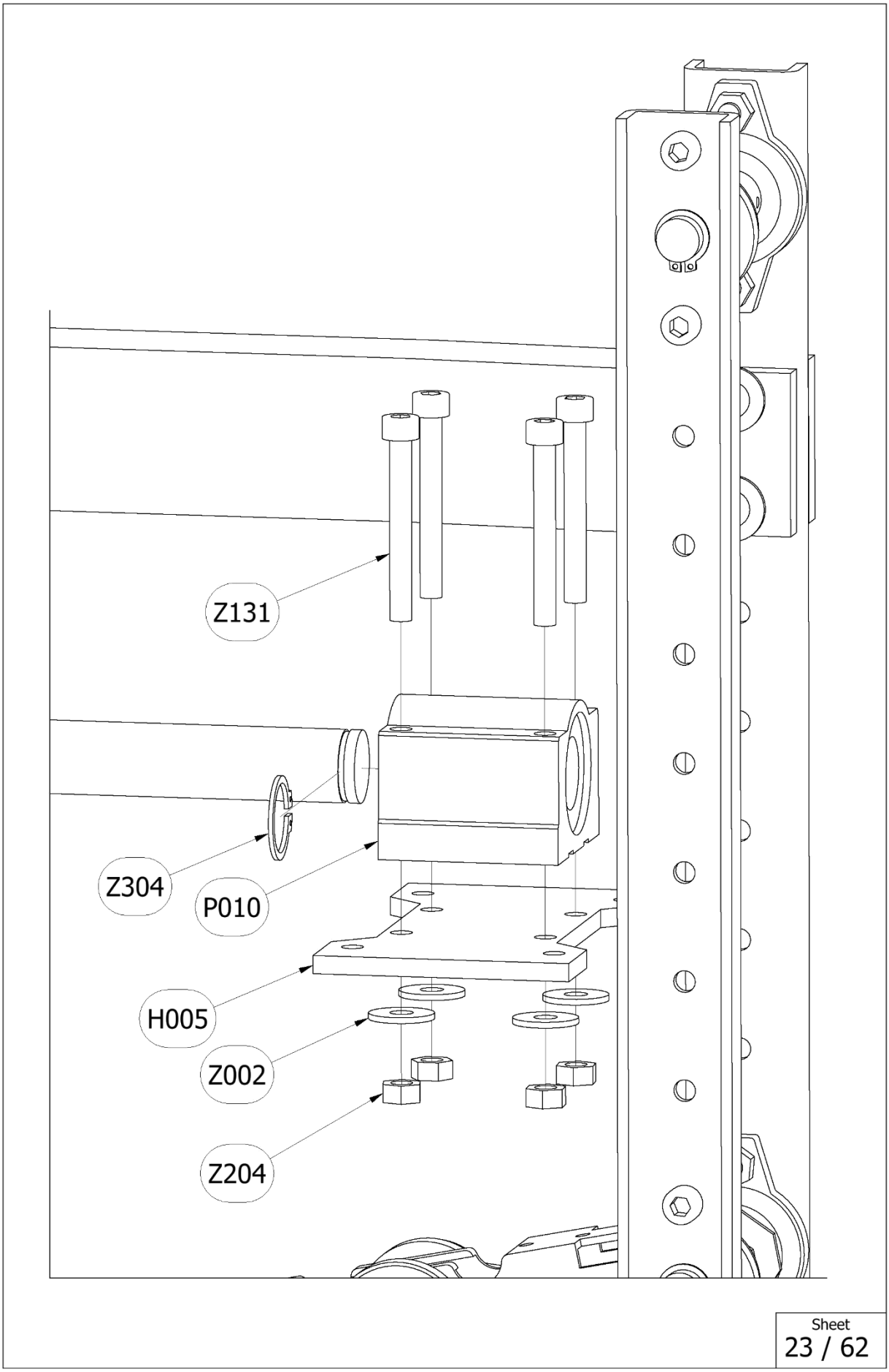

H001
H000
Z126
Q100
E001
Z203
Z304
H002
Sheet
18 / 62

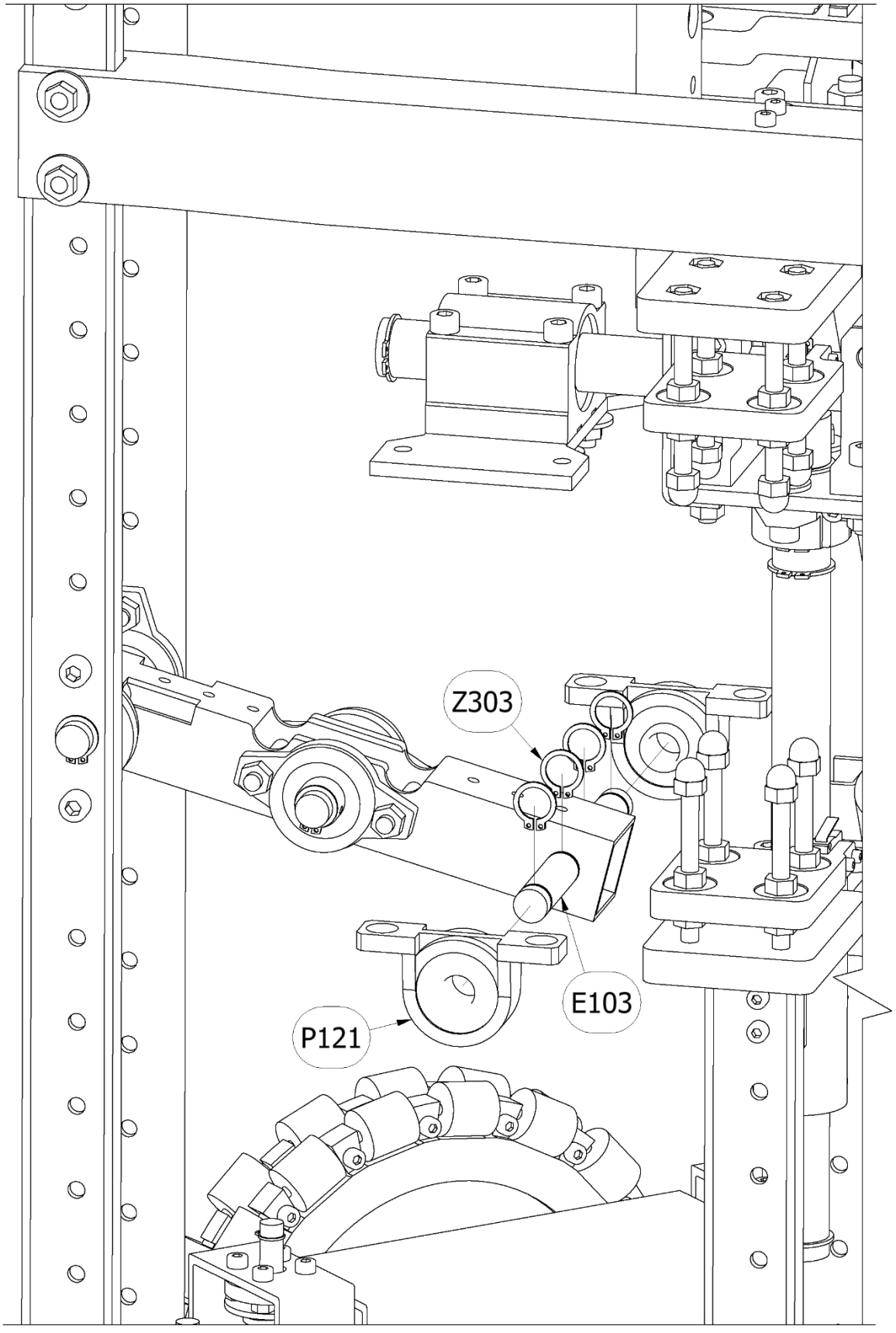

Z204
Z129
Z002
Z204
Q113
Z128
E000
Z304
Sheet
19 / 62


Z205
N101
H003
Z127
Z204
Z134
Sheet
20 / 62

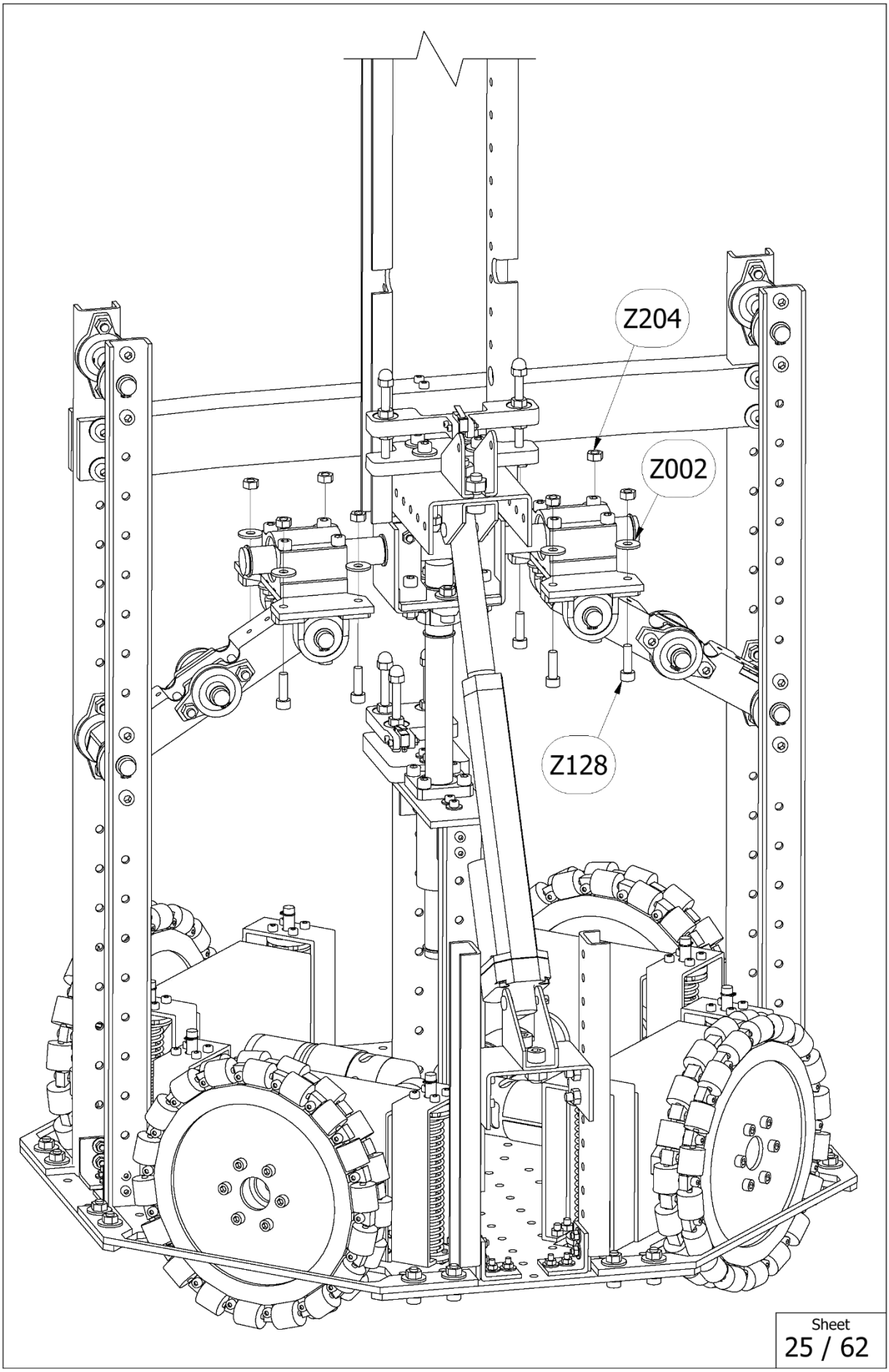

Z210
Z204
Z002
Z500
F101
Z102
R000
Z200
Z002
Z129
F100
Z002
Z002
Z204
Sheet
21 / 62

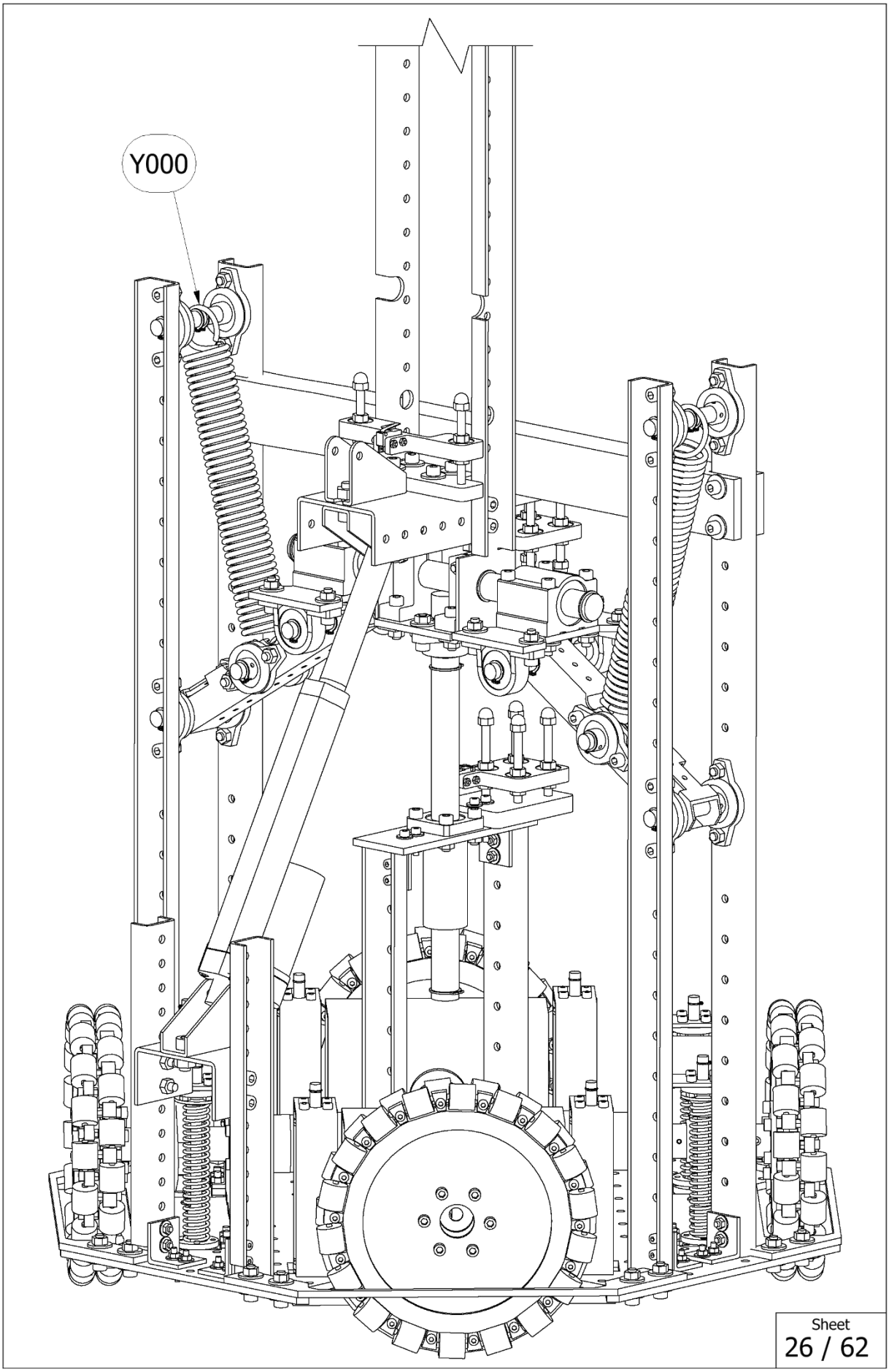

N102
N100
Z304
Sheet
22 / 62

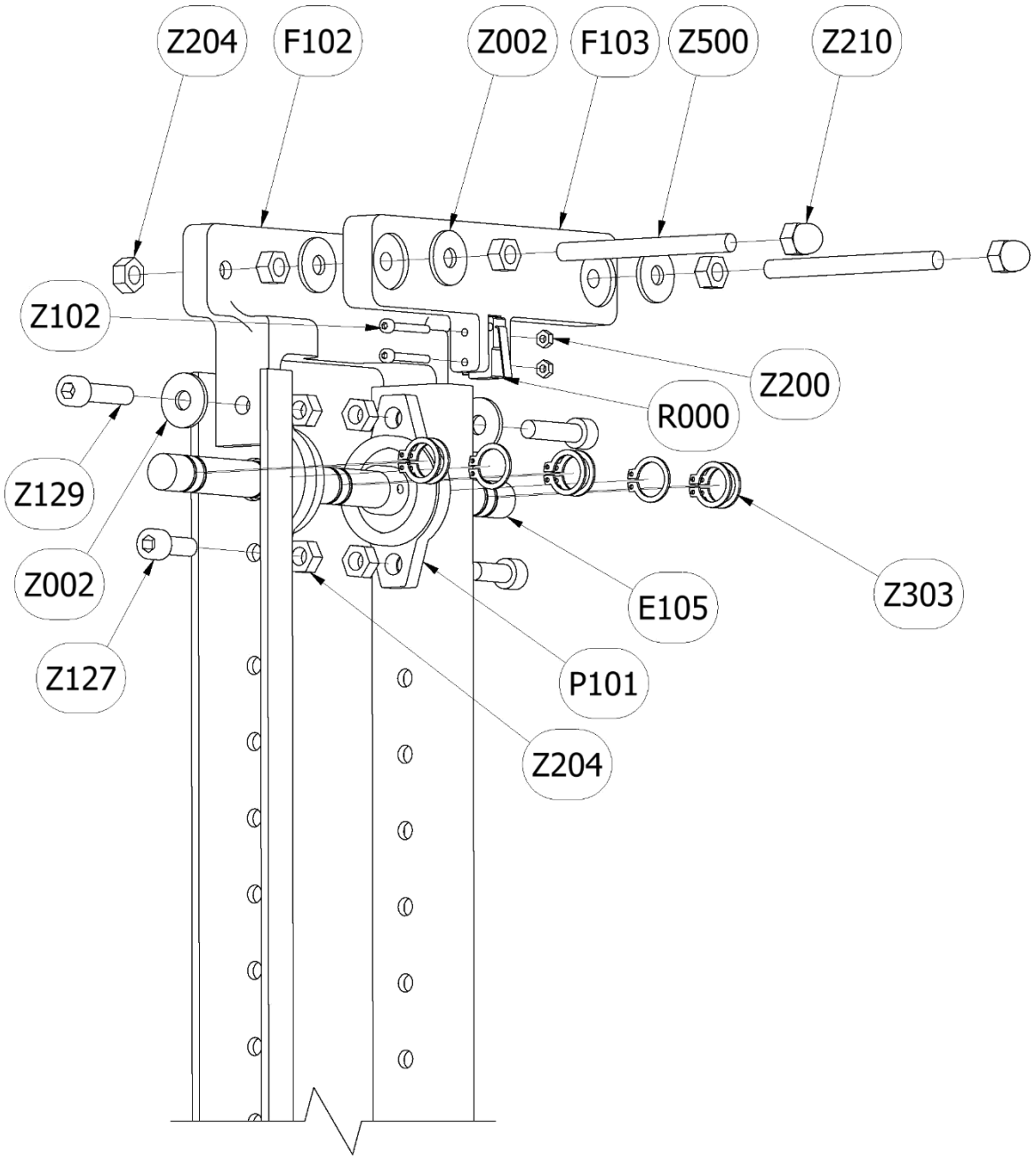

Z131
Z304
P010
H005
Z002
Z204
Sheet
23 / 62



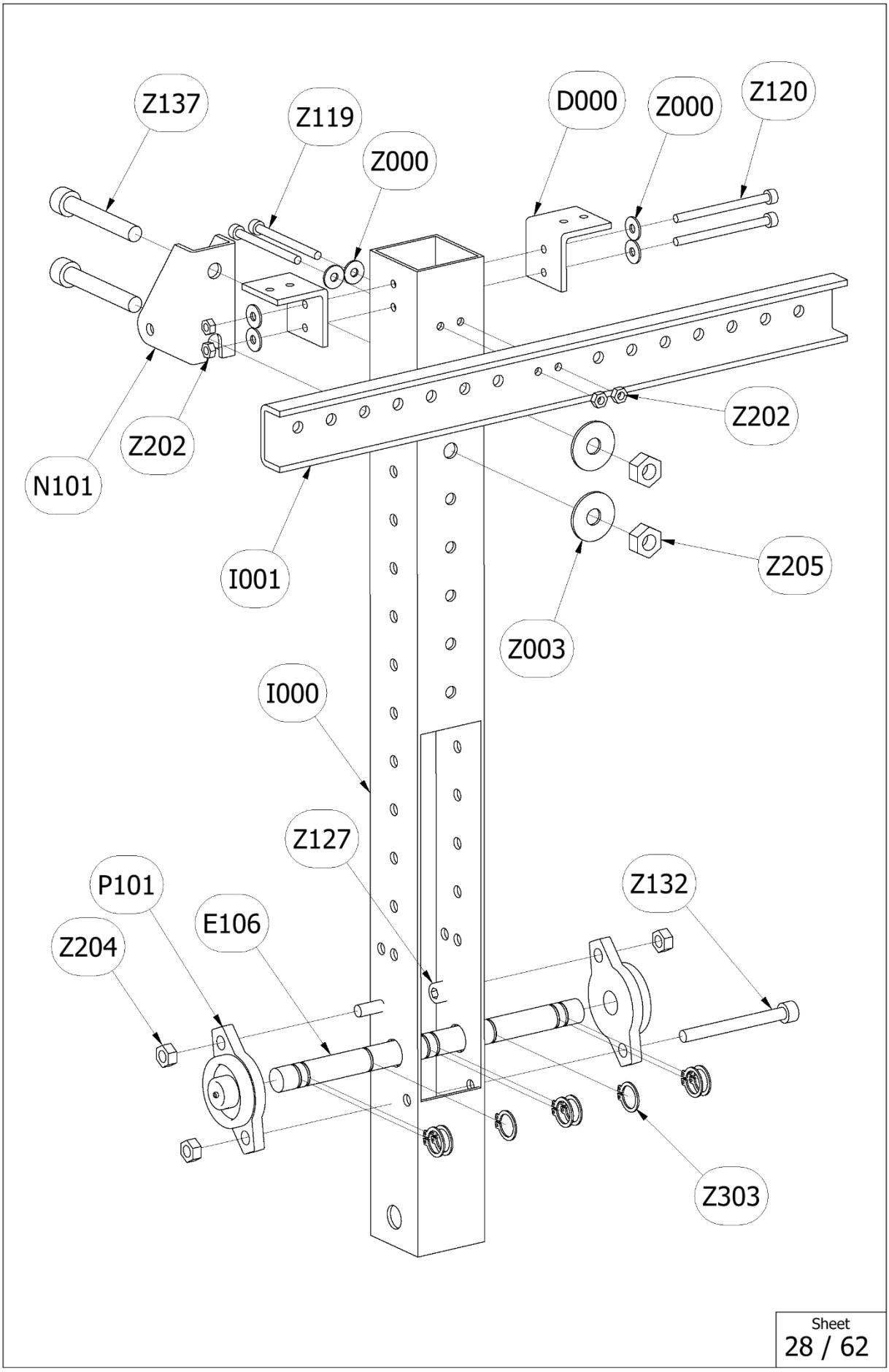
Sheet
24 / 62

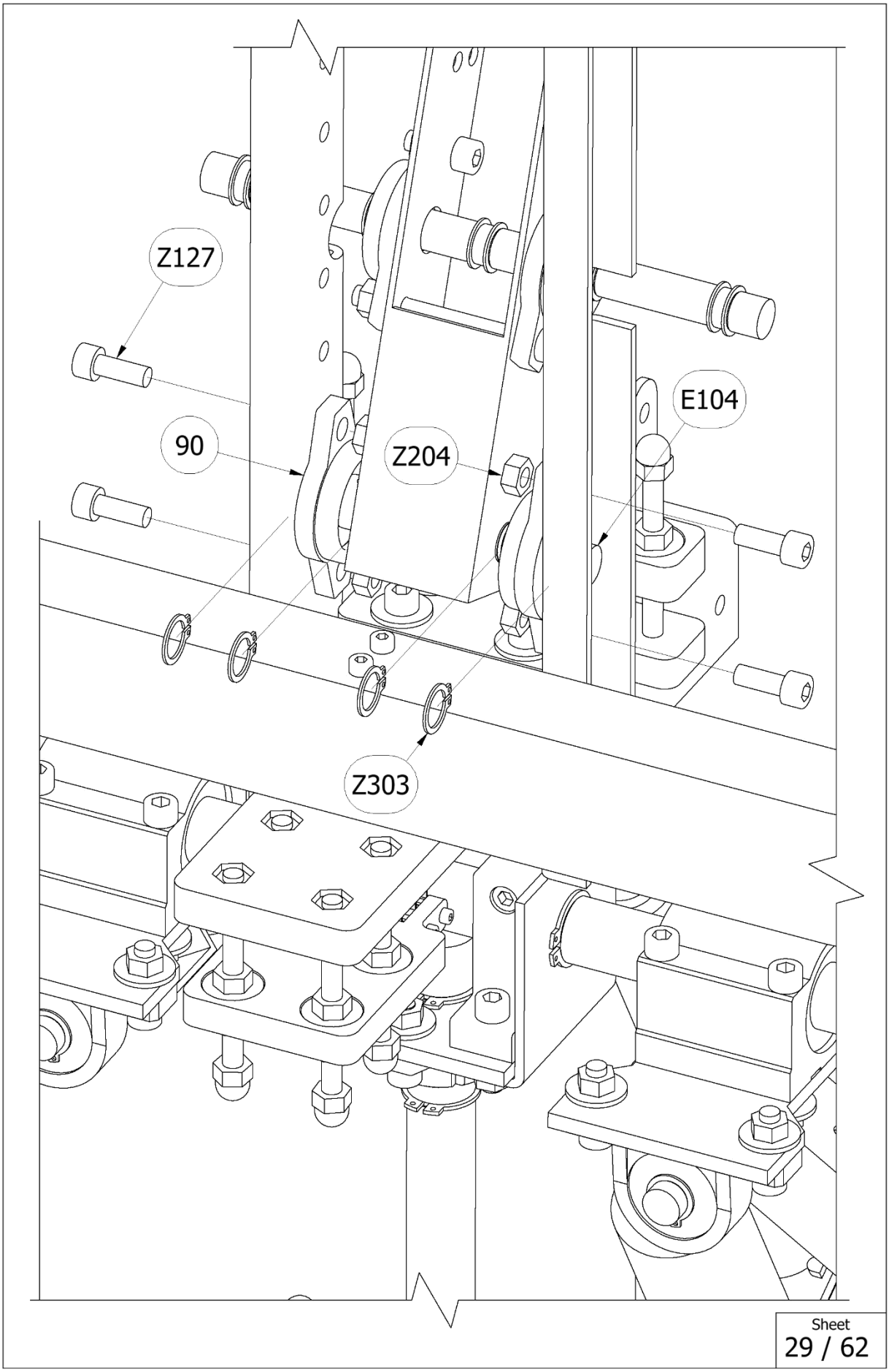

Z204
Z002
Z128
Sheet
25 / 62

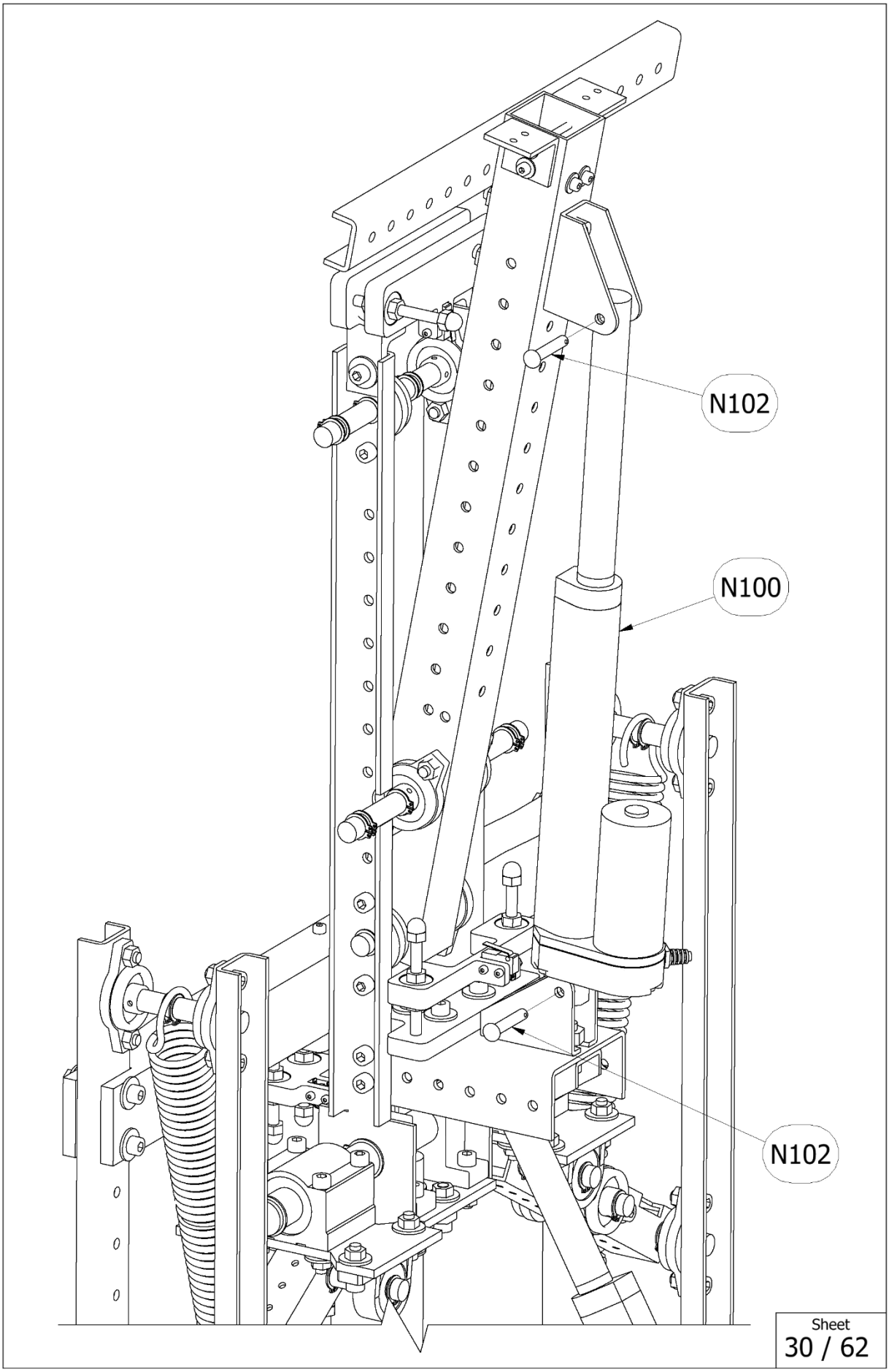

Y000
Sheet
26 / 62

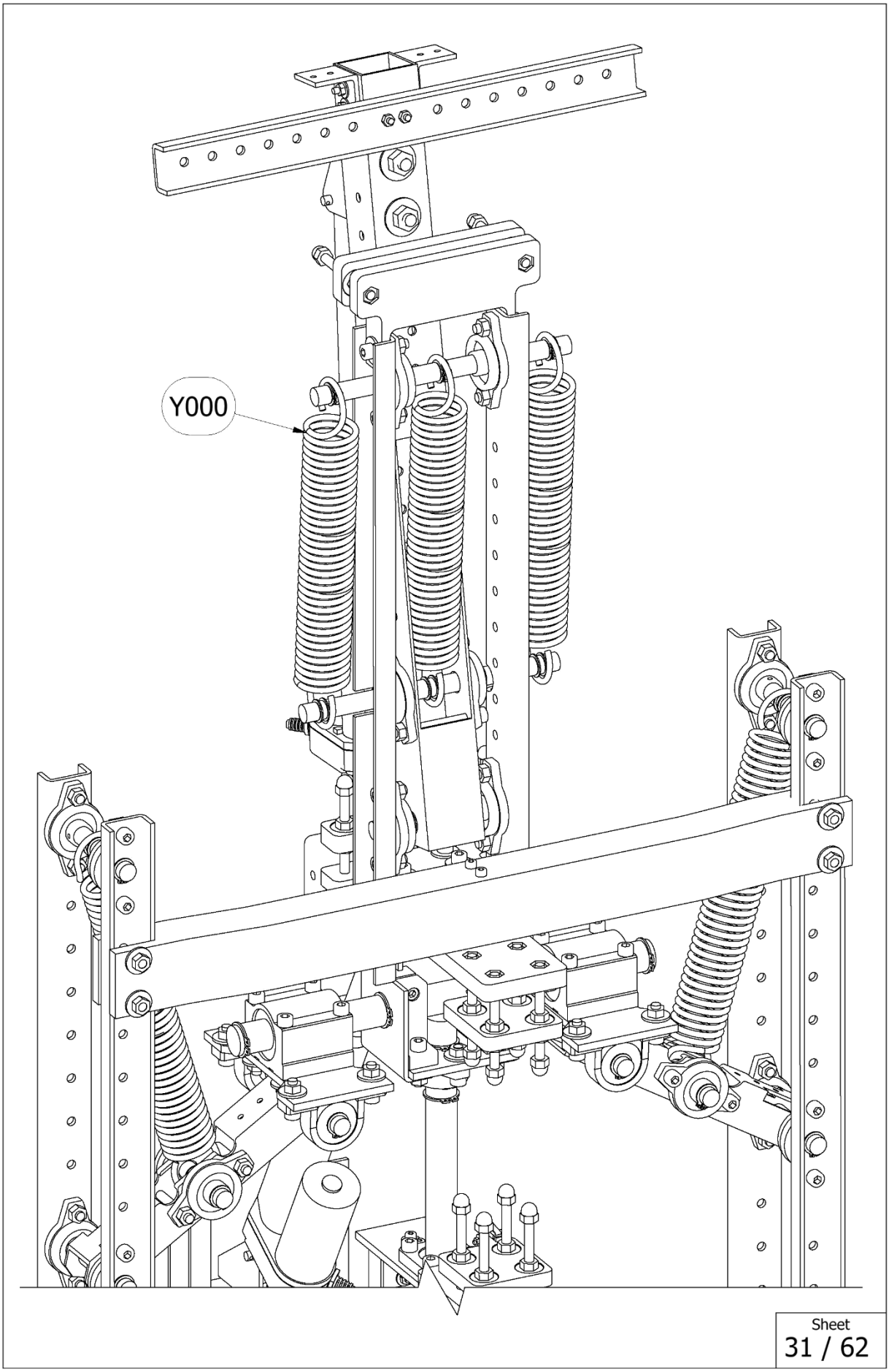

Z204
F102
Z002
F103
Z500
Z210
Z102
Z200
R000
Z129
Z002
Z303
E105
Z127
P101
Z204

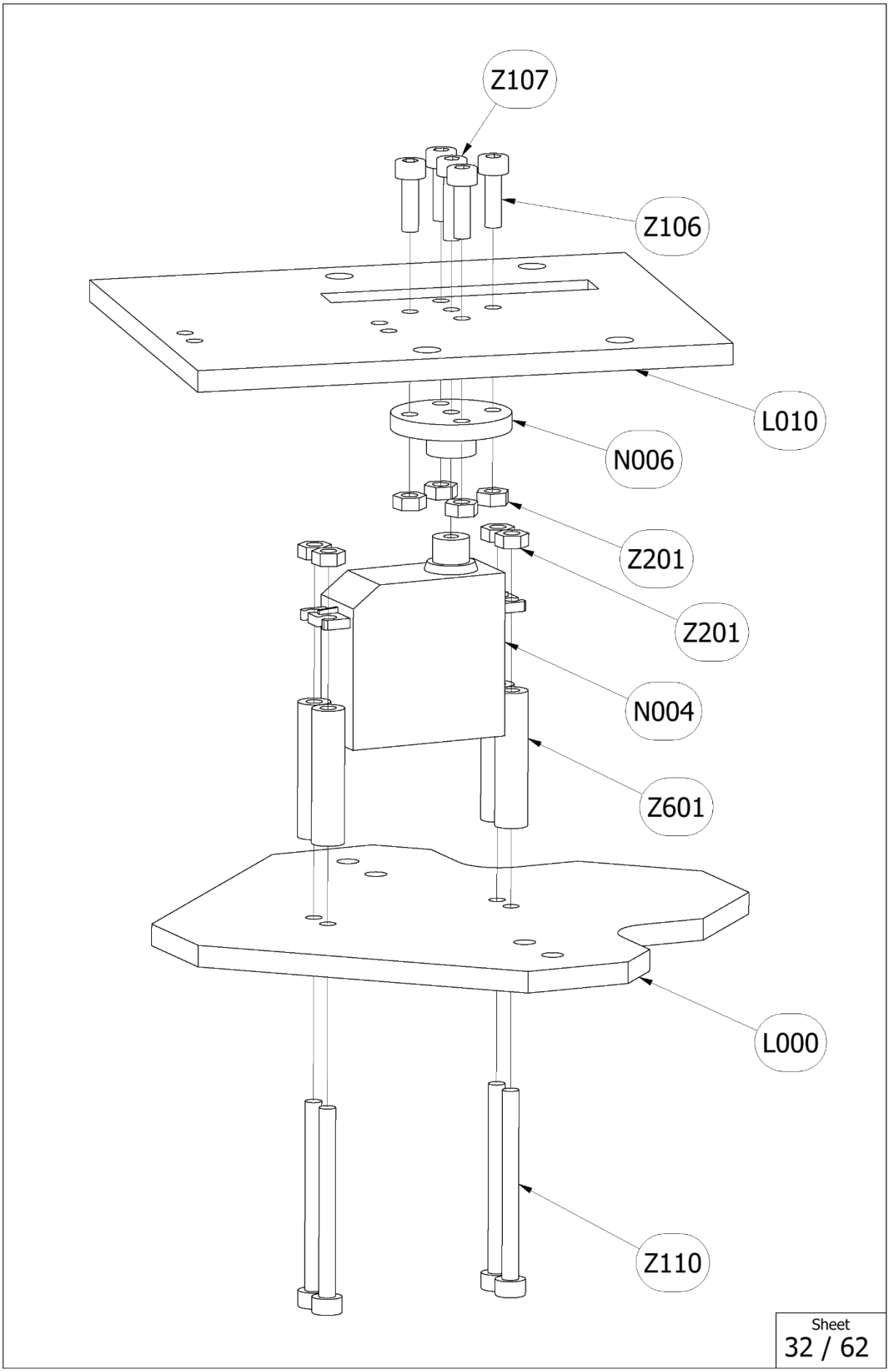
Sheet
27 / 62

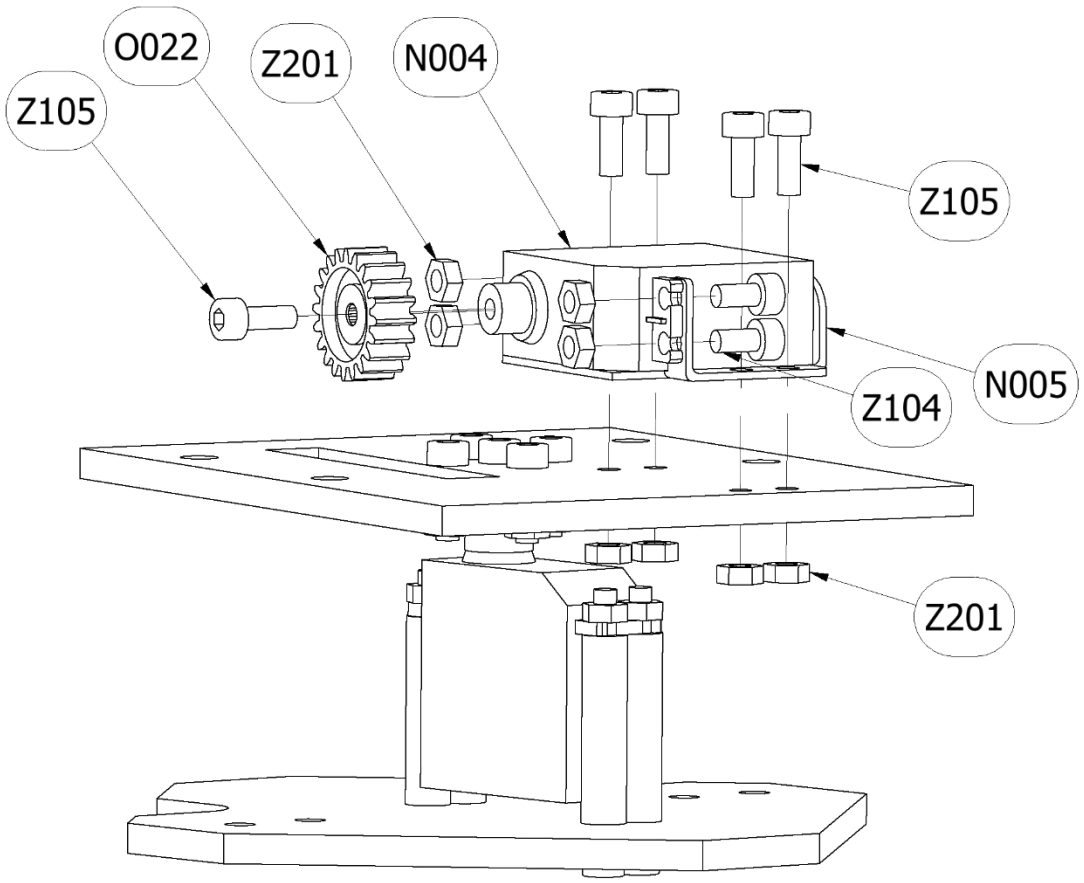

Z137
Z119
Z000
D000
Z000
Z120
Z202
N101
Z202
I001
Z205
Z003
I000
Z127
P101
E106
Z204
Z132
Z303
Sheet
28 / 62


Z127
90
Z204
E104
Z303
Sheet
29 / 62

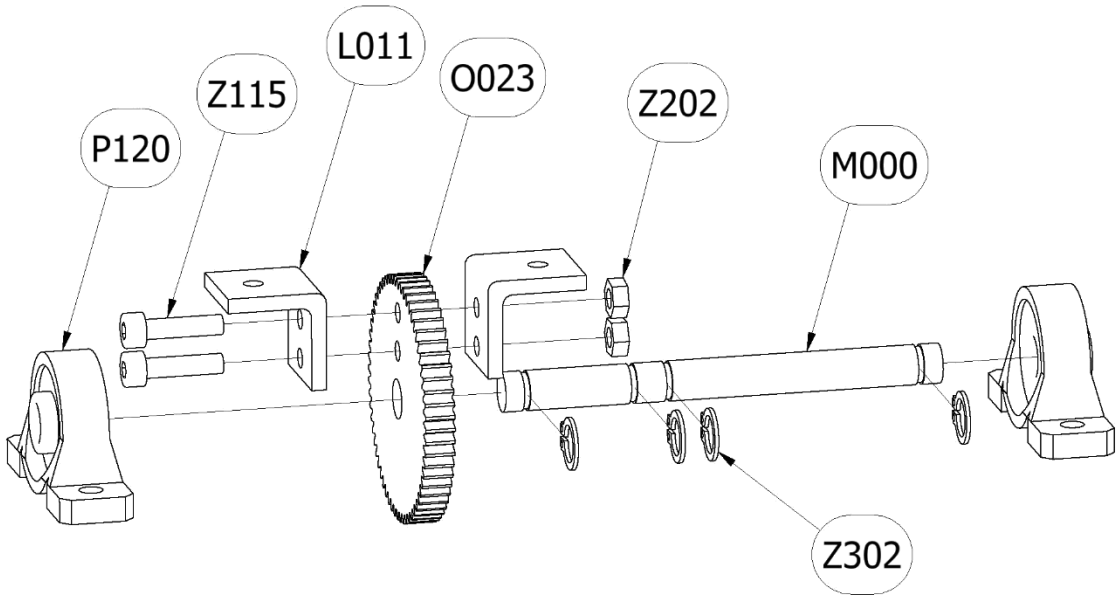

N102
N100
N102
Sheet
30 / 62


Y000
Sheet
31 / 62

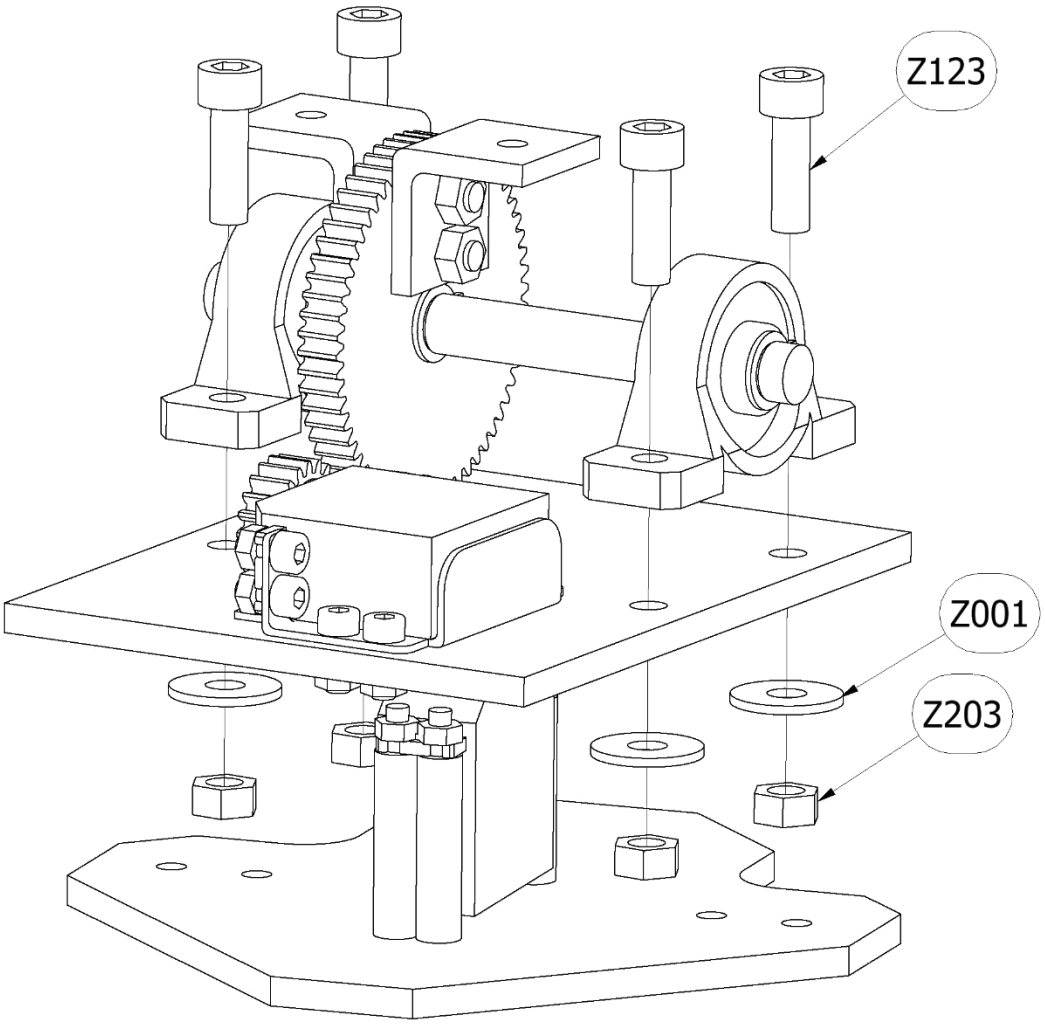

Z107
Z106
L010
N006
Z201
Z201
N004
Z601
L000
Z110
Sheet
32 / 62



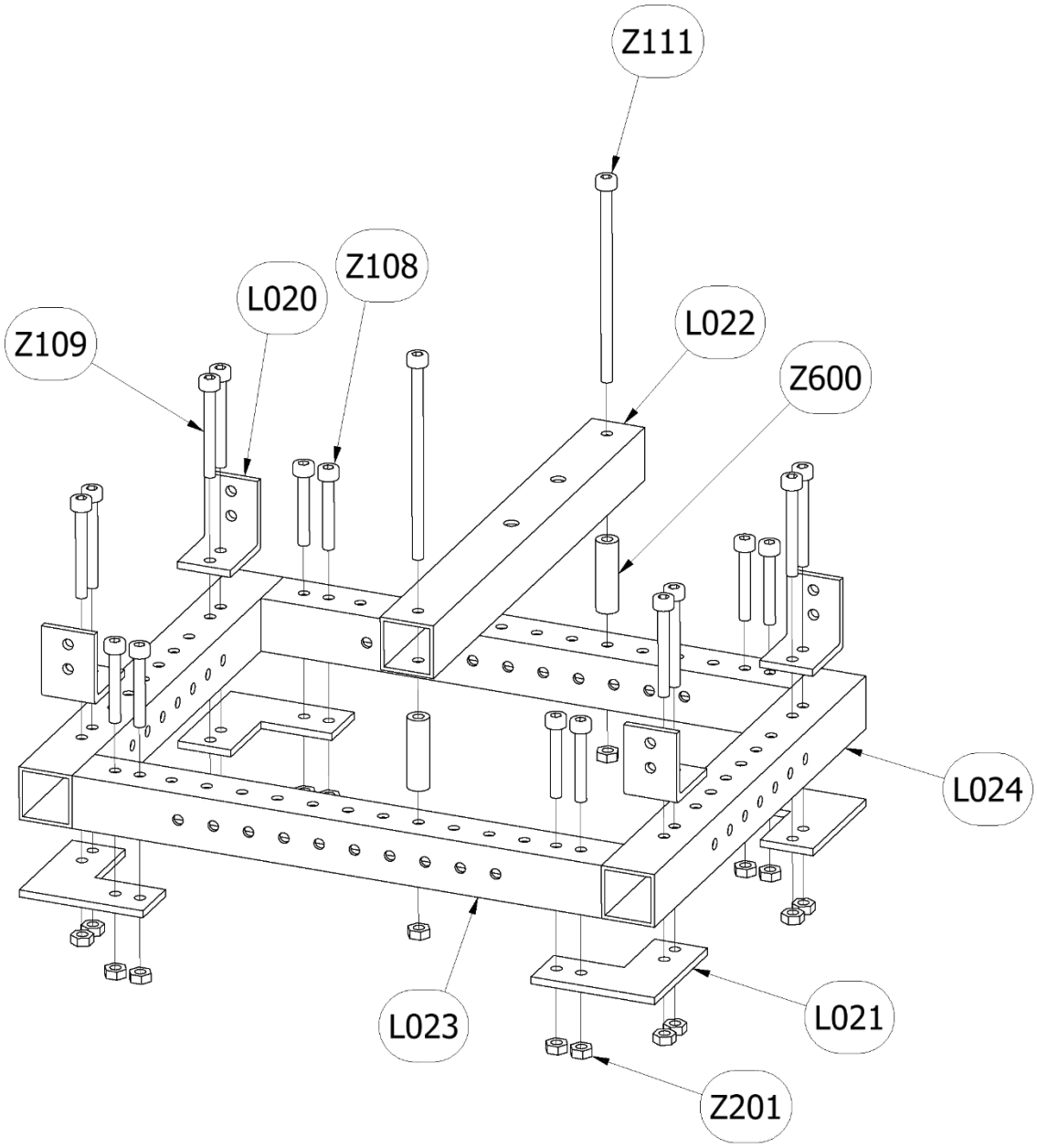
Sheet
33 / 62



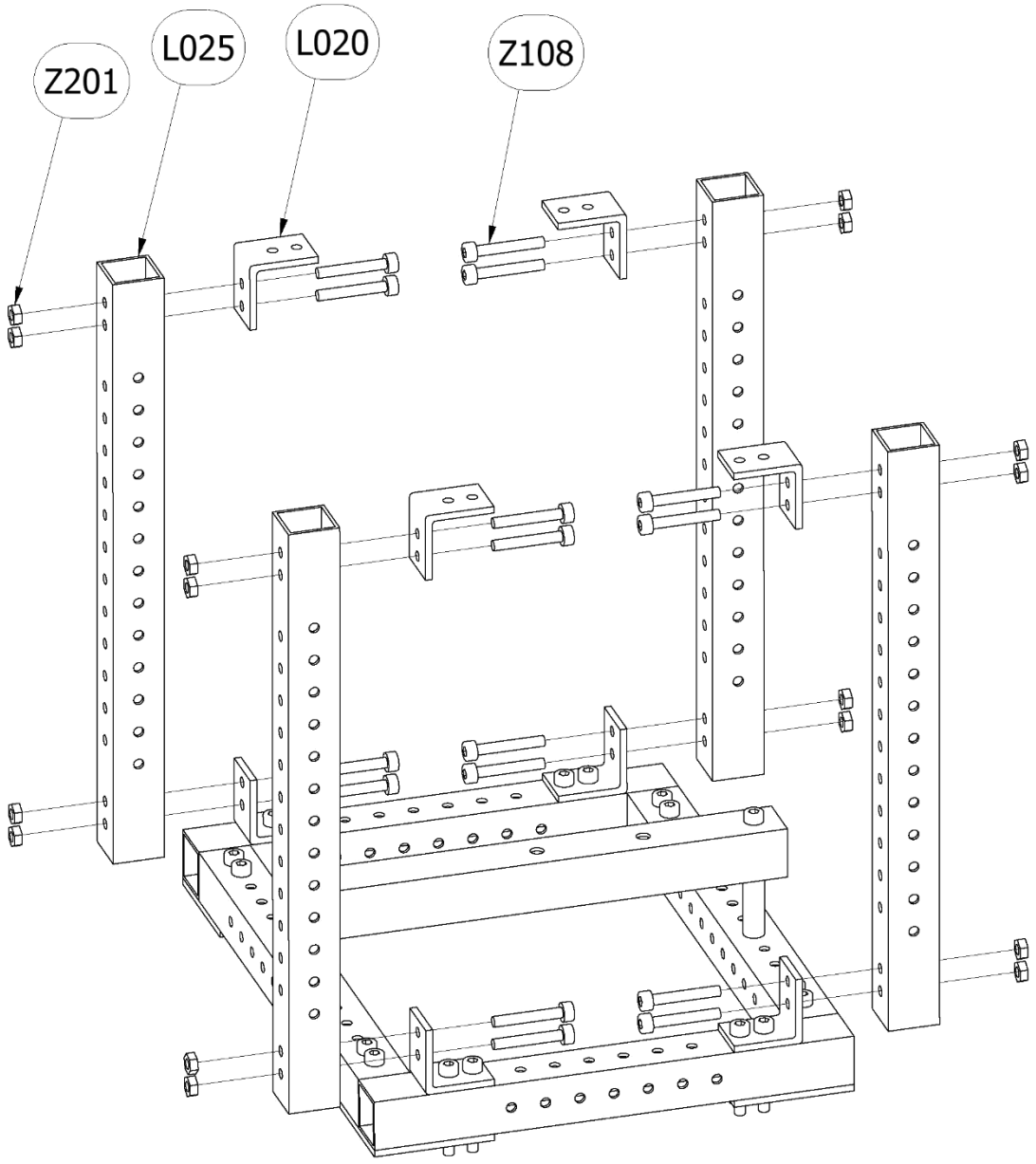
Sheet
34 / 62



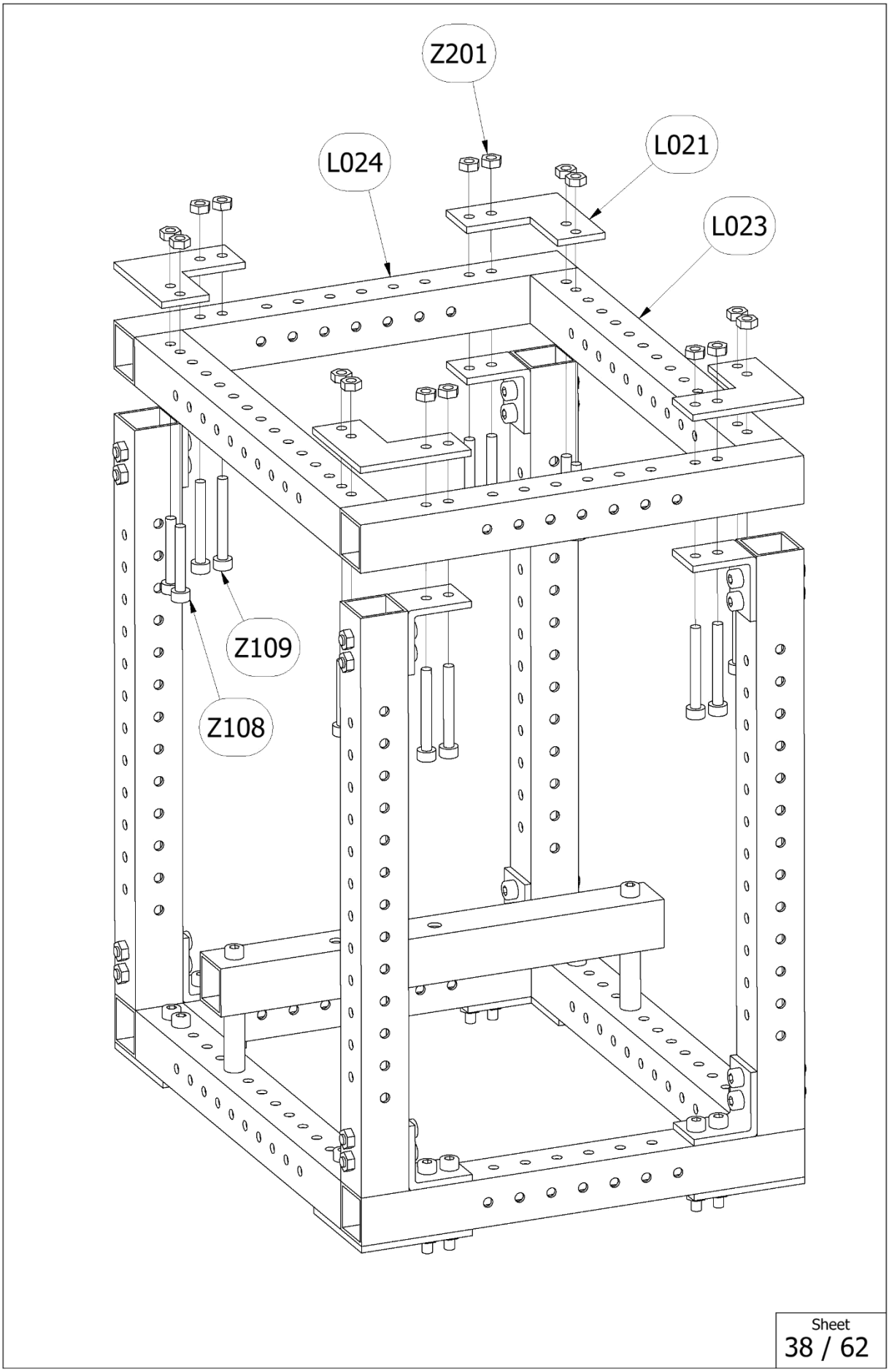
Sheet
35 / 62



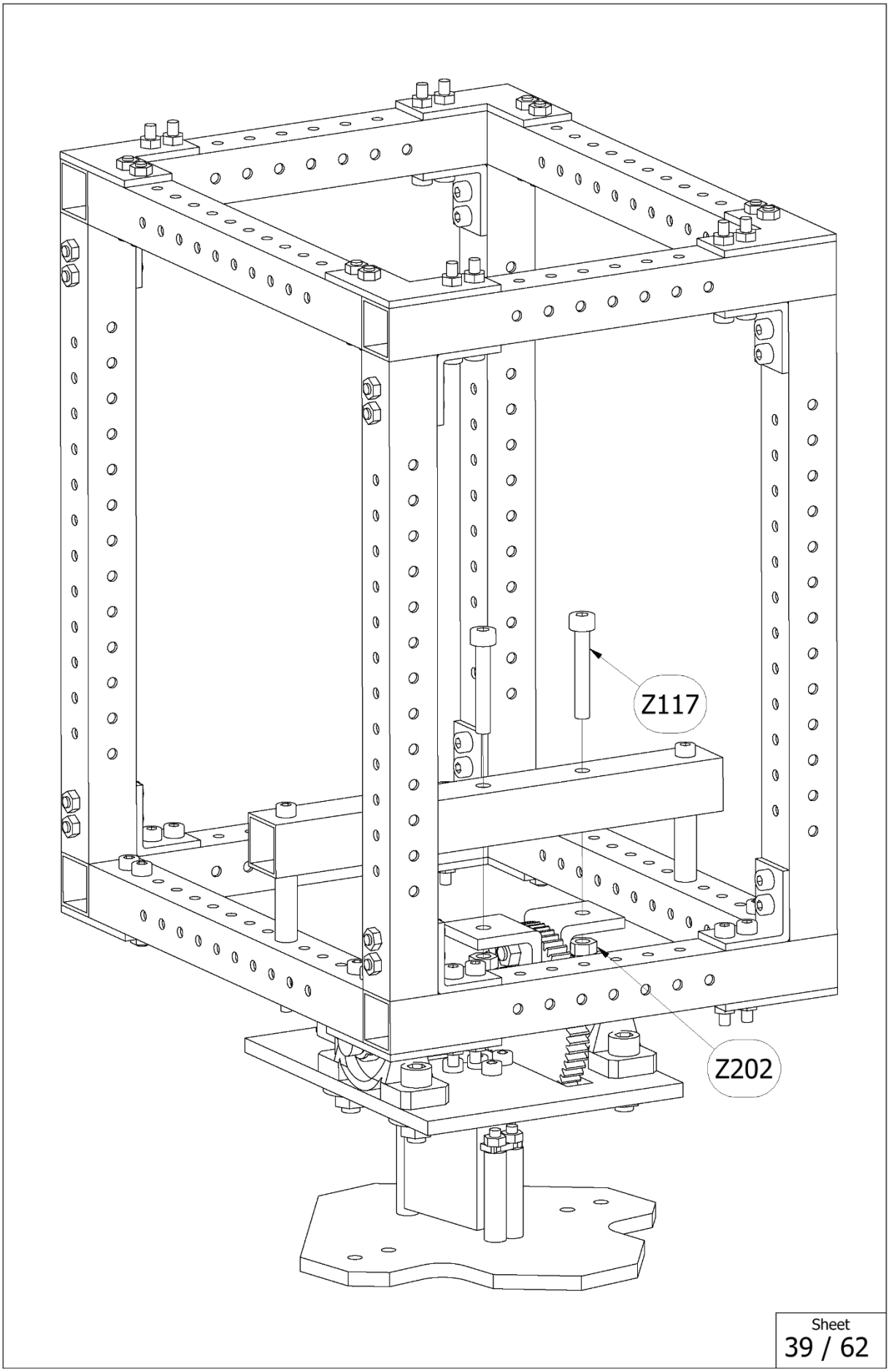
Sheet
36 / 62

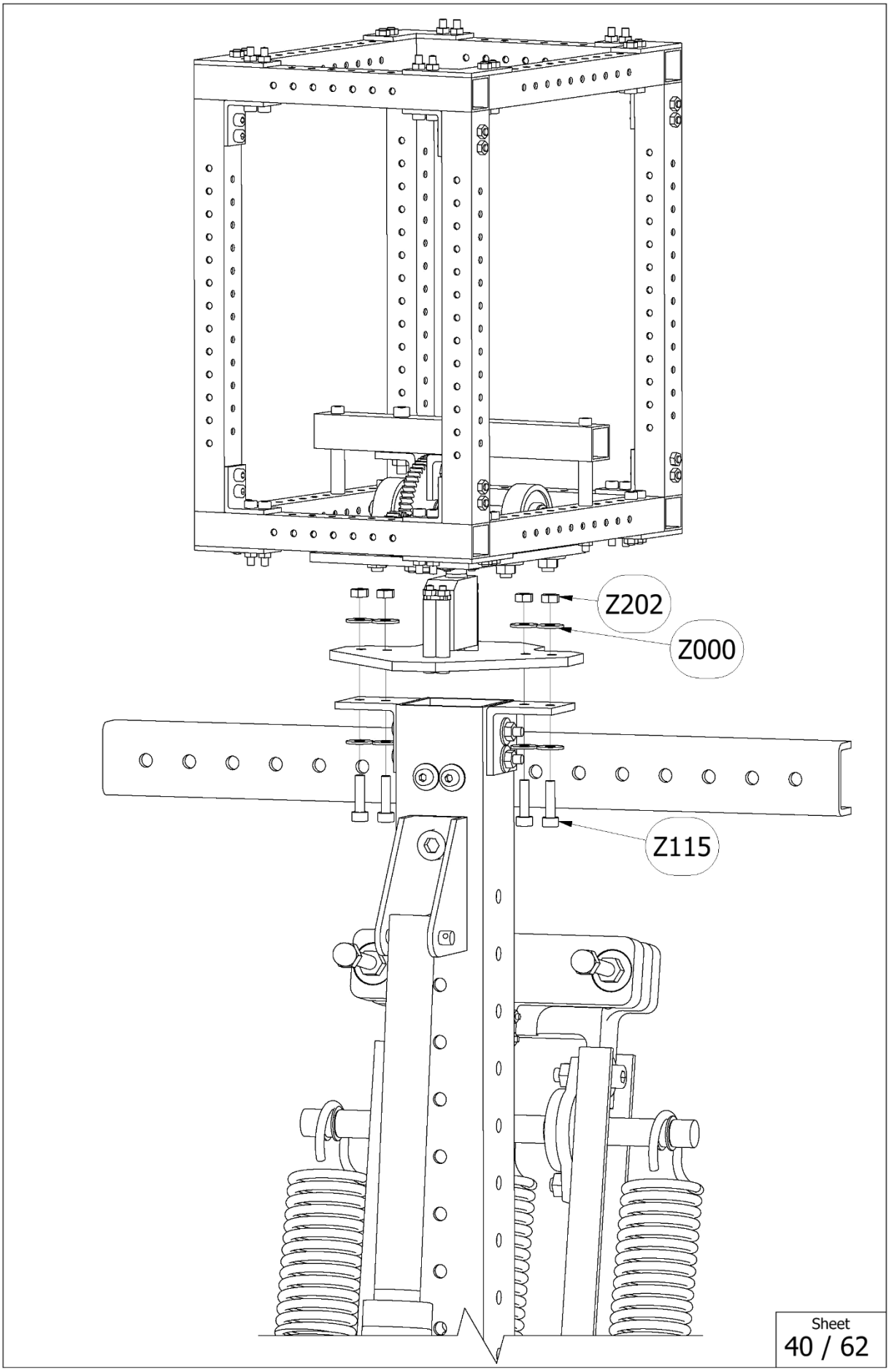


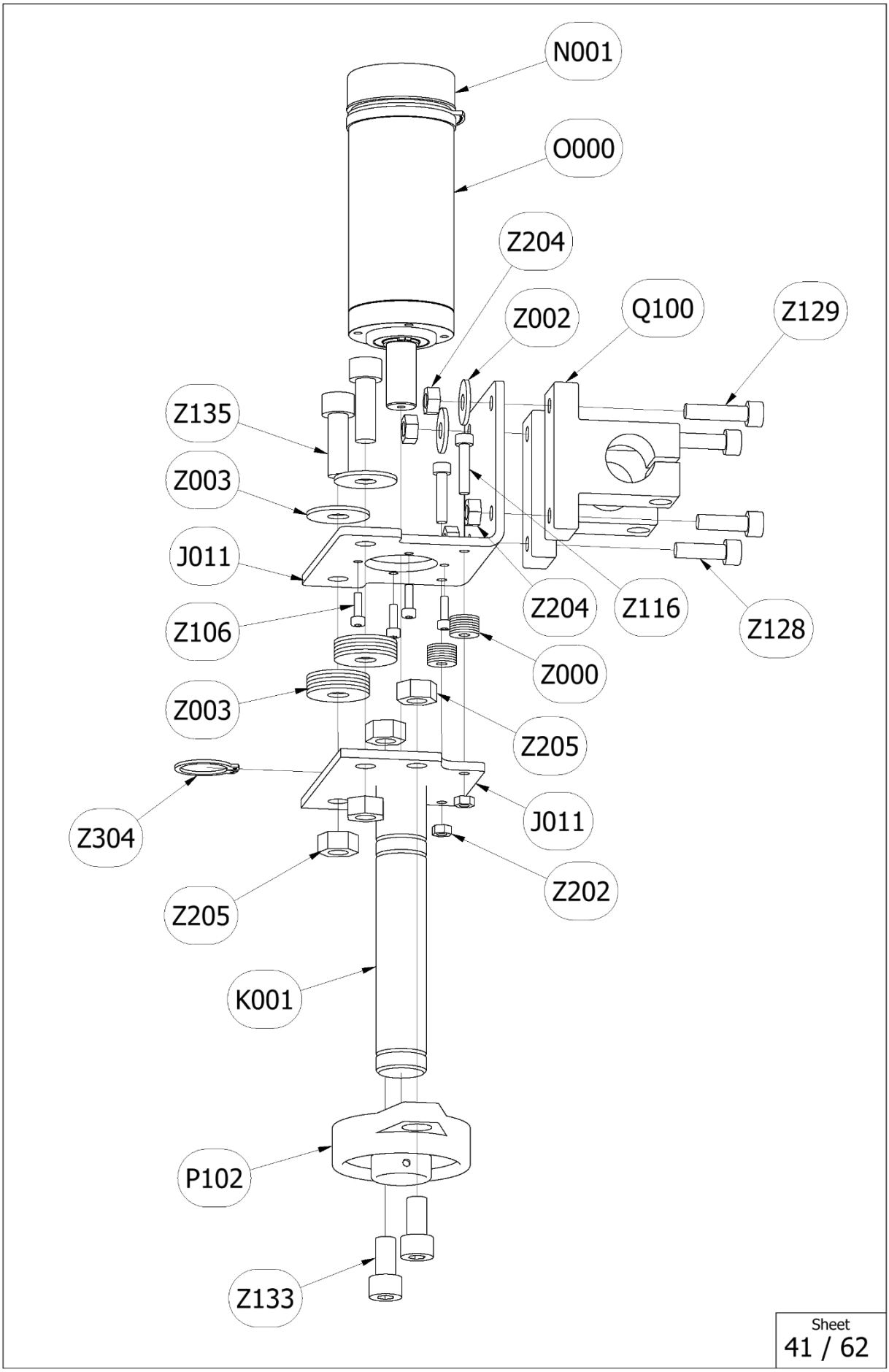
Sheet
37 / 62

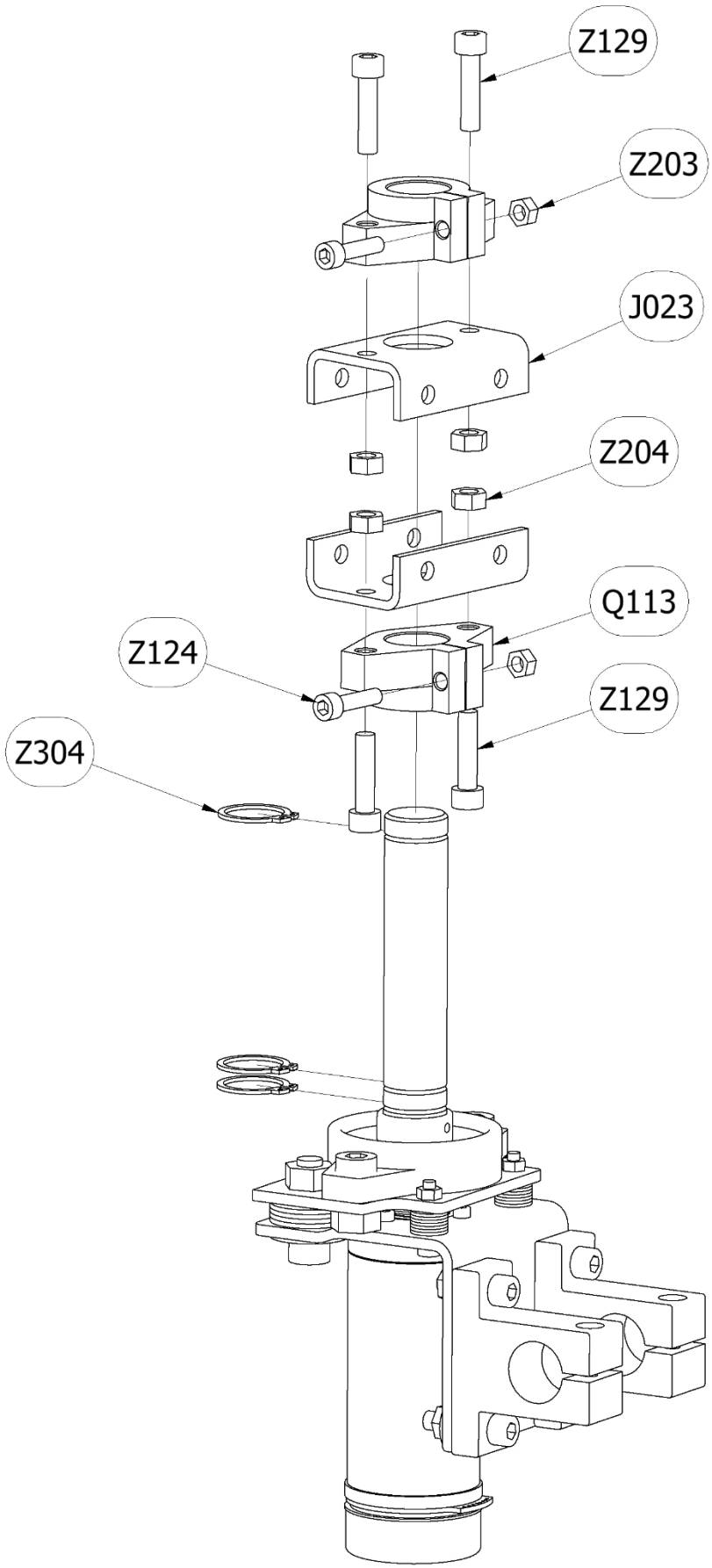


Sheet
38 / 62

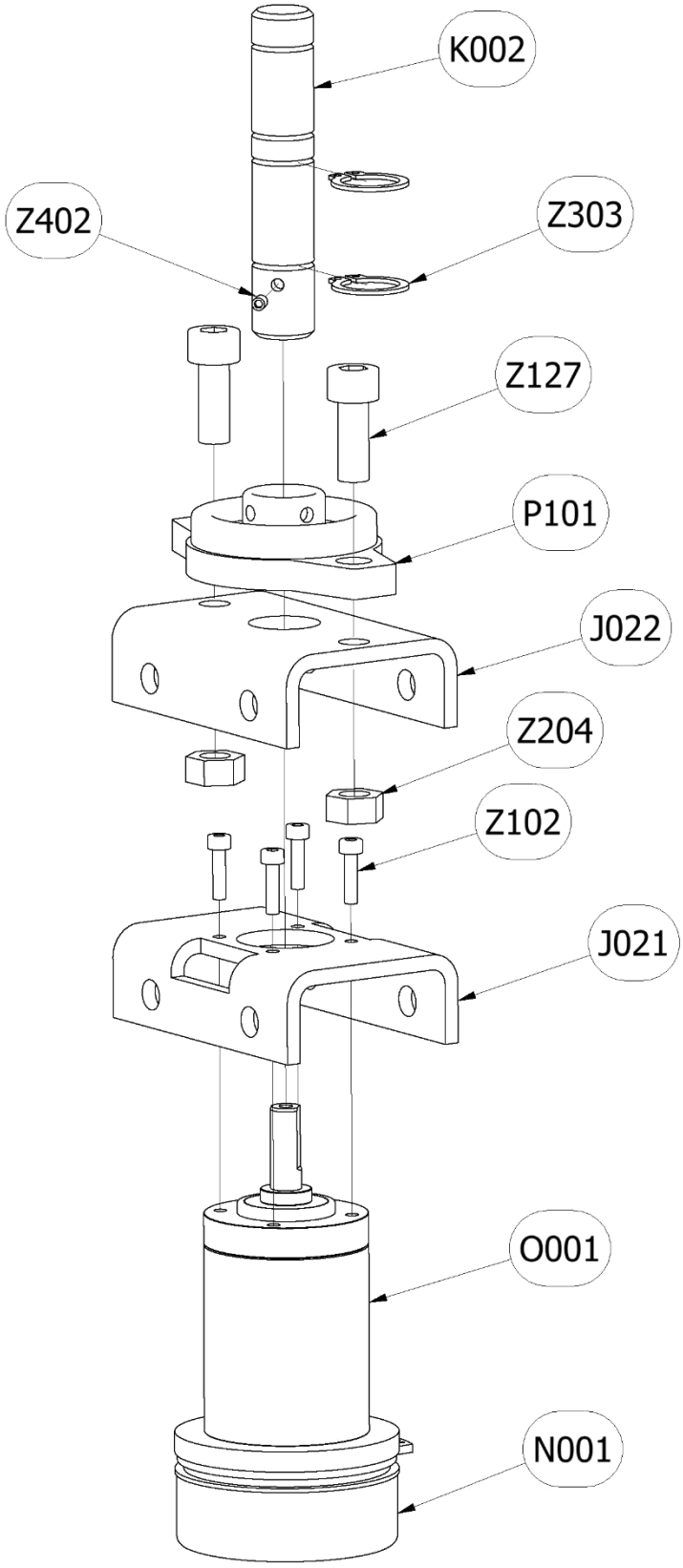

Z117
Z202
Sheet
39 / 62


Z202
Z000
Z115
Sheet
40 / 62

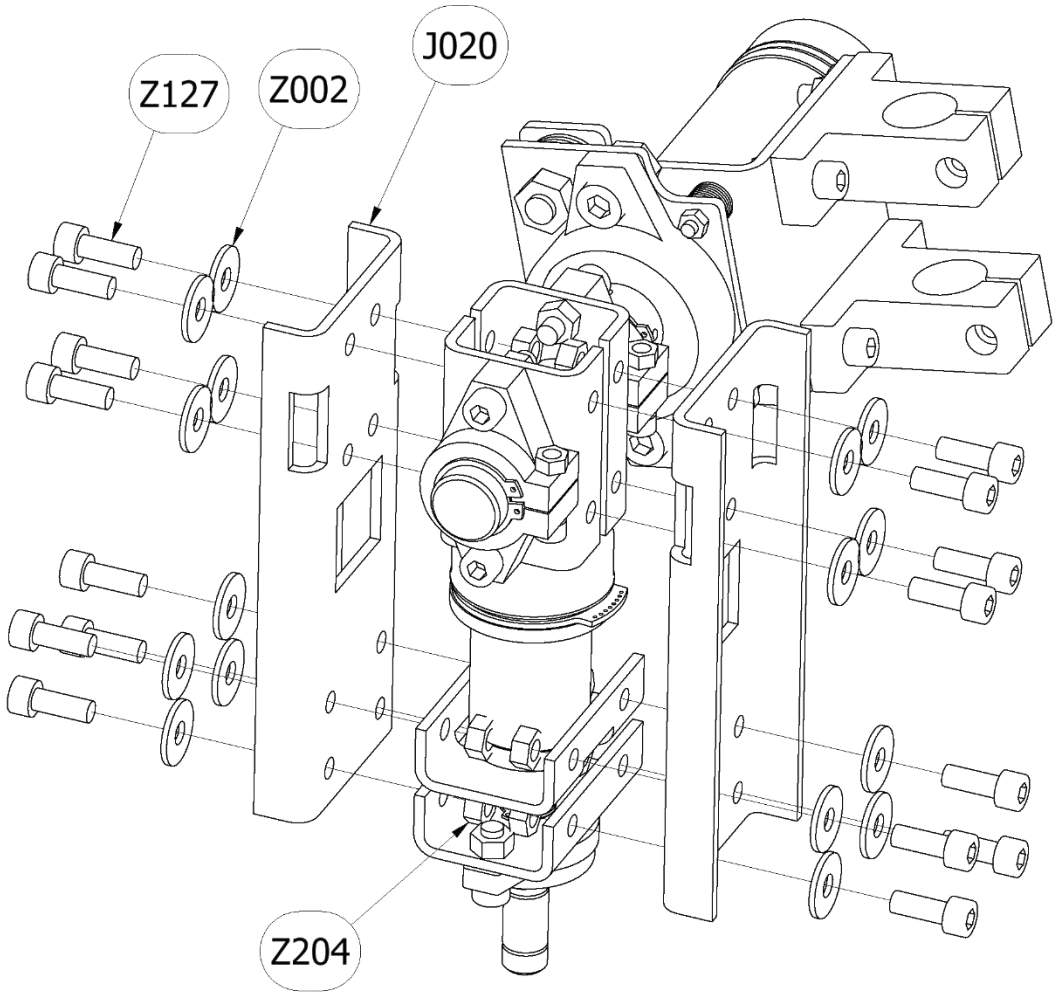

N001
O000
Z204
Z002
Q100
Z129
Z135
Z003
J011
Z106
Z003
Z304
Z205
K001
P102
Z133
Z204
Z116
Z128
Z000
Z205
J011
Z202
Sheet
41 / 62



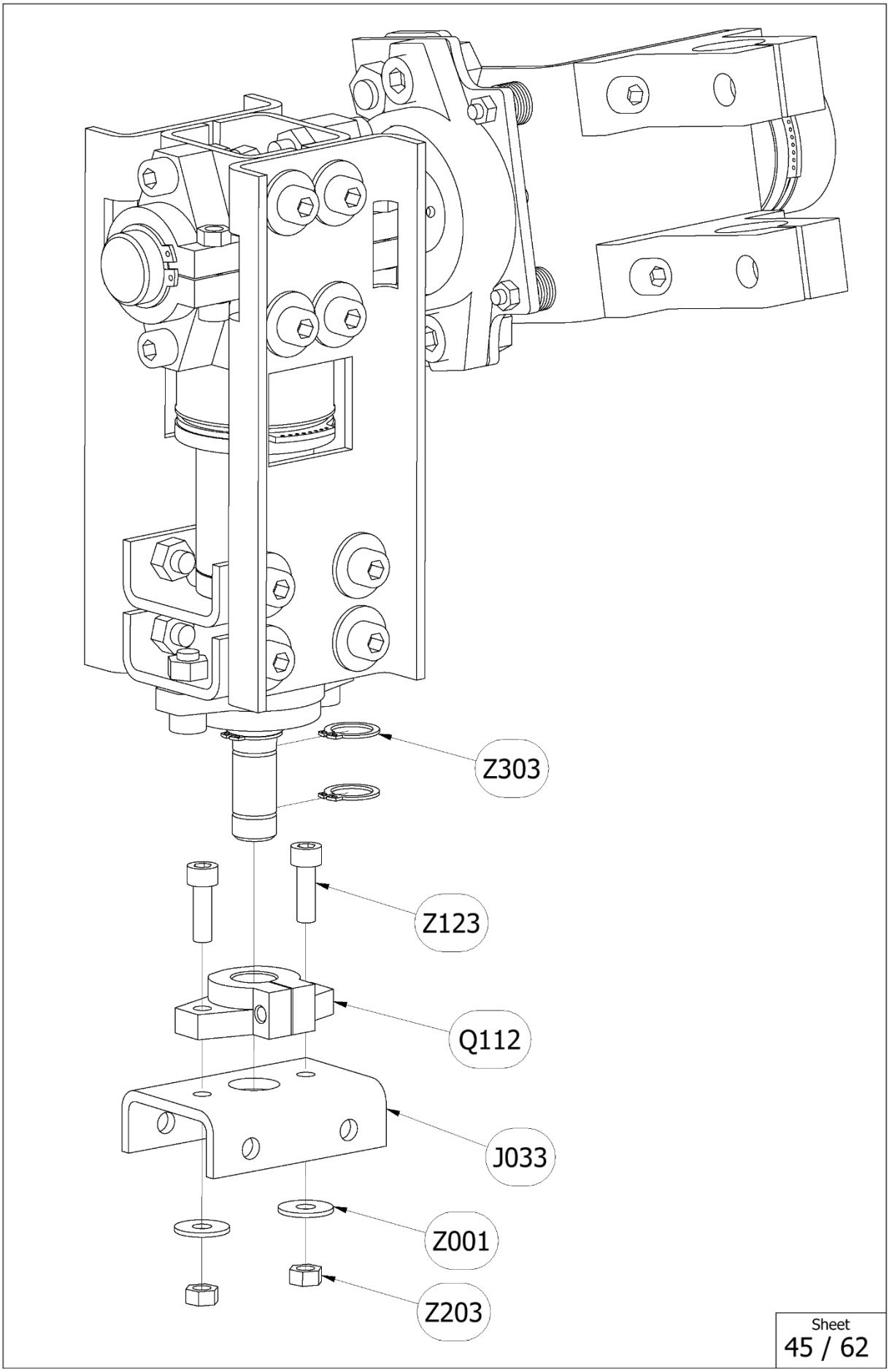
Sheet
42 / 62

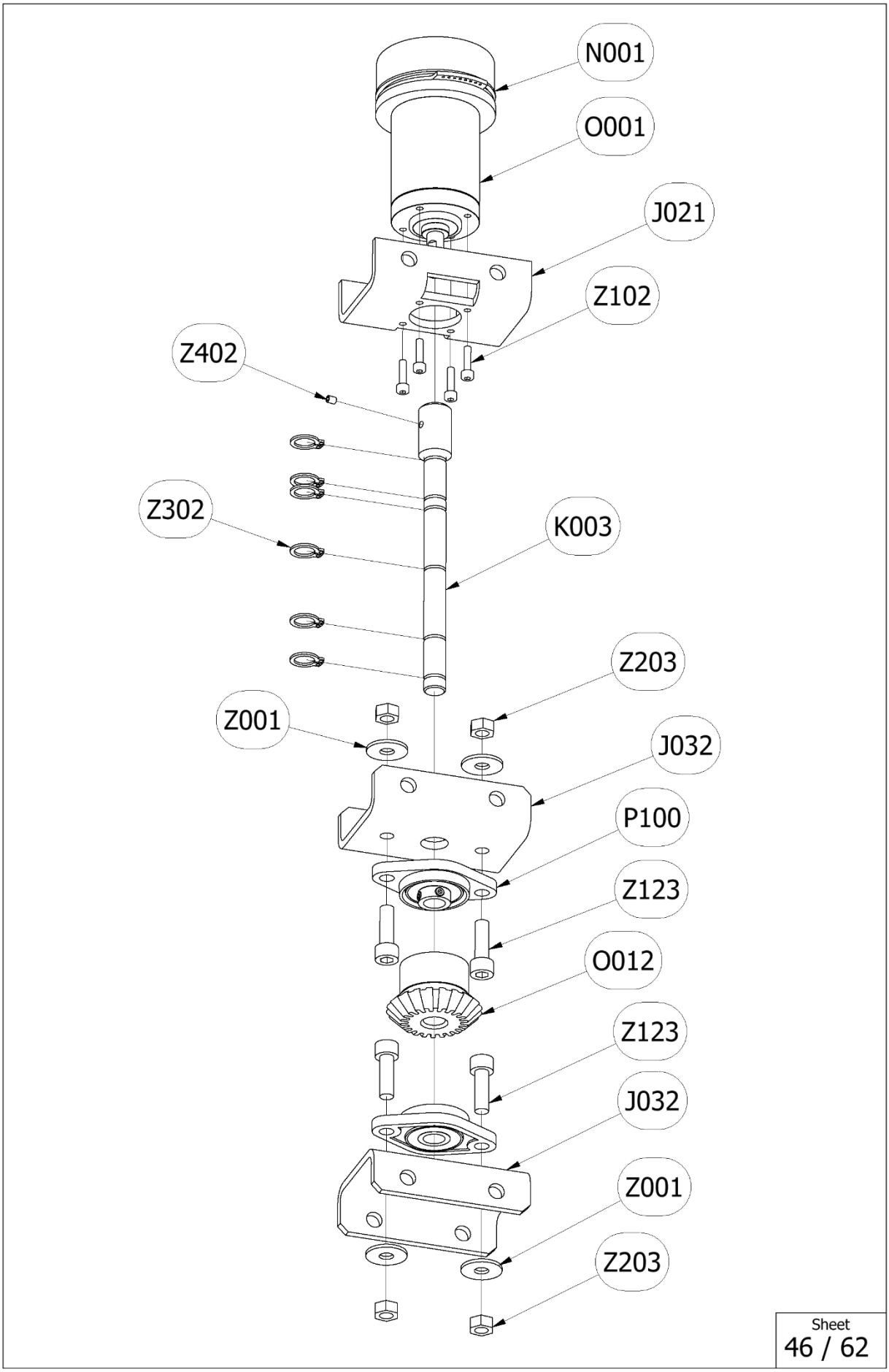


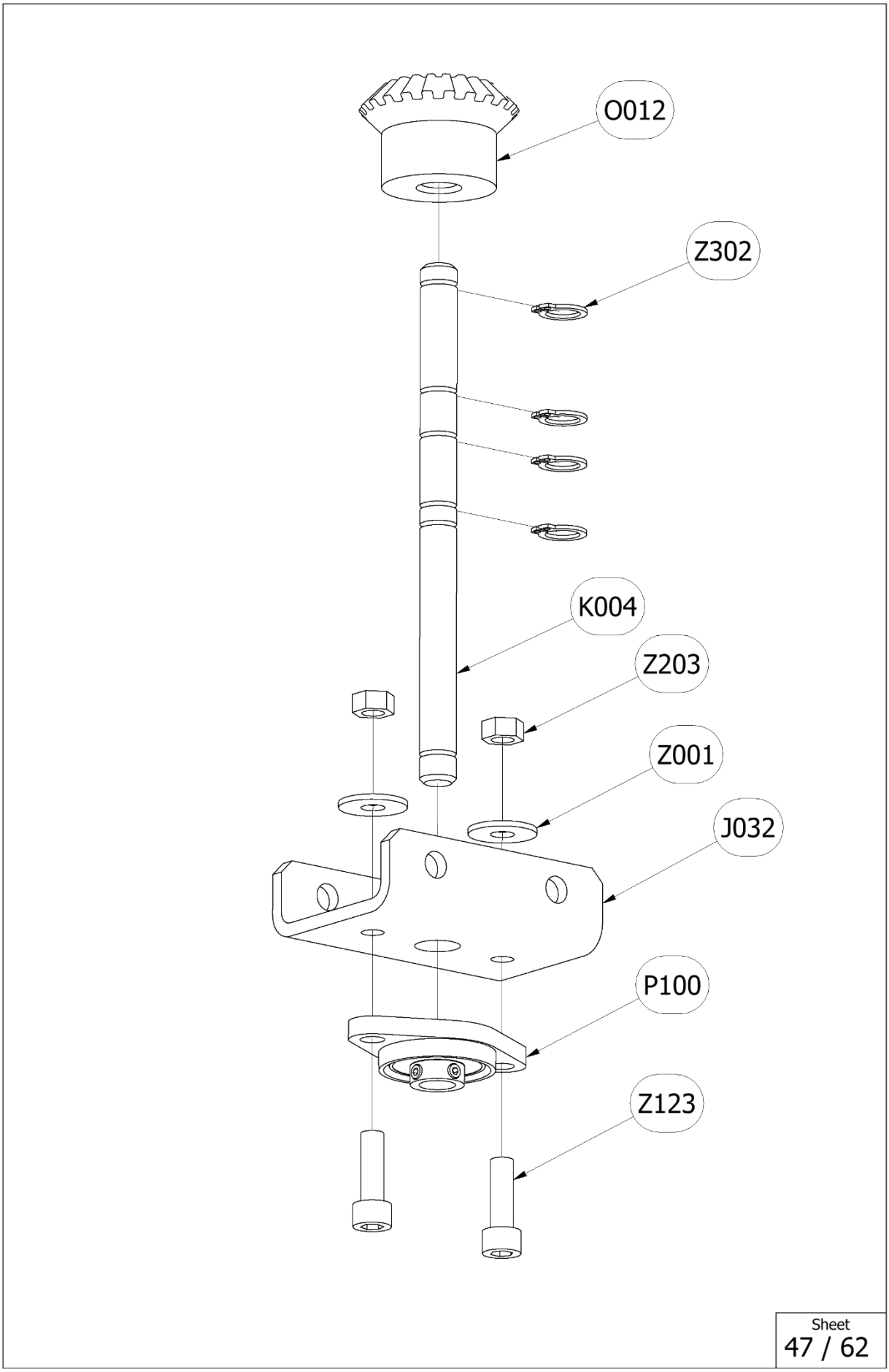
Sheet
43 / 62

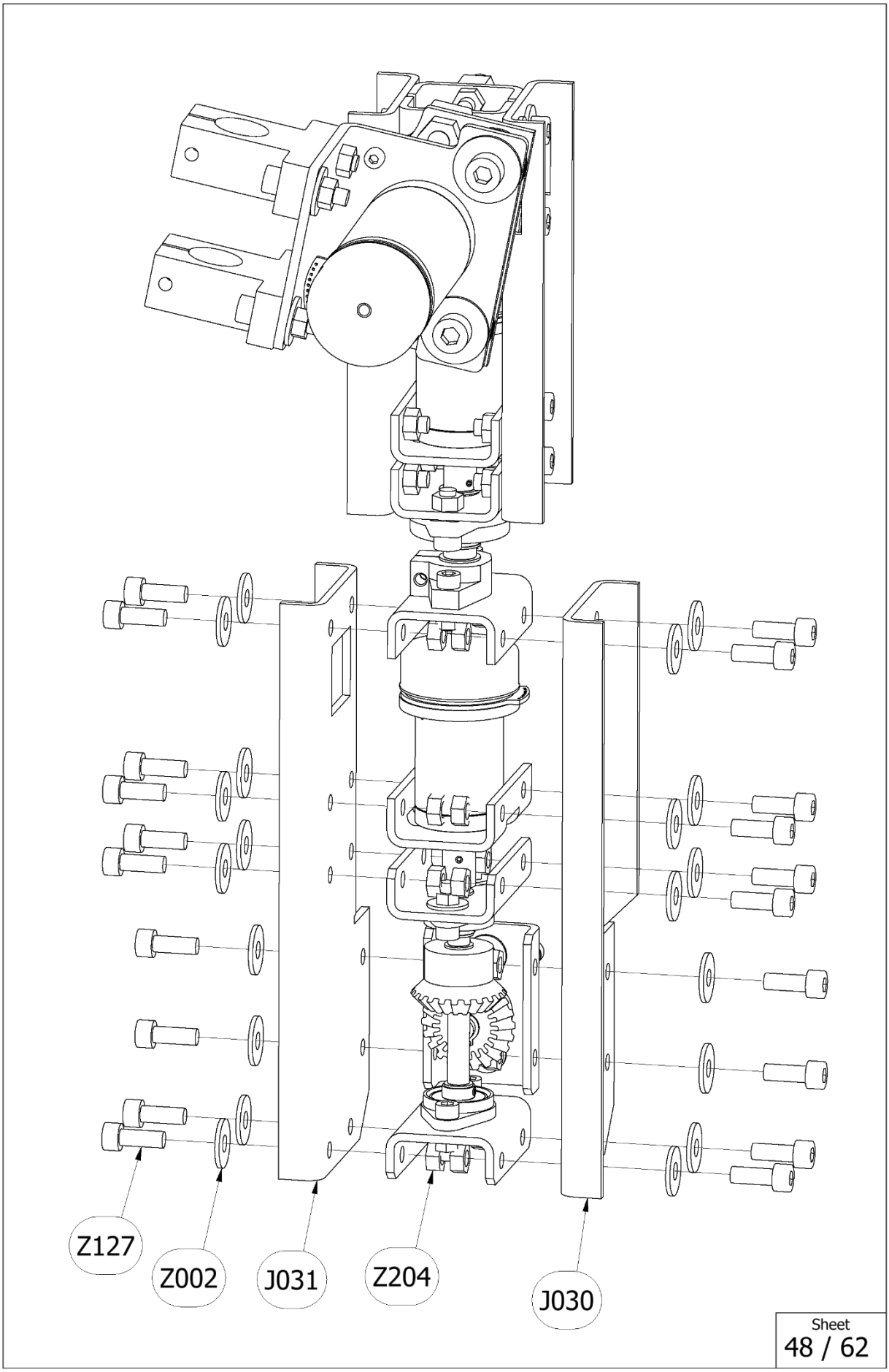


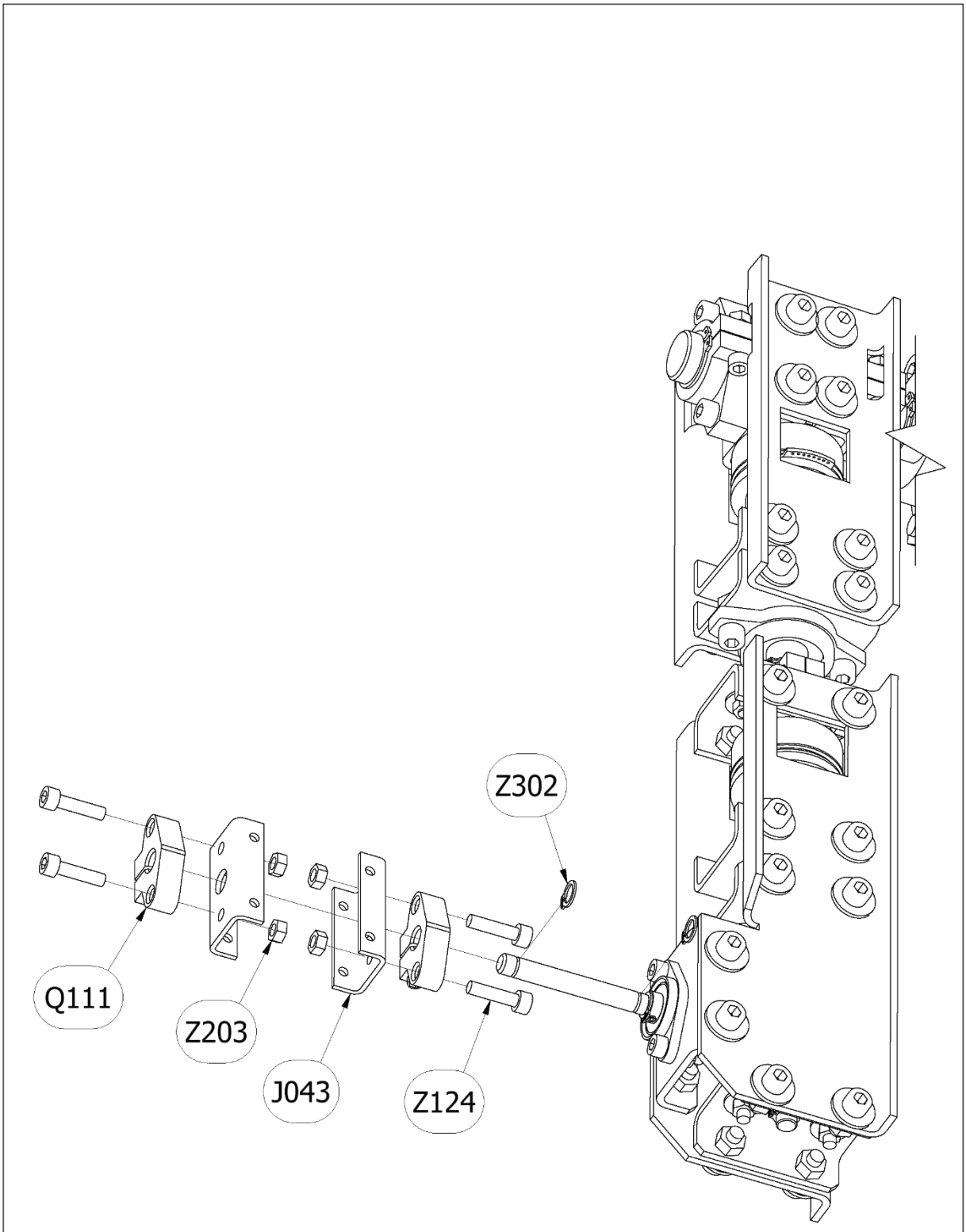
Sheet
44 / 62


Z303
Z123
Q112
J033
Z001
Z203
Sheet
45 / 62

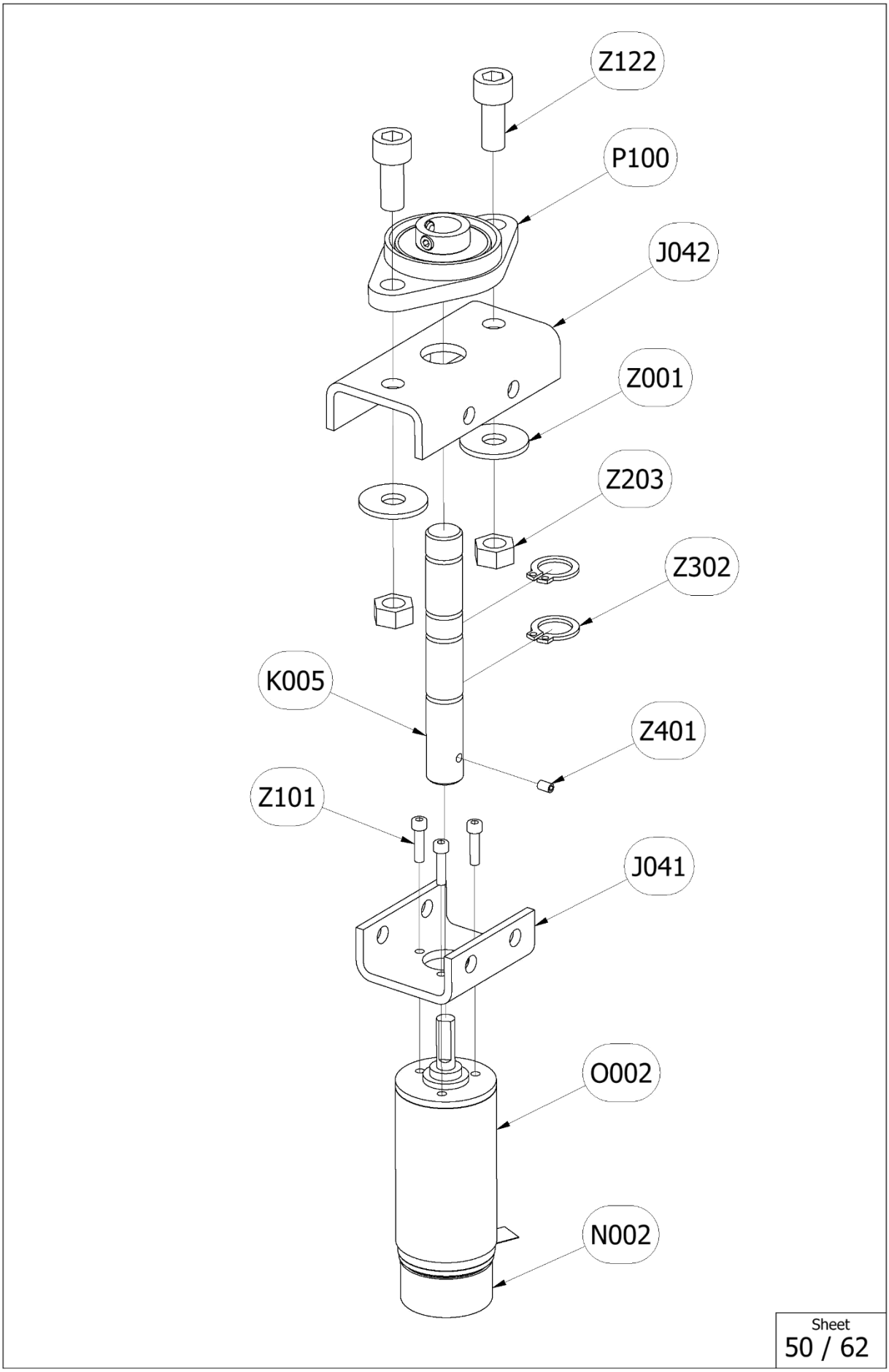

N001
O001
J021
Z102
Z402
Z302
K003
Z203
Z001
J032
P100
Z123
O012
Z123
J032
Z001
Z203
Sheet
46 / 62

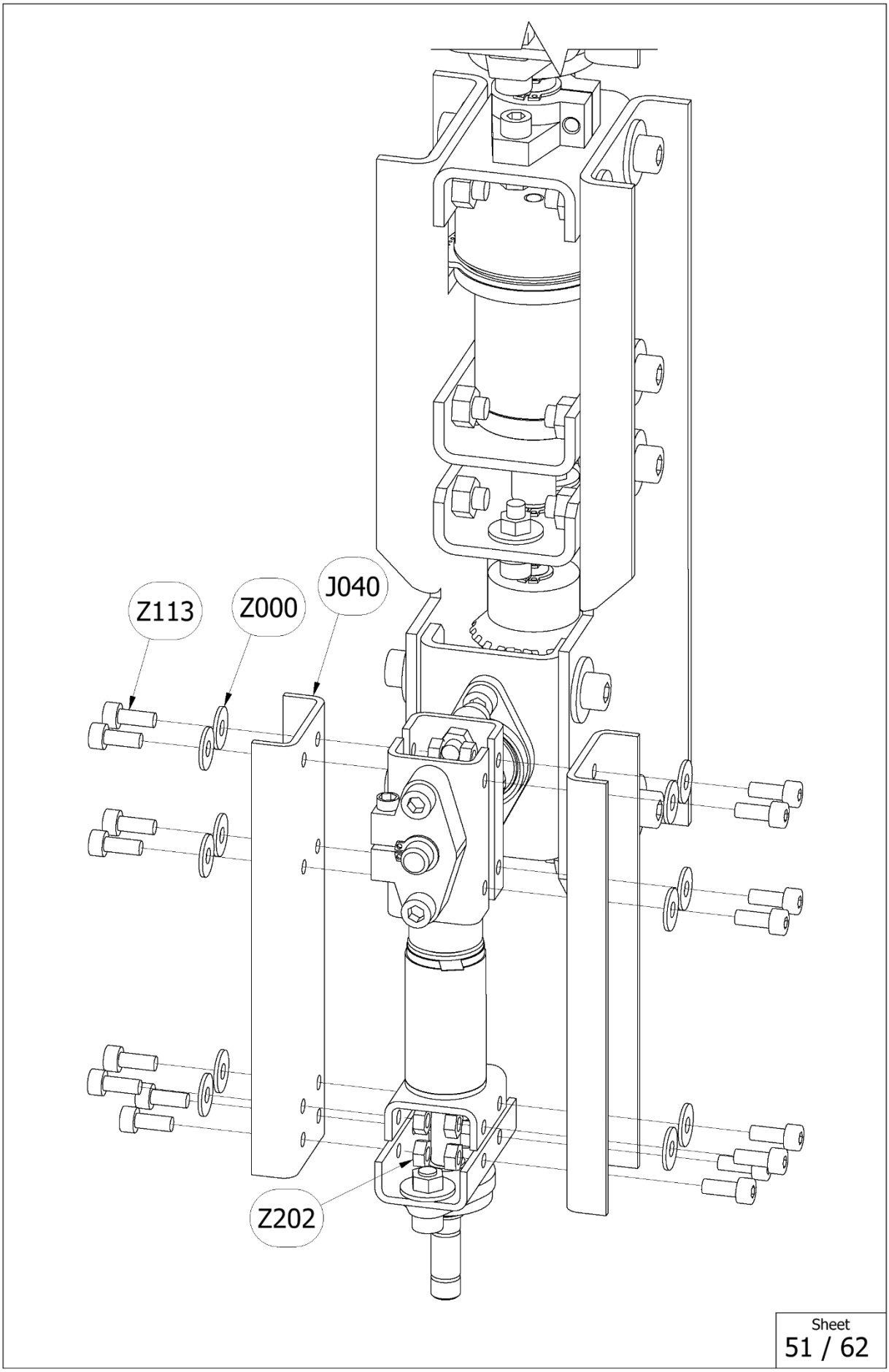

O012
Z302
K004
Z203
Z001
J032
P100
Z123
Sheet
47 / 62

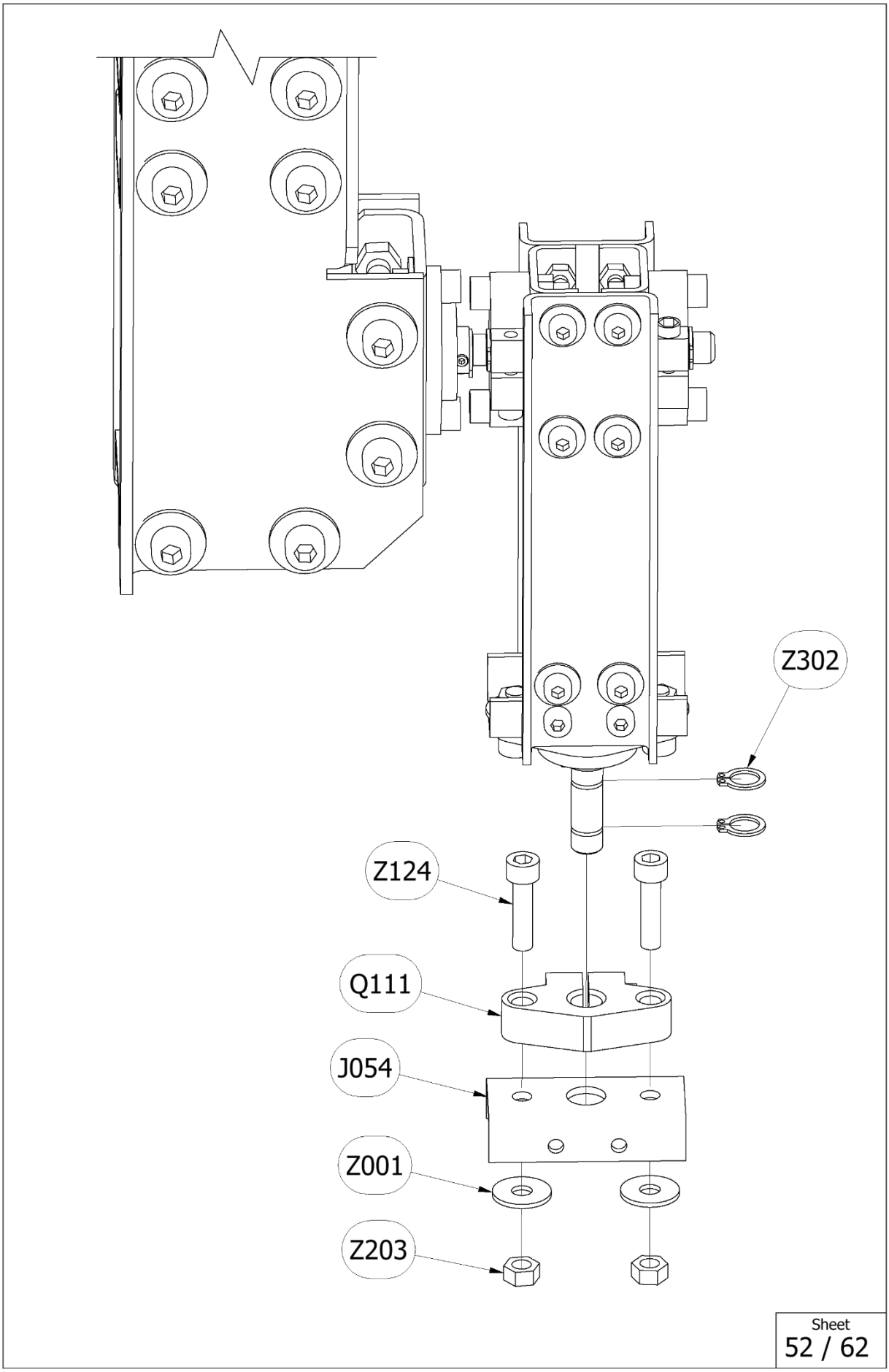

Z127
Z002
J031
Z204
J030
Sheet
48 / 62

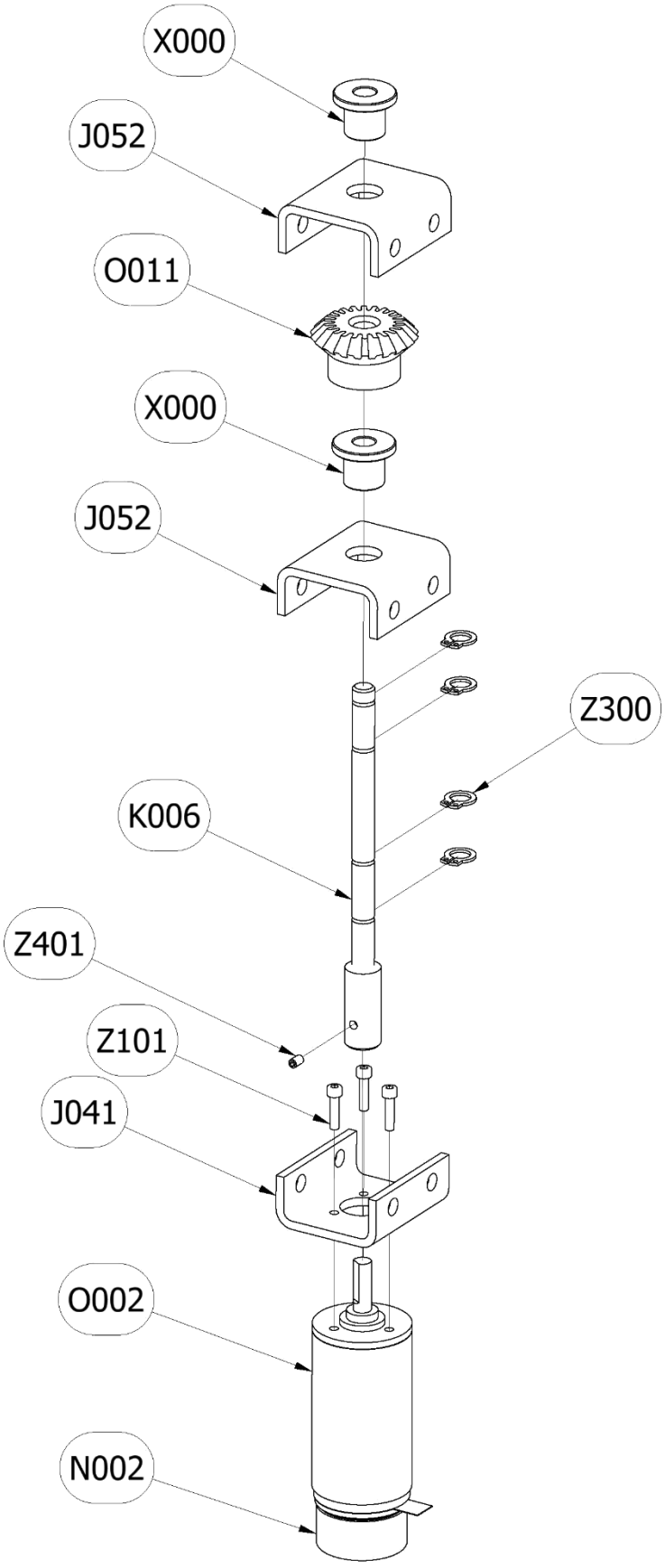


Sheet
49 / 62

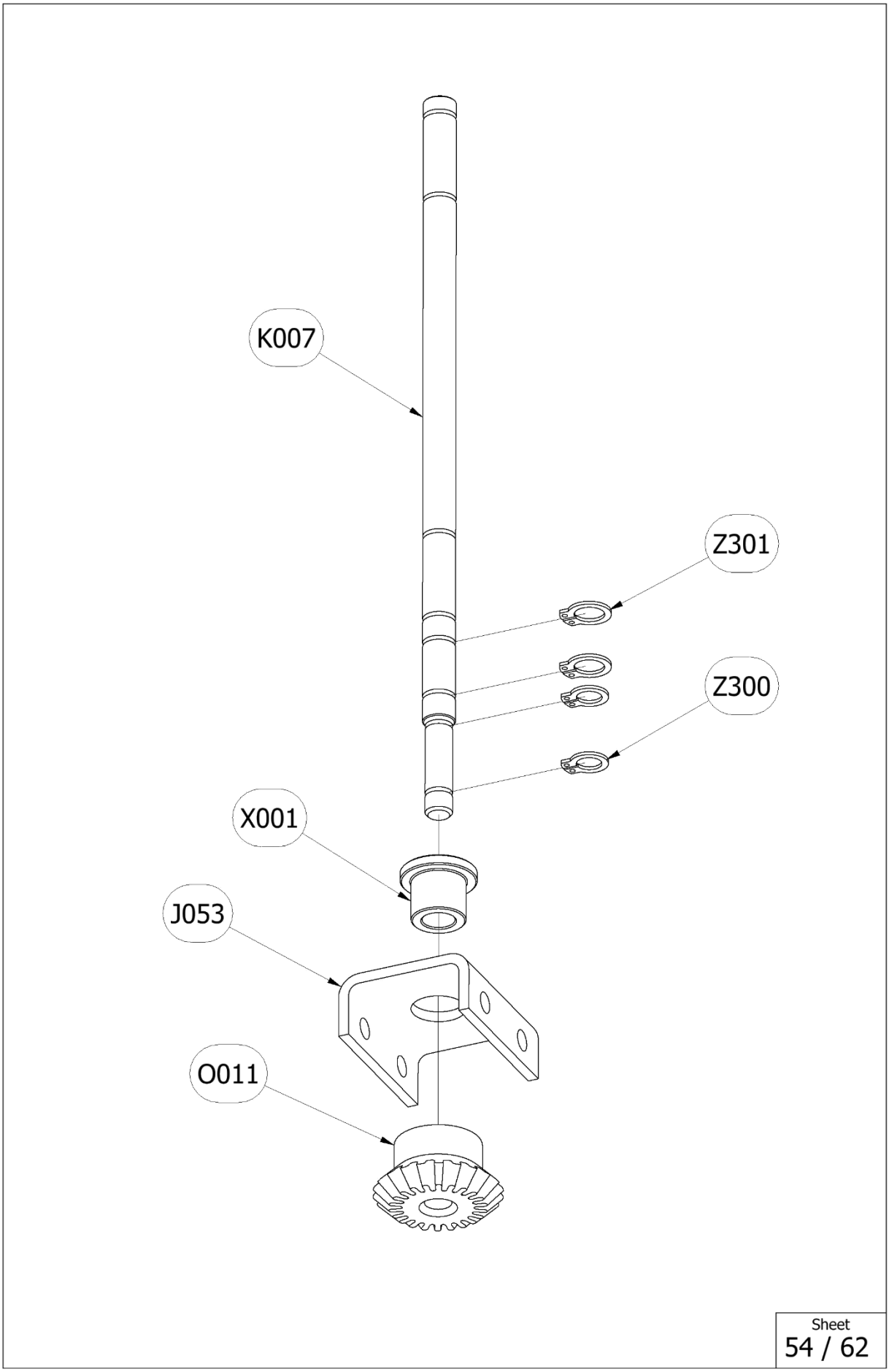

Z122
P100
J042
Z001
Z203
Z302
K005
Z401
Z101
J041
O002
N002
Sheet
50 / 62

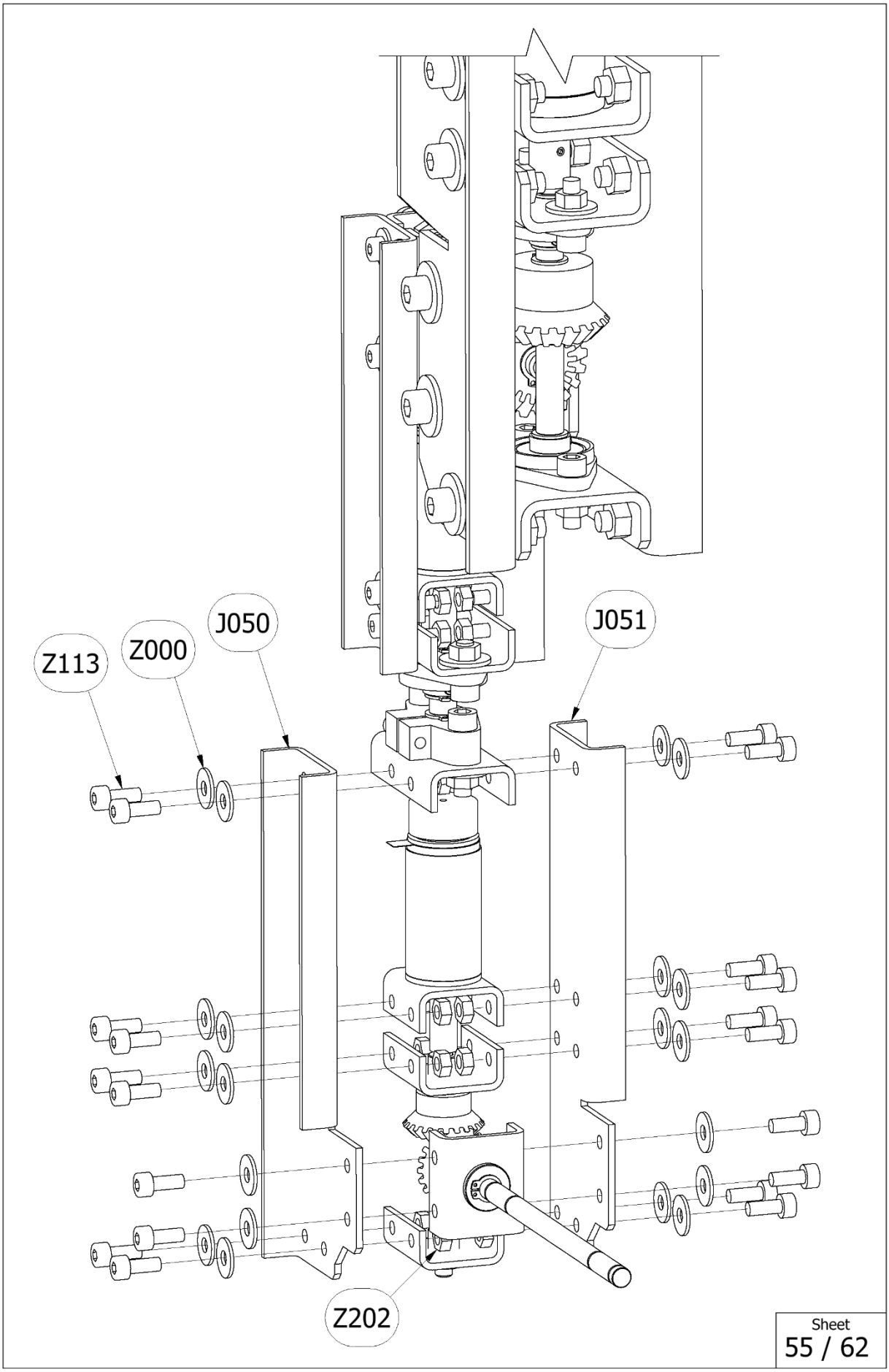

Z113
Z000
J040
Z202
Sheet
51 / 62

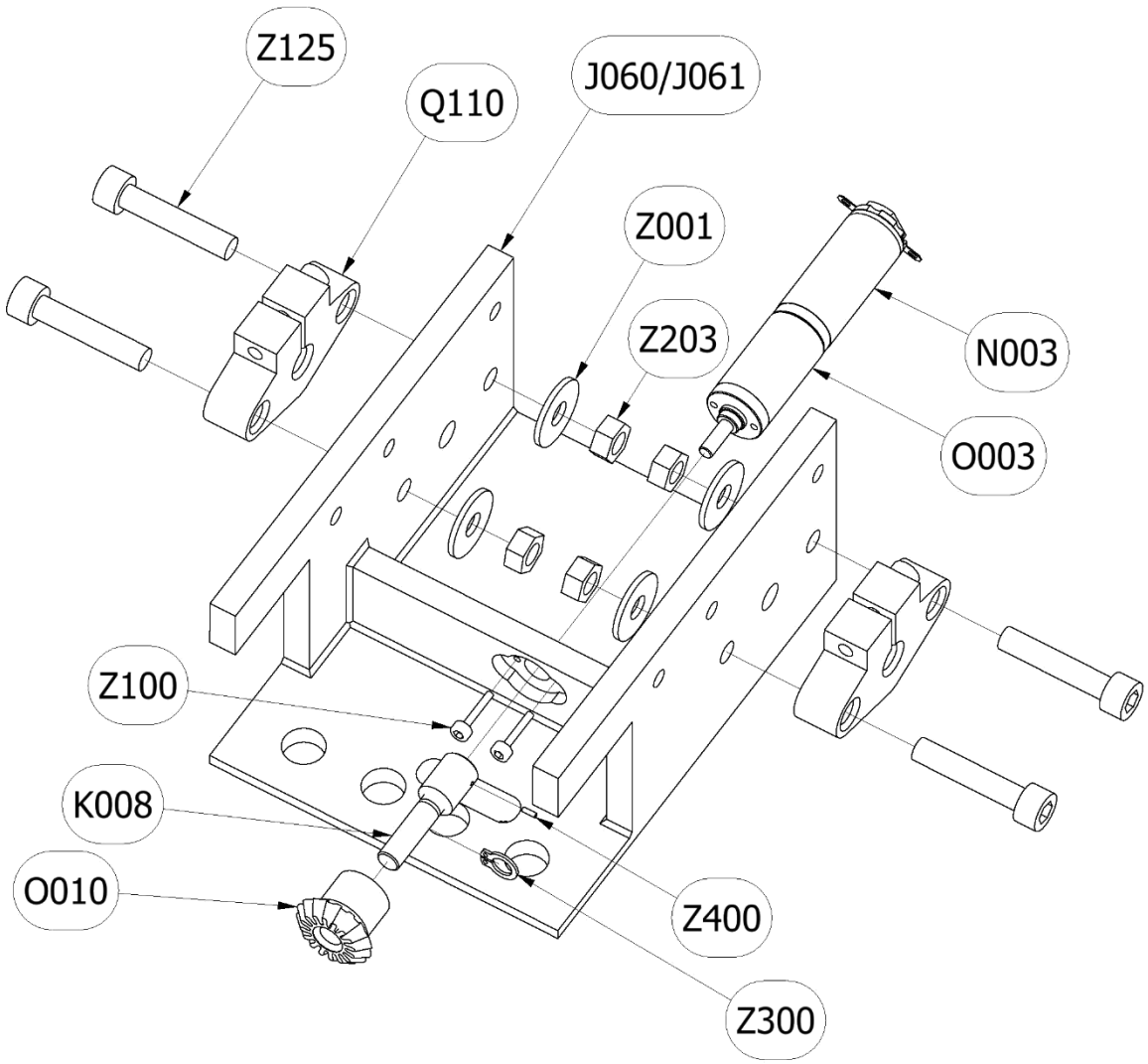

Z302
Z124
Q111
J054
Z001
Z203
Sheet
52 / 62



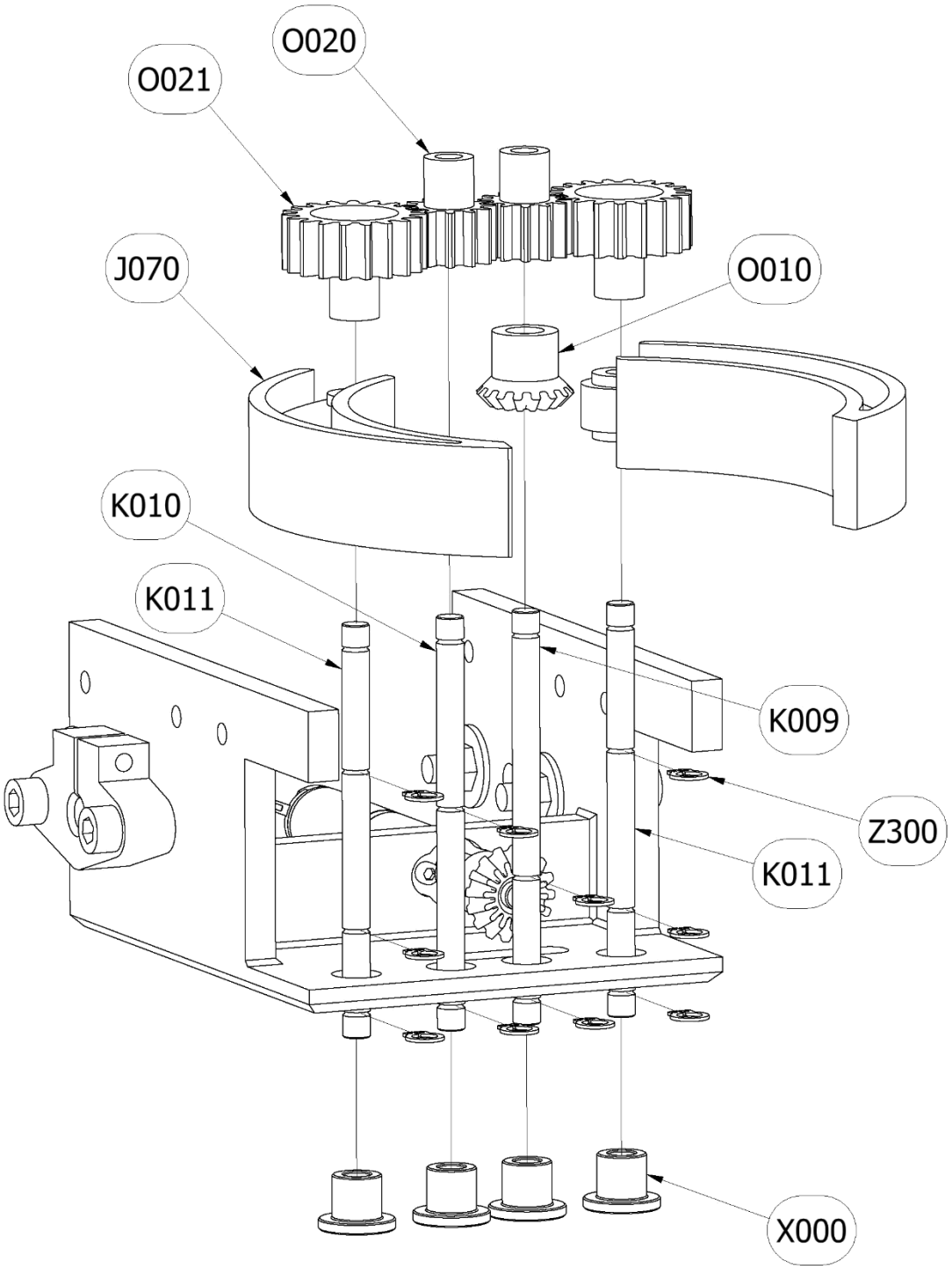
Sheet
53 / 62


K007
Z301
Z300
X001
J053
O011
Sheet
54 / 62

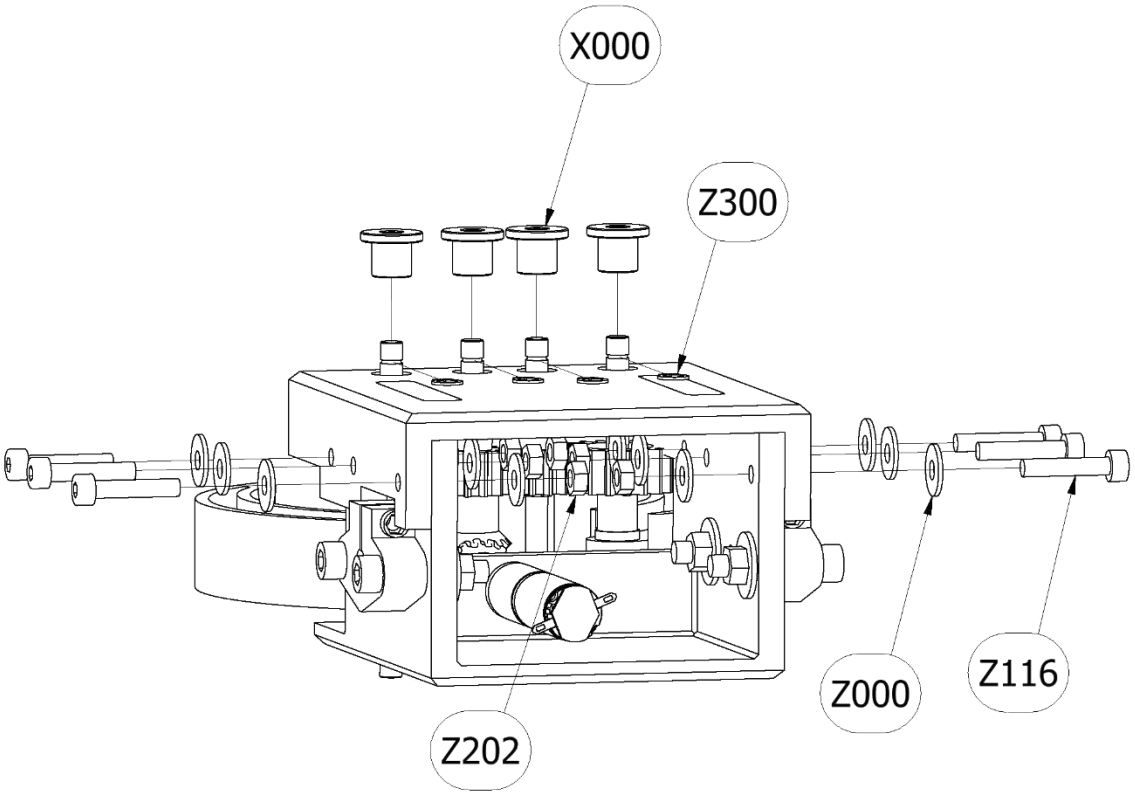

Z113
Z000
J050
J051
Z202
Sheet
55 / 62


Z125
Q110
J060/J061
Z001
Z203
N003
O003
Z100
K008
O010
Z400
Z300

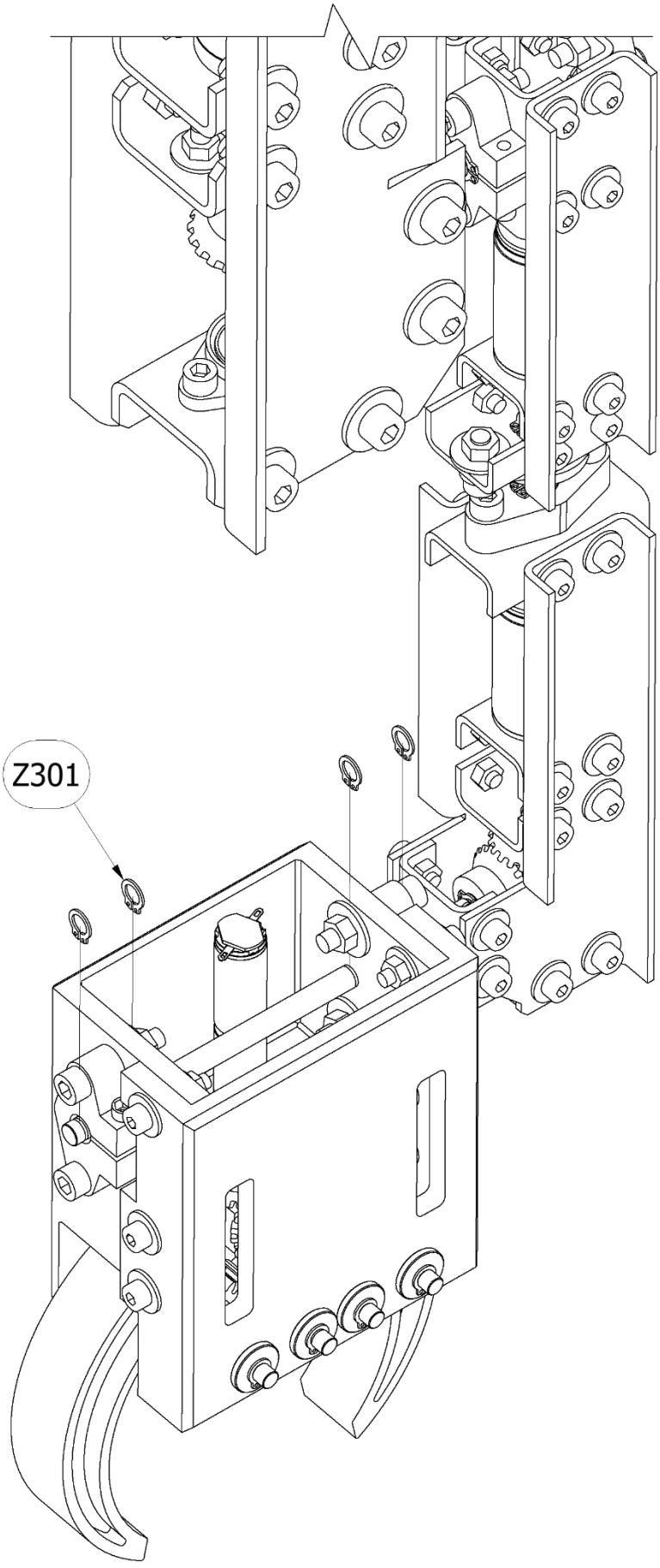
Sheet
56 / 62



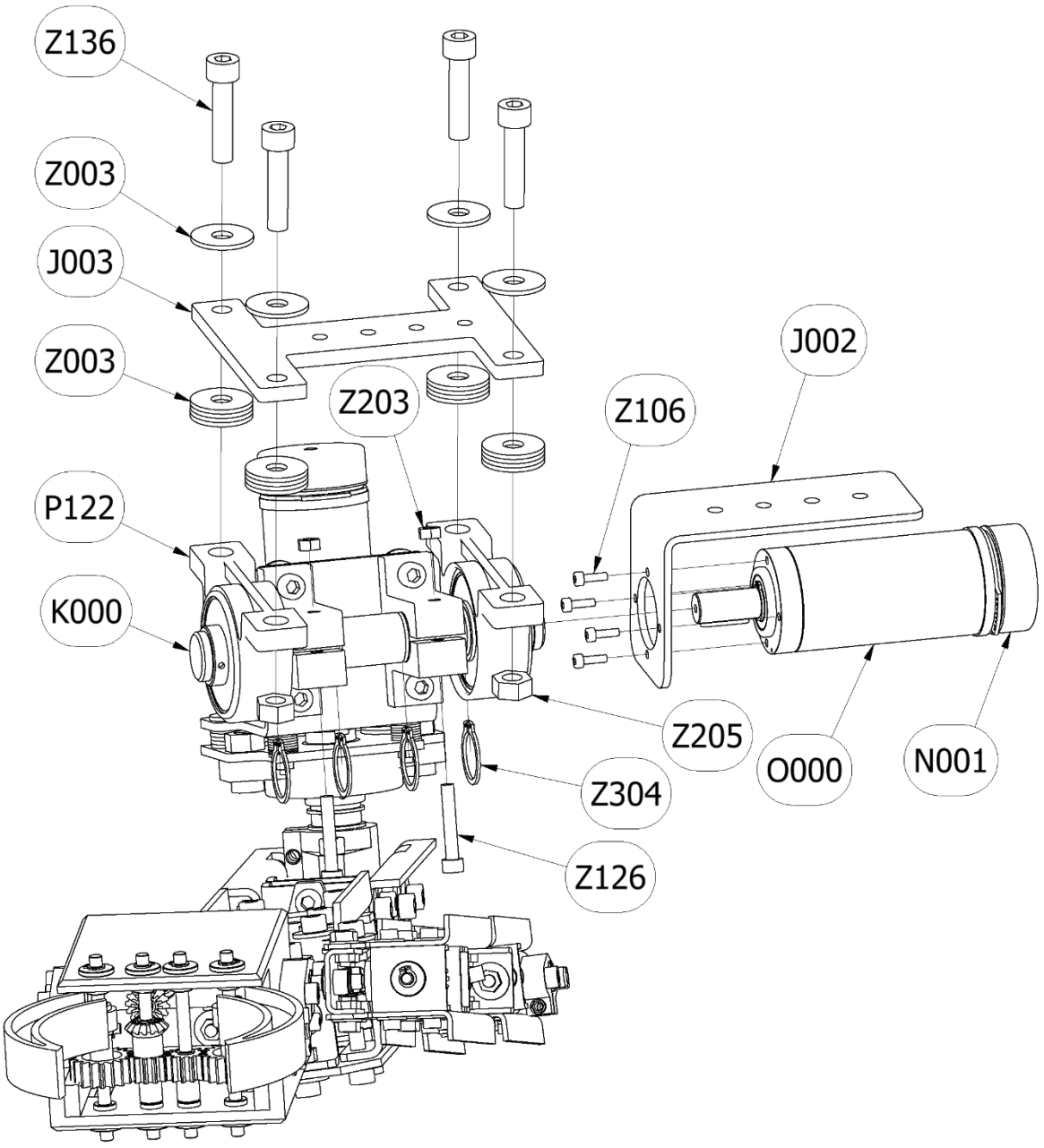
Sheet
57 / 62



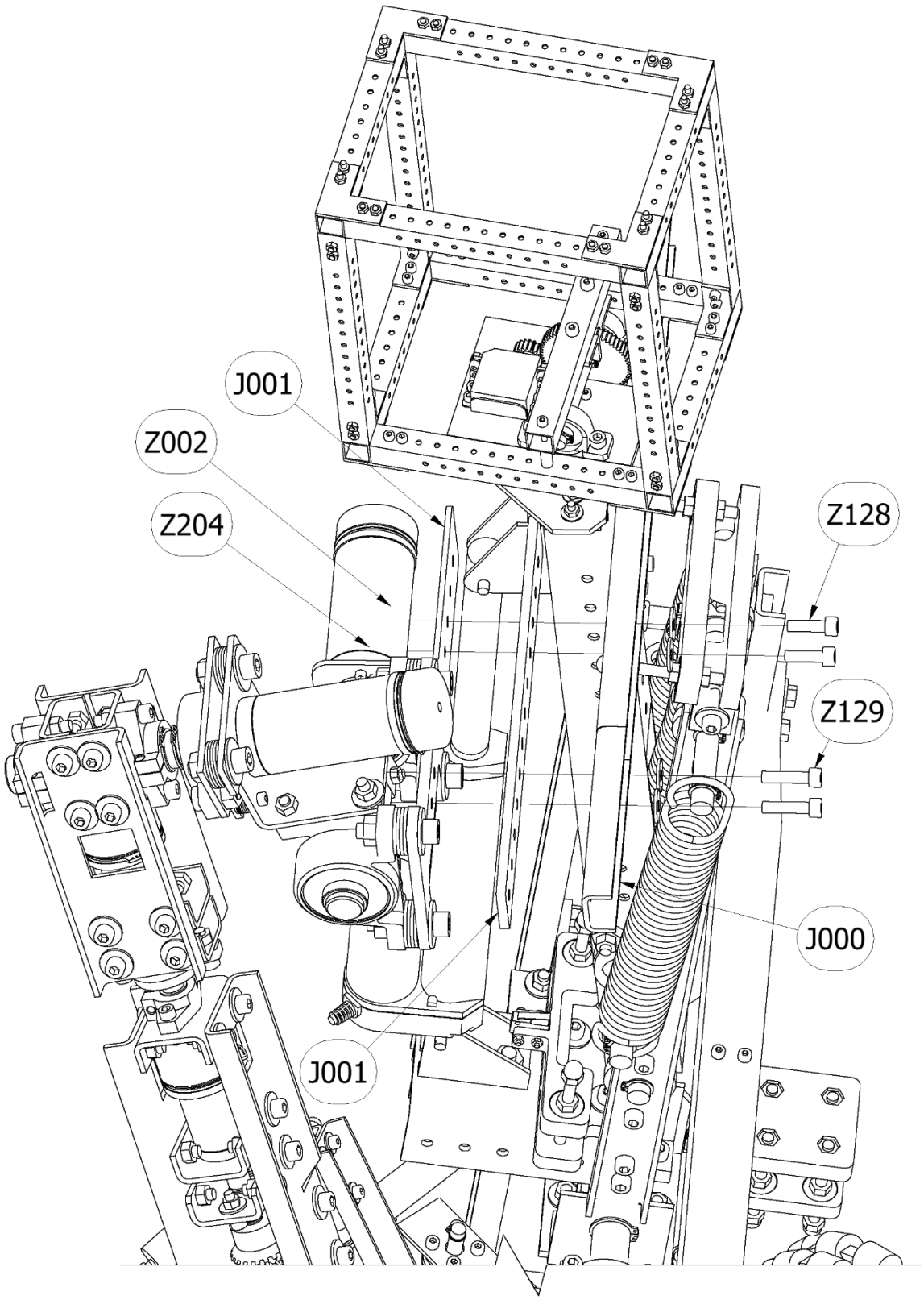
Sheet
58 / 62



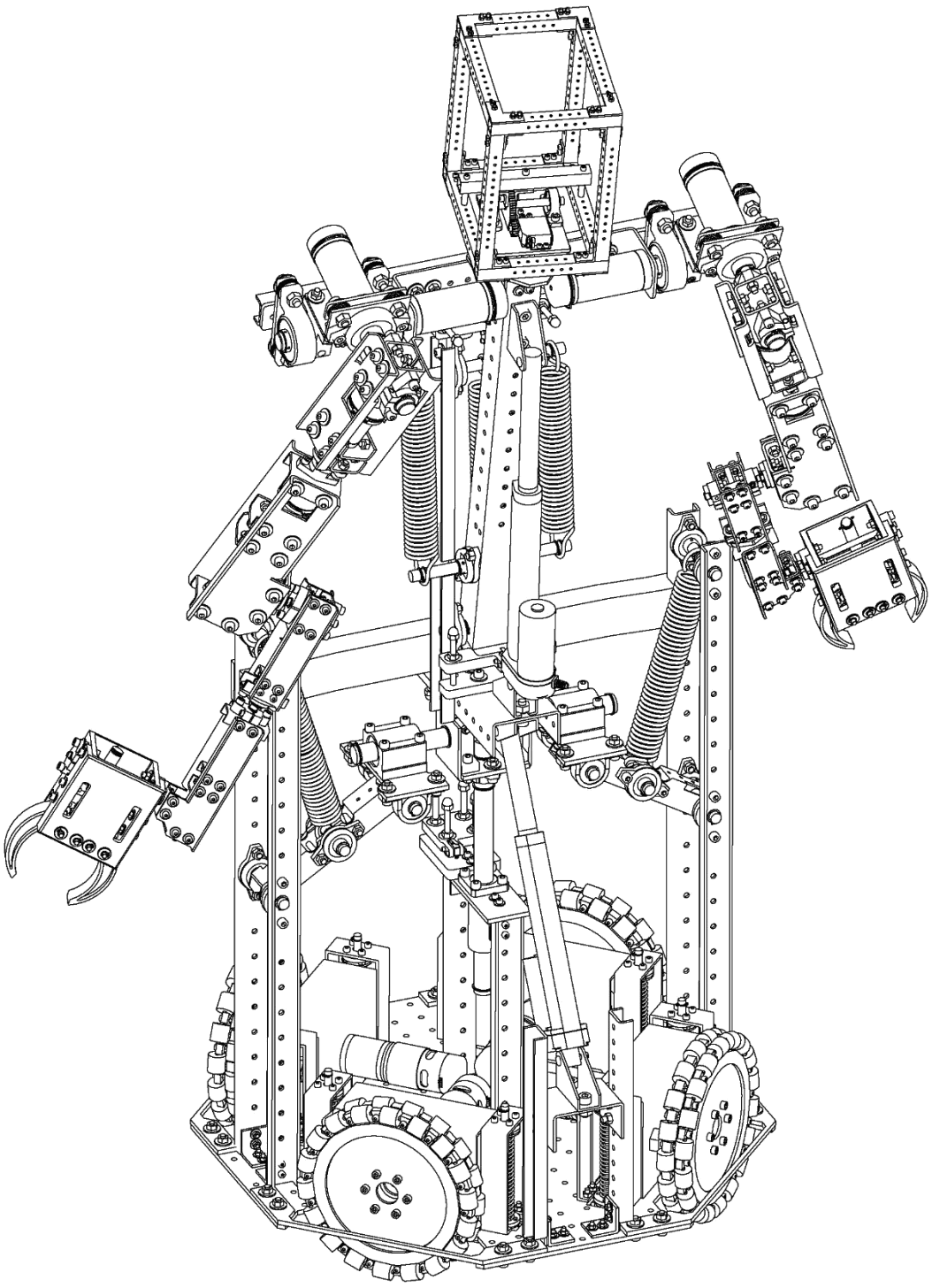
Sheet
59 / 62


Z136
Z003
J003
Z003
Z203
Z106
J002
P122
K000
Z205
O000
N001
Z304
Z126

Sheet
60 / 62



Sheet
61 / 62



Sheet
62 / 62

APPENDIX V - GEOMETRIC PROPERTIES FOR COMPUTATIONAL MODEL

Table A5.1 – Local coordinates of body 1) Base

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	29	0.0	70.0	290.0
C1	− 1.5	26.3	165.8	30	0.0	− 70.0	290.0
B2	0.0	0.0	258.0	31	0.0	− 70.0	0.0
B3	245.0	0.0	341.0	32	245.0	51.0	0.0
B4	− 245.0	0.0	341.0	33	245.0	51.0	665.0
B5	245.0	0.0	631.0	34	245.0	0.0	665.0
B6	− 245.0	0.0	631.0	35	245.0	− 51.0	665.0
B7	0,0	238.0	204,0	36	245.0	− 51.0	0.0
B8	− 129.8	129.8	5.0	37	− 245.0	51.0	0.0
B9	− 129.8	− 129.8	5.0	38	− 245.0	51.0	665.0
B10	129.8	− 129.8	5.0	39	− 245.0	0.0	665.0
B11	129.8	129.8	5.0	40	− 245.0	− 51.0	665.0
12	282.5	0.0	0.0	41	− 245.0	− 51.0	0.0
13	282.5	117.0	0.0	42	− 40.0	238.0	0.0
14	199.8	199.8	0.0	43	− 40.0	238.0	255.0
15	117.0	282.5	0.0	44	− 40.0	238.0	204.0
16	0.0	282.5	0.0	45	40.0	238.0	204.0
17	− 117.0	282.5	0.0	46	40.0	238.0	255.0
18	− 199.8	199.8	0.0	47	40.0	238.0	0.0
19	− 282.5	117.0	0.0	48	245.0	51.0	341.0
20	− 282.5	0.0	0.0	49	245.0	− 51.0	341.0
21	− 282.5	− 117.0	0.0	50	245.0	51.0	631.0
22	− 199.8	− 199.8	0.0	51	245.0	− 51.0	631.0
23	− 117.0	− 282.5	0.0	52	− 245.0	51.0	341.0
24	0.0	− 282.5	0.0	53	− 245.0	− 51.0	341.0
25	117.0	− 282.5	0.0	54	− 245.0	51.0	631.0
26	199.8	− 199.8	0.0	55	− 245.0	− 51.0	631.0
27	282.5	− 117.0	0.0	56	0.0	238.0	0.0
28	0.0	70.0	0.0	-	-	-	-

Table A5.2 – Local coordinates of bodies 1a\1d\1g\1j) Suspension system intermediary link

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	12	100.0	0,0	106.8
C1	0.0	0.0	70.2	13	85.0	− 30.0	0.0
B2	0.0	0.0	105.0	14	85.0	− 30.0	120.0
3	− 100.0	0,0	106.8	15	85.0	30.0	120.0
4	− 115.0	− 30.0	0.0	16	85.0	30.0	0.0
5	− 115.0	− 30.0	120.0	17	115.0	− 30.0	0.0
6	− 115.0	30.0	120.0	18	115.0	− 30.0	120.0
7	− 115.0	30.0	0.0	19	115.0	30.0	120.0
8	− 85.0	− 30.0	0.0	20	115.0	30.0	0.0
9	− 85.0	− 30.0	120.0	21	− 100.0	0.0	120.0
10	− 85.0	30.0	120.0	22	100.0	0.0	120.0
11	− 85.0	30.0	0.0	-	-	-	-

Table A5.3 – Local coordinates of bodies 1b\1e\1h\1k) DC motor assembly

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	13	117.0	− 15.8	− 16.0
C1	0.0	− 13.7	52.3	14	117.0	− 15.8	0.0
B2	0.0	9.1	60.0	15	117.0	15.8	0.0
3	0.0	− 157.0	60.0	16	117.0	15.8	− 16.0
4	0.0	− 30.0	60.0	17	− 117.0	− 15.8	− 16.0
5	80.0	− 30.0	0.0	18	− 117.0	− 15.8	0.0
6	80.0	30.0	0.0	19	− 117.0	15.8	0.0
7	80.0	30.0	120.0	20	− 117.0	15.8	− 16.0
8	80.0	− 30.0	120.0	21	100.0	0.0	12.2
9	− 80.0	− 30.0	0.0	22	− 100.0	0.0	12.2
10	− 80.0	30.0	0.0	23	100.0	0.0	0.0
11	− 80.0	30.0	120.0	24	− 100.0	0.0	0.0
12	− 80.0	− 30.0	120.0	-	-	-	-

Table A5.4 – Local coordinates of bodies 1c\1f\1i\1l) Swedish omni wheel

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	23	101.5	41.1	0.0
C1	0.0	50.6	0.0	24	101.5	70.7	0.0
2	0.0	− 39.9	0.0	25	95.4	70.7	34.7
3	0.0	41.1	0.0	26	77.8	70.7	65.2
4	0.0	70.7	0.0	27	50.8	70.7	87.9
5	101.5	41.1	0.0	28	17.6	70.7	100.0
6	95.4	41.1	34.7	29	− 17.6	70.7	100.0
7	77.8	41.1	65.2	30	− 50.7	70.7	87.9
8	50.8	41.1	87.9	31	− 77.8	70.7	65.2
9	17.6	41.1	100.0	32	− 95.4	70.7	34.7
10	− 17.6	41.1	100.0	33	− 101.5	70.7	0.0
11	− 50.7	41.1	87.9	34	− 95.4	70.7	-34.7
12	− 77.8	41.1	65.2	35	− 77.8	70.7	-65.2
13	− 95.4	41.1	34.7	36	− 50.8	70.7	-87.9
14	− 101.5	41.1	0.0	37	− 17.6	70.7	− 100.0
15	− 95.4	41.1	− 34.7	38	17.6	70.7	− 100.0
16	− 77.8	41.1	− 65.2	39	50.8	70.7	− 87.9
17	− 50.8	41.1	− 87.9	40	77.8	70.7	− 65.2
18	− 17.6	41.1	− 100.0	41	95.4	70.7	− 34.7
19	17.6	41.1	− 100.0	42	101.5	70.7	0.0
20	50.8	41.1	− 87.9	43	0.0	111.8	0.0
21	77.8	41.1	− 65.2	44	0.0	55.9	0.0
22	95.4	41.1	− 34.7	-	-	-	-

Table A5.5 – Local coordinates of body 2) Hip

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	17	− 54.5	0.0	743.3
C1	0.0	18.0	348.5	18	38.5	0.0	345.2
B2	0.0	0.0	398.2	19	− 38.5	0.0	345.2
B3	0.0	0.0	708.2	20	38.5	66.2	345.2
B4	0.0	105.3	381.2	21	38.5	66.2	306.2
5	0.0	0.0	272.2	22	0.0	66.2	306.2
6	162.5	0.0	272.2	23	− 38.5	66.2	306.2
7	54.5	0.0	272.2	24	− 38.5	66.2	345.2
8	− 54.5	0.0	272.2	25	38.5	105.2	345.2
9	− 162.5	0.0	272.2	26	38.5	105.2	381.2
10	54.5	0.0	345.2	27	− 38.5	105.2	381.2
11	− 54.5	0.0	345.2	28	− 38.5	105.2	345.2
12	54.5	0.0	398.2	29	38.5	116.0	345.2
13	− 54.5	0.0	398.2	30	− 38.5	116.0	345.2
14	54.5	0.0	708.2	31	0.0	66.2	345.2
15	− 54.5	0.0	708.2	32	0.0	105.2	345.2
16	54.5	0.0	743.3	-	-	-	-

Table A5.6 – Local coordinates of body 3) Torso

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	6	0.0	0.0	85.0
C1	0.0	− 5.7	338.5	7	0.0	0.0	397.2
B2	110.0	− 20.0	460.0	8	0.0	56.0	397.2
B3	− 110.0	− 20.0	460.0	9	0.0	0.0	460.0
B4	0.0	0.0	495.0	10	0.0	− 20.0	460.0
5	0.0	0.0	− 15.0	-	-	-	-

Table A5.7 – Local coordinates of body 4a) Left scapula

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	− 46.2	0.0	100.2
C1	− 28.0	0.0	56.0	-	-	-	-

Table A5.8 – Local coordinates of body 5a) Left shoulder

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	− 120.3	66.5	0.0
C1	− 28.6	60.7	0.0	-	-	-	-

Table A5.9 – Local coordinates of body 6a) Left proximal arm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	0.0	0.0	150.8
C1	0.0	0.0	54.3	-	-	-	-

Table A5.10 – Local coordinates of body 7a) Left distal arm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	− 89.8	0.0	170.4
C1	− 6.0	0.0	115.4	-	-	-	-

Table A5.11 – Local coordinates of body 8a) Left proximal forearm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	0.0	0.0	130.6
C1	0.0	− 0.8	54.5	-	-	-	-

Table A5.12 – Local coordinates of body 9a) Left distal forearm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	− 93.8	0.0	126.6
C1	− 8.6	− 0.3	80.8	-	-	-	-

Table A5.13 – Local coordinates of body 10a) Left hand

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	4	0.0	5.9	116.4
C1	− 1.1	5.5	29.9	5	0.0	5.9	0.0
B2	26.3	5.9	68.1	6	116.4	5.9	0.0
B3	− 26.3	5.9	68.1	-	-	-	-

Table A5.14 – Local coordinates of body 11a) Left thumb claw

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	17	− 10.0	− 10.0	0.0
C1	− 34.7	0.0	26.4	18	0.0	10.0	14.2
B2	35.2	0.0	42.3	19	9.8	10.0	42.3
3	0.0	− 10.0	14.2	20	0.0	10.0	70.4
4	9.8	− 10.0	42.3	21	− 10.0	10.0	79.6
5	0.0	− 10.0	70.4	22	− 20.7	10.0	85.1
6	− 10.0	− 10.0	79.6	23	− 20.7	10.0	90.4
7	− 20.7	− 10.0	85.1	24	− 10.0	10.0	88.8
8	− 20.7	− 10.0	90.4	25	0.0	10.0	84.6
9	− 10.0	− 10.0	88.8	26	9.8	10.0	77.8
10	0.0	− 10.0	84.6	27	20.0	10.0	66.0
11	9.8	− 10.0	77.8	28	28.7	10.0	36.4
12	20.0	− 10.0	66.0	29	20.0	10.0	6.9
13	28.7	− 10.0	36.4	30	9.8	10.0	− 5.0
14	20.0	− 10.0	6.9	31	0.0	10.0	− 10.0
15	9.8	− 10.0	− 5.0	32	− 10.0	10.0	0.0
16	0.0	− 10.0	− 10.0	-	-	-	-

Table A5.15 – Local coordinates of body 11d) Left opposite claw

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	17	10.0	− 10.0	0.0
C1	34.7	0.0	26.4	18	0.0	10.0	14.2
B2	− 35.2	0.0	42.3	19	− 9.8	10.0	42.3
3	0.0	− 10.0	14.2	20	0.0	10.0	70.4
4	− 9.8	− 10.0	42.3	21	10.0	10.0	79.6
5	0.0	− 10.0	70.4	22	20.7	10.0	85.1
6	10.0	− 10.0	79.6	23	20.7	10.0	90.4
7	20.7	− 10.0	85.1	24	10.0	10.0	88.8
8	20.7	− 10.0	90.4	25	0.0	10.0	84.6
9	10.0	− 10.0	88.8	26	− 9.8	10.0	77.8
10	0.0	− 10.0	84.6	27	− 20.0	10.0	66.0
11	− 9.8	− 10.0	77.8	28	− 28.7	10.0	36.4
12	− 20.0	− 10.0	66.0	29	− 20.0	10.0	6.9
13	− 28.7	− 10.0	36.4	30	− 9.8	10.0	− 5.0
14	− 20.0	− 10.0	6.9	31	0.0	10.0	− 10.0
15	− 9.8	− 10.0	− 5.0	32	10.0	10.0	0.0
16	0.0	− 10.0	− 10.0	-	-	-	-

Table A5.16 – Local coordinates of body 4b) Right scapula

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	46.2	0.0	100.2
C1	28.0	0.0	56.0	-	-	-	-

Table A5.17 – Local coordinates of body 5b) Right shoulder

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	120.3	66.5	0.0
C1	28.6	60.7	0.0	-	-	-	-

Table A5.18 – Local coordinates of body 6b) Right proximal arm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	0.0	0.0	150.8
C1	0.0	0.0	54.3	-	-	-	-

Table A5.19 – Local coordinates of body 7b) Right distal arm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	89.8	0.0	170.4
C1	6.0	0.0	115.4	-	-	-	-

Table A5.20 – Local coordinates of body 8b) Right proximal forearm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	0.0	0.0	130.6
C1	0.0	− 0.8	54.5	-	-	-	-

Table A5.21 – Local coordinates of body 9b) Right distal forearm

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	B2	93.8	0.0	126.6
C1	8.6	− 0.3	80.8	-	-	-	-

Table A5.22 – Local coordinates of body 10b) Right hand

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	4	0.0	5.9	116.4
C1	1.1	5.5	29.9	5	0.0	5.9	0.0
B2	− 26.3	5.9	68.1	6	− 116.4	5.9	0.0
B3	26.3	5.9	68.1	-	-	-	-

Table A5.23 – Local coordinates of body 11b) Right thumb claw

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	17	10.0	− 10.0	0.0
C1	34.7	0.0	26.4	18	0.0	10.0	14.2
B2	− 35.2	0.0	42.3	19	− 9.8	10.0	42.3
3	0.0	− 10.0	14.2	20	0.0	10.0	70.4
4	− 9.8	− 10.0	42.3	21	10.0	10.0	79.6
5	0.0	− 10.0	70.4	22	20.7	10.0	85.1
6	10.0	− 10.0	79.6	23	20.7	10.0	90.4
7	20.7	− 10.0	85.1	24	10.0	10.0	88.8
8	20.7	− 10.0	90.4	25	0.0	10.0	84.6
9	10.0	− 10.0	88.8	26	− 9.8	10.0	77.8
10	0.0	− 10.0	84.6	27	− 20.0	10.0	66.0
11	− 9.8	− 10.0	77.8	28	− 28.7	10.0	36.4
12	− 20.0	− 10.0	66.0	29	− 20.0	10.0	6.9
13	− 28.7	− 10.0	36.4	30	− 9.8	10.0	− 5.0
14	− 20.0	− 10.0	6.9	31	0.0	10.0	− 10.0
15	− 9.8	− 10.0	− 5.0	32	10.0	10.0	0.0
16	0.0	− 10.0	− 10.0	-	-	-	-

Table A5.24 – Local coordinates of body 11e) Right opposite claw

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	17	− 10.0	− 10.0	0.0
C1	− 34.7	0.0	26.4	18	0.0	10.0	14.2
B2	35.2	0.0	42.3	19	9.8	10.0	42.3
3	0.0	− 10.0	14.2	20	0.0	10.0	70.4
4	9.8	− 10.0	42.3	21	− 10.0	10.0	79.6
5	0.0	− 10.0	70.4	22	− 20.7	10.0	85.1
6	− 10.0	− 10.0	79.6	23	− 20.7	10.0	90.4
7	− 20.7	− 10.0	85.1	24	− 10.0	10.0	88.8
8	− 20.7	− 10.0	90.4	25	0.0	10.0	84.6
9	− 10.0	− 10.0	88.8	26	9.8	10.0	77.8
10	0.0	− 10.0	84.6	27	20.0	10.0	66.0
11	9.8	− 10.0	77.8	28	28.7	10.0	36.4
12	20.0	− 10.0	66.0	29	20.0	10.0	6.9
13	28.7	− 10.0	36.4	30	9.8	10.0	− 5.0
14	20.0	− 10.0	6.9	31	0.0	10.0	− 10.0
15	9.8	− 10.0	− 5.0	32	− 10.0	10.0	0.0
16	0.0	− 10.0	− 10.0	-	-	-	-

Table A5.25 – Local coordinates of body 4c) Lower cervical spine

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	11	6.9	22.7	7.0
C1	0.0	− 1.3	5.1	12	6.9	− 7.6	7.0
B2	0.0	0.0	46.3	13	− 6.9	− 7.6	7.0
3	20.0	45.0	0.0	14	− 6.9	22.7	7.0
4	45.0	20.0	0.0	15	6.9	22.7	37.2
5	45.0	− 20.0	0.0	16	6.9	− 7.6	37.2
6	20.0	− 45.0	0.0	17	− 6.9	− 7.6	37.2
7	− 20.0	− 45.0	0.0	18	− 6.9	22.7	37.2
8	− 45.0	− 20.0	0.0	19	0.0	0.0	37.2
9	− 45.0	20.0	0.0	20	0.0	0.0	23.2
10	− 20.0	45.0	0.0	-	-	-	-

Table A5.26 – Local coordinates of body 5c) Neck

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
A0	0.0	0.0	0.0	12	32.5	− 5.3	0.0
C1	3.0	3.8	10.8	13	− 24.5	18.5	4.0
B2	14.7	− 5.3	18.3	14	− 24.5	48.8	4.0
3	− 40.0	− 57.5	0.0	15	7.7	48.8	4.0
4	− 40.0	42.5	0.0	16	7.7	18.5	4.0
5	40.0	42.5	0.0	17	− 24.5	18.5	18.8
6	40.0	− 57.5	0.0	18	− 24.5	48.8	18.8
7	− 32.5	− 5.3	0.0	19	7.7	48.8	18.8
8	− 44.5	− 5.3	18.3	20	7.7	18.5	18.8
9	− 32.5	− 5.3	18.3	21	7.7	26.1	11.4
10	32.5	− 5.3	18.3	22	12.7	26.1	11.4
11	44.5	− 5.3	18.3	23	0.0	0.0	9.2

Table A5.27 – Local coordinates of body 6c) Head

Point	Local Coordinates [mm]			Point	Local Coordinates [mm]		
	x	y	z		x	y	z
0	0.0	0.0	0.0	8	55.3	− 70.0	− 9.0
C1	− 11.8	15.2	75.7	9	− 84.8	− 70.0	− 9.0
2	0.0	0.0	41.0	10	− 84.8	110.0	− 9.0
3	55.3	0.0	41.0	11	55.3	110.0	201.0
4	− 84.8	0.0	41.0	12	55.3	− 70.0	201.0
5	55.3	0.0	− 9.0	13	− 84.8	− 70.0	201.0
6	− 84.8	0.0	− 9.0	14	− 84.8	110.0	201.0
7	55.3	110.0	− 9.0	15	− 14.8	20.0	96.0

ANNEX I - DATABASE OF TENSION SPRINGS

Table a1.1 – Properties of the compression springs used for CHARMIE's suspension system

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
0	300.0	300.0	1310.0	2.3	0.2	1.5	AISI 302 Stainless Steel
1	200.0	200.0	666.7	7.7	0.3	0.2	AISI 302 Stainless Steel
2	20.5	27.5	67.2	3.8	0.3	20.3	AISI 302 Stainless Steel
3	5.0	9.5	16.4	2.2	0.3	430.5	AISI 302 Stainless Steel
4	7.0	11.0	23.0	1.9	0.3	477.9	AISI 302 Stainless Steel
5	300.0	300.0	983.6	2.2	0.3	7.2	AISI 302 Stainless Steel
6	13.0	19.5	36.5	3.9	0.4	61.8	AISI 302 Stainless Steel
7	16.5	20.5	46.3	2.1	0.4	303.3	AISI 302 Stainless Steel
8	23.0	28.0	64.4	2.6	0.4	117.8	AISI 302 Stainless Steel
9	200.0	200.0	500.0	9.6	0.4	0.5	AISI 302 Stainless Steel
10	320.0	320.0	800.0	2.6	0.4	15.7	AISI 302 Stainless Steel
11	20.0	26.5	49.3	3.4	0.4	121.7	AISI 302 Stainless Steel
12	11.0	16.0	27.1	2.6	0.4	495.5	AISI 302 Stainless Steel
13	12.5	19.0	30.8	3.9	0.4	128.9	AISI 302 Stainless Steel
14	20.0	25.0	49.3	2.7	0.4	243.3	AISI 302 Stainless Steel
15	226.0	232.0	556.7	3.3	0.4	11.8	AISI 302 Stainless Steel
16	151.0	157.0	371.9	2.5	0.4	40.6	AISI 302 Stainless Steel
17	18.5	24.5	40.5	3.7	0.5	177.2	AISI 302 Stainless Steel
18	200.0	200.0	400.0	11.5	0.5	0.9	AISI 302 Stainless Steel
19	20.0	33.0	25.0	5.2	0.8	1005.0	AISI 302 Stainless Steel
20	320.0	320.0	640.0	3.5	0.5	19.6	AISI 302 Stainless Steel
21	21.0	30.0	42.0	4.5	0.5	140.8	AISI 302 Stainless Steel
22	19.5	32.0	38.4	8.0	0.5	29.3	AISI 302 Stainless Steel
23	300.0	300.0	590.6	2.4	0.5	71.1	AISI 302 Stainless Steel
24	7.5	13.5	14.8	3.3	0.5	1090.5	AISI 302 Stainless Steel
25	180.0	190.0	354.3	5.2	0.5	11.6	AISI 302 Stainless Steel
26	39.5	46.5	77.8	4.0	0.5	116.1	AISI 302 Stainless Steel
27	54.0	61.0	106.3	3.8	0.5	99.1	AISI 302 Stainless Steel
28	44.0	51.5	86.6	3.9	0.5	112.5	AISI 302 Stainless Steel
29	8.0	15.0	15.7	3.6	0.5	787.0	AISI 302 Stainless Steel
30	17.0	24.0	33.5	3.9	0.5	291.1	AISI 302 Stainless Steel
31	9.5	17.0	18.7	3.8	0.5	563.3	AISI 302 Stainless Steel
32	26.5	33.5	52.2	4.0	0.5	173.1	AISI 302 Stainless Steel
33	145.0	154.0	259.4	4.5	0.6	34.7	AISI 302 Stainless Steel
34	21.0	28.0	37.6	3.8	0.6	395.6	AISI 302 Stainless Steel
35	320.0	320.0	533.3	7.4	0.6	5.2	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
36	18.5	25.0	29.1	3.4	0.6	1263.3	AISI 302 Stainless Steel
37	56.0	65.0	88.2	4.9	0.6	138.1	AISI 302 Stainless Steel
38	16.5	24.0	26.0	4.0	0.6	865.8	AISI 302 Stainless Steel
39	112.0	124.0	176.4	5.8	0.6	41.5	AISI 302 Stainless Steel
40	290.0	290.0	456.7	3.6	0.6	67.8	AISI 302 Stainless Steel
41	35.0	43.5	55.1	4.5	0.6	285.8	AISI 302 Stainless Steel
42	55.0	65.5	86.6	6.1	0.6	72.6	AISI 302 Stainless Steel
43	26.5	34.0	41.7	3.9	0.6	582.0	AISI 302 Stainless Steel
44	190.0	200.0	299.2	3.9	0.6	81.2	AISI 302 Stainless Steel
45	65.0	73.5	102.4	4.4	0.6	164.7	AISI 302 Stainless Steel
46	60.0	73.5	94.5	8.0	0.6	29.4	AISI 302 Stainless Steel
47	23.0	36.5	36.2	6.9	0.6	119.7	AISI 302 Stainless Steel
48	38.0	47.0	59.8	4.9	0.6	203.5	AISI 302 Stainless Steel
49	320.0	320.0	457.1	4.3	0.7	57.0	AISI 302 Stainless Steel
50	34.5	45.0	49.3	5.3	0.7	282.2	AISI 302 Stainless Steel
51	29.0	40.0	41.4	5.3	0.7	335.8	AISI 302 Stainless Steel
52	39.5	54.0	55.6	8.5	0.7	64.9	AISI 302 Stainless Steel
53	143.0	158.0	201.1	7.8	0.7	23.2	AISI 302 Stainless Steel
54	80.5	92.0	113.2	6.9	0.7	59.5	AISI 302 Stainless Steel
55	16.5	34.0	23.2	8.1	0.7	179.4	AISI 302 Stainless Steel
56	13.0	22.0	18.3	4.4	0.7	1425.8	AISI 302 Stainless Steel
57	79.5	95.5	111.8	7.8	0.7	41.7	AISI 302 Stainless Steel
58	56.0	74.0	78.8	8.8	0.7	41.2	AISI 302 Stainless Steel
59	35.5	51.0	49.9	7.6	0.7	101.0	AISI 302 Stainless Steel
60	84.0	92.5	118.1	4.0	0.7	293.9	AISI 302 Stainless Steel
61	21.5	29.5	30.2	4.0	0.7	1148.4	AISI 302 Stainless Steel
62	46.5	58.5	65.4	6.1	0.7	149.3	AISI 302 Stainless Steel
63	54.5	63.5	76.7	5.5	0.7	173.9	AISI 302 Stainless Steel
64	23.0	33.0	32.3	4.8	0.7	620.4	AISI 302 Stainless Steel
65	178.0	193.0	250.4	7.5	0.7	21.0	AISI 302 Stainless Steel
66	300.0	300.0	421.9	2.5	0.7	338.8	AISI 302 Stainless Steel
67	300.0	300.0	421.9	4.5	0.7	57.7	AISI 302 Stainless Steel
68	320.0	320.0	400.0	6.2	0.8	37.1	AISI 302 Stainless Steel
69	38.0	50.0	47.5	6.2	0.8	312.1	AISI 302 Stainless Steel
70	13.0	24.5	16.0	6.4	0.8	904.4	AISI 302 Stainless Steel
71	28.5	38.0	35.1	5.2	0.8	770.2	AISI 302 Stainless Steel
72	200.0	214.0	246.0	7.8	0.8	32.4	AISI 302 Stainless Steel
73	300.0	300.0	369.0	6.3	0.8	41.1	AISI 302 Stainless Steel
74	26.0	40.0	32.0	9.2	0.8	152.0	AISI 302 Stainless Steel
75	42.0	52.0	51.7	5.2	0.8	522.7	AISI 302 Stainless Steel
76	37.5	51.0	46.1	7.3	0.8	211.1	AISI 302 Stainless Steel
77	50.5	64.5	62.1	7.4	0.8	150.5	AISI 302 Stainless Steel
78	18.5	31.5	22.8	6.7	0.8	553.8	AISI 302 Stainless Steel
79	45.0	56.5	55.4	5.7	0.8	370.1	AISI 302 Stainless Steel
80	40.0	76.0	45.0	8.1	0.9	224.4	AISI 302 Stainless Steel
81	208.0	226.0	234.0	10.5	0.9	19.8	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
82	183.0	205.0	205.8	10.1	0.9	25.3	AISI 302 Stainless Steel
83	238.0	255.0	267.7	8.7	0.9	30.4	AISI 302 Stainless Steel
84	49.5	64.0	55.7	8.6	0.9	151.5	AISI 302 Stainless Steel
85	94.0	80.5	105.7	8.3	0.9	88.8	AISI 302 Stainless Steel
86	51.0	64.0	57.4	7.5	0.9	221.6	AISI 302 Stainless Steel
87	123.0	130.0	138.4	3.8	0.9	703.5	AISI 302 Stainless Steel
88	64.0	78.0	72.0	7.4	0.9	183.8	AISI 302 Stainless Steel
89	67.5	80.0	75.9	6.2	0.9	296.1	AISI 302 Stainless Steel
90	19.0	29.0	21.4	6.1	0.9	1104.5	AISI 302 Stainless Steel
91	14.5	25.5	16.3	5.8	0.9	1683.2	AISI 302 Stainless Steel
92	76.0	89.0	85.5	7.1	0.9	175.3	AISI 302 Stainless Steel
93	64.0	76.5	72.0	7.3	0.9	191.5	AISI 302 Stainless Steel
94	106.5	106.5	119.8	89.6	0.9	0.1	AISI 302 Stainless Steel
95	300.0	300.0	337.5	7.1	0.9	44.4	AISI 302 Stainless Steel
96	60.0	70.0	67.5	5.1	0.9	597.9	AISI 302 Stainless Steel
97	82.0	100.0	92.2	9.7	0.9	63.8	AISI 302 Stainless Steel
98	320.0	320.0	320.0	7.0	1.0	78.6	AISI 302 Stainless Steel
99	47.0	65.0	47.0	9.0	1.0	251.7	AISI 302 Stainless Steel
100	40.0	55.0	40.0	7.0	1.0	628.6	AISI 302 Stainless Steel
101	56.5	70.0	56.5	7.0	1.0	445.1	AISI 302 Stainless Steel
102	320.0	320.0	320.0	11.0	1.0	20.3	AISI 302 Stainless Steel
103	79.5	188.0	79.5	9.0	1.0	148.8	AISI 302 Stainless Steel
104	39.2	64.0	39.2	9.0	1.0	301.8	AISI 302 Stainless Steel
105	29.5	65.0	29.5	9.0	1.0	401.1	AISI 302 Stainless Steel
106	21.0	3.0	21.0	5.0	1.0	3285.7	AISI 302 Stainless Steel
107	90.0	108.0	84.3	10.4	1.1	116.7	AISI 302 Stainless Steel
108	37.0	68.0	34.7	9.6	1.1	360.7	AISI 302 Stainless Steel
109	19.5	32.5	18.3	6.7	1.1	2004.1	AISI 302 Stainless Steel
110	23.5	42.0	22.0	8.5	1.1	817.0	AISI 302 Stainless Steel
111	91.0	103.5	85.3	6.9	1.1	393.3	AISI 302 Stainless Steel
112	113.5	133.0	106.4	12.4	1.1	54.7	AISI 302 Stainless Steel
113	300.0	300.0	281.2	11.1	1.1	28.8	AISI 302 Stainless Steel
114	26.5	38.0	24.8	6.4	1.1	1690.8	AISI 302 Stainless Steel
115	53.5	95.0	50.1	12.4	1.1	116.0	AISI 302 Stainless Steel
116	91.0	124.0	85.3	9.6	1.1	146.6	AISI 302 Stainless Steel
117	61.0	112.0	57.2	8.5	1.1	314.7	AISI 302 Stainless Steel
118	43.2	62.0	40.5	9.0	1.1	374.6	AISI 302 Stainless Steel
119	50.0	65.5	46.9	8.9	1.1	334.7	AISI 302 Stainless Steel
120	65.0	78.0	60.9	6.0	1.1	835.7	AISI 302 Stainless Steel
121	39.0	58.0	36.6	10.9	1.1	234.0	AISI 302 Stainless Steel
122	89.0	100.0	83.4	5.8	1.1	675.3	AISI 302 Stainless Steel
123	37.0	53.0	34.7	8.9	1.1	452.3	AISI 302 Stainless Steel
124	9.0	19.5	8.4	5.6	1.1	7415.1	AISI 302 Stainless Steel
125	122.0	144.0	114.3	11.4	1.1	65.4	AISI 302 Stainless Steel
126	79.5	94.0	74.5	8.5	1.1	241.5	AISI 302 Stainless Steel
127	320.0	320.0	266.7	8.8	1.2	98.4	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
128	70.5	92.0	58.8	10.8	1.2	241.7	AISI 302 Stainless Steel
129	57.0	75.0	47.5	8.8	1.2	552.5	AISI 302 Stainless Steel
130	91.0	130.0	75.8	10.3	1.2	215.8	AISI 302 Stainless Steel
131	43.5	100.0	36.3	12.8	1.2	235.3	AISI 302 Stainless Steel
132	93.5	160.0	77.9	11.3	1.2	159.1	AISI 302 Stainless Steel
133	110.0	134.0	91.7	8.8	1.2	286.3	AISI 302 Stainless Steel
134	82.5	116.0	68.8	10.3	1.2	238.1	AISI 302 Stainless Steel
135	32.0	68.0	26.7	12.8	1.2	319.8	AISI 302 Stainless Steel
136	60.5	84.0	50.4	8.8	1.2	520.5	AISI 302 Stainless Steel
137	33.0	65.0	27.5	12.8	1.2	310.1	AISI 302 Stainless Steel
138	34.0	60.0	28.3	10.8	1.2	501.1	AISI 302 Stainless Steel
139	68.5	94.0	57.1	10.8	1.2	248.7	AISI 302 Stainless Steel
140	36.0	78.0	30.0	10.8	1.2	473.3	AISI 302 Stainless Steel
141	22.5	50.0	18.8	10.8	1.2	757.2	AISI 302 Stainless Steel
142	59.0	89.0	47.4	11.0	1.2	332.6	AISI 302 Stainless Steel
143	62.0	78.0	49.8	8.6	1.2	664.6	AISI 302 Stainless Steel
144	89.0	112.0	71.5	11.0	1.2	220.5	AISI 302 Stainless Steel
145	120.0	140.0	96.4	9.2	1.2	280.2	AISI 302 Stainless Steel
146	54.5	111.0	43.8	10.3	1.2	438.9	AISI 302 Stainless Steel
147	26.5	45.0	21.3	8.8	1.2	1450.7	AISI 302 Stainless Steel
148	138.0	157.0	110.8	11.0	1.2	142.2	AISI 302 Stainless Steel
149	138.0	153.0	110.8	8.6	1.2	298.6	AISI 302 Stainless Steel
150	81.0	102.0	65.1	12.0	1.2	186.4	AISI 302 Stainless Steel
151	72.0	92.5	57.8	12.3	1.2	194.7	AISI 302 Stainless Steel
152	42.0	61.0	33.7	9.1	1.2	827.4	AISI 302 Stainless Steel
153	10.0	29.5	8.0	8.8	1.2	3844.5	AISI 302 Stainless Steel
154	55.5	72.0	44.6	8.6	1.2	742.4	AISI 302 Stainless Steel
155	55.5	76.5	44.6	12.3	1.2	252.6	AISI 302 Stainless Steel
156	35.0	52.5	28.1	10.5	1.2	645.0	AISI 302 Stainless Steel
157	113.0	135.0	90.8	11.8	1.2	140.6	AISI 302 Stainless Steel
158	140.0	158.0	112.4	9.8	1.2	198.5	AISI 302 Stainless Steel
159	113.0	130.0	90.8	9.1	1.2	307.5	AISI 302 Stainless Steel
160	121.0	145.0	97.2	12.3	1.2	115.8	AISI 302 Stainless Steel
161	42.0	62.5	33.7	11.8	1.2	378.2	AISI 302 Stainless Steel
162	76.5	96.5	61.4	11.8	1.2	207.6	AISI 302 Stainless Steel
163	300.0	300.0	241.0	8.8	1.2	128.1	AISI 302 Stainless Steel
164	121.0	158.0	97.2	8.3	1.2	379.0	AISI 302 Stainless Steel
165	98.0	180.0	70.0	10.6	1.4	397.4	AISI 302 Stainless Steel
166	29.5	118.0	21.1	13.6	1.4	625.1	AISI 302 Stainless Steel
167	36.5	59.0	24.8	6.5	1.5	5893.0	AISI 302 Stainless Steel
168	61.5	81.0	41.8	11.0	1.5	725.3	AISI 302 Stainless Steel
169	105.5	127.0	71.6	11.5	1.5	370.1	AISI 302 Stainless Steel
170	51.0	83.0	34.6	15.0	1.5	345.6	AISI 302 Stainless Steel
171	59.5	82.0	40.4	12.3	1.5	536.6	AISI 302 Stainless Steel
172	225.0	245.0	152.7	9.5	1.5	307.4	AISI 302 Stainless Steel
173	58.5	100.0	39.7	14.0	1.5	370.4	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
174	82.0	99.0	55.7	8.0	1.5	1410.3	AISI 302 Stainless Steel
175	43.5	64.0	29.5	11.0	1.5	1025.4	AISI 302 Stainless Steel
176	62.0	80.0	42.1	9.2	1.5	1228.0	AISI 302 Stainless Steel
177	82.0	104.0	55.7	11.5	1.5	476.2	AISI 302 Stainless Steel
178	52.5	83.0	35.6	15.3	1.5	316.4	AISI 302 Stainless Steel
179	142.0	160.0	96.4	8.0	1.5	814.4	AISI 302 Stainless Steel
180	300.0	300.0	203.7	9.7	1.5	216.6	AISI 302 Stainless Steel
181	19.0	30.0	12.9	7.0	1.5	9072.1	AISI 302 Stainless Steel
182	36.0	68.0	24.4	10.5	1.5	1424.2	AISI 302 Stainless Steel
183	29.0	62.0	19.7	12.0	1.5	1185.5	AISI 302 Stainless Steel
184	320.0	320.0	213.3	10.5	1.5	176.8	AISI 302 Stainless Steel
185	105.0	130.0	70.0	12.5	1.5	319.4	AISI 302 Stainless Steel
186	77.5	100.0	51.7	10.5	1.5	730.0	AISI 302 Stainless Steel
187	126.5	150.0	84.3	9.5	1.5	603.9	AISI 302 Stainless Steel
188	21.5	70.0	14.3	10.5	1.5	2631.5	AISI 302 Stainless Steel
189	38.0	65.0	25.3	11.5	1.5	1133.3	AISI 302 Stainless Steel
190	54.0	77.0	32.7	12.3	1.7	1040.4	AISI 302 Stainless Steel
191	41.0	85.0	24.8	8.6	1.7	3988.5	AISI 302 Stainless Steel
192	300.0	300.0	181.7	6.4	1.7	1314.9	AISI 302 Stainless Steel
193	87.0	107.0	52.7	11.0	1.7	901.6	AISI 302 Stainless Steel
194	83.5	107.0	50.6	12.8	1.7	597.3	AISI 302 Stainless Steel
195	38.5	96.0	23.3	12.3	1.7	1459.3	AISI 302 Stainless Steel
196	43.0	58.5	26.0	8.8	1.7	3550.9	AISI 302 Stainless Steel
197	305.0	305.0	184.7	11.6	1.7	219.4	AISI 302 Stainless Steel
198	29.5	47.5	17.9	9.3	1.7	4389.1	AISI 302 Stainless Steel
199	300.0	300.0	181.7	16.1	1.7	83.7	AISI 302 Stainless Steel
200	34.0	58.5	20.6	12.9	1.7	1433.2	AISI 302 Stainless Steel
201	116.0	140.0	70.3	12.3	1.7	484.3	AISI 302 Stainless Steel
202	110.0	140.0	61.1	14.2	1.8	517.4	AISI 302 Stainless Steel
203	29.5	90.0	16.4	11.2	1.8	3932.3	AISI 302 Stainless Steel
204	80.0	104.0	43.7	13.2	1.8	965.8	AISI 302 Stainless Steel
205	65.0	88.0	35.5	10.2	1.8	2581.2	AISI 302 Stainless Steel
206	500.0	511.0	273.4	9.7	1.8	390.3	AISI 302 Stainless Steel
207	305.0	305.0	166.8	12.5	1.8	298.4	AISI 302 Stainless Steel
208	78.5	101.0	42.9	13.2	1.8	984.2	AISI 302 Stainless Steel
209	74.0	100.5	40.5	14.2	1.8	838.3	AISI 302 Stainless Steel
210	52.0	74.0	28.4	11.2	1.8	2435.3	AISI 302 Stainless Steel
211	76.5	95.5	41.8	9.7	1.8	2551.2	AISI 302 Stainless Steel
212	84.0	107.5	45.9	11.2	1.8	1507.6	AISI 302 Stainless Steel
213	42.5	74.0	23.2	16.2	1.8	982.3	AISI 302 Stainless Steel
214	31.5	55.5	17.2	6.4	1.8	21671.8	AISI 302 Stainless Steel
215	320.0	320.0	160.0	12.0	2.0	499.1	AISI 302 Stainless Steel
216	150.0	175.0	75.0	13.0	2.0	837.5	AISI 302 Stainless Steel
217	46.0	70.0	23.0	11.0	2.0	4507.9	AISI 302 Stainless Steel
218	35.0	60.0	17.5	11.0	2.0	5924.7	AISI 302 Stainless Steel
219	26.5	50.0	13.3	11.0	2.0	7825.0	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
220	17.2	45.0	8.6	11.0	2.0	12049.0	AISI 302 Stainless Steel
221	16.4	40.0	8.2	11.0	2.0	12644.1	AISI 302 Stainless Steel
222	9.7	30.0	4.9	11.0	2.0	21377.6	AISI 302 Stainless Steel
223	40.0	62.5	19.0	11.9	2.1	5336.9	AISI 302 Stainless Steel
224	32.0	63.0	15.2	15.4	2.1	3076.6	AISI 302 Stainless Steel
225	58.0	91.0	27.5	16.9	2.1	1284.2	AISI 302 Stainless Steel
226	100.0	131.0	47.4	16.4	2.1	815.1	AISI 302 Stainless Steel
227	74.0	105.0	35.1	16.4	2.1	1101.5	AISI 302 Stainless Steel
228	58.0	85.0	27.5	13.4	2.1	2577.2	AISI 302 Stainless Steel
229	58.0	80.0	27.5	12.4	2.1	3252.8	AISI 302 Stainless Steel
230	125.0	148.0	59.3	11.5	2.1	1892.4	AISI 302 Stainless Steel
231	10.0	32.5	4.7	10.6	2.1	30211.9	AISI 302 Stainless Steel
232	62.0	102.0	29.4	13.4	2.1	2410.9	AISI 302 Stainless Steel
233	300.0	300.0	142.3	15.9	2.1	298.2	AISI 302 Stainless Steel
234	50.0	73.5	23.7	12.9	2.1	3351.1	AISI 302 Stainless Steel
235	78.0	105.5	37.0	14.9	2.1	1393.7	AISI 302 Stainless Steel
236	46.5	77.0	22.1	15.1	2.1	2246.0	AISI 302 Stainless Steel
237	78.5	100.0	37.2	10.9	2.1	3539.3	AISI 302 Stainless Steel
238	105.5	128.0	50.0	11.9	2.1	2023.5	AISI 302 Stainless Steel
239	106.0	133.0	50.3	14.5	2.1	1112.8	AISI 302 Stainless Steel
240	138.0	158.0	65.5	10.5	2.1	2252.5	AISI 302 Stainless Steel
241	128.0	154.0	60.7	13.6	2.1	1117.0	AISI 302 Stainless Steel
242	58.5	85.5	27.8	14.9	2.1	1858.2	AISI 302 Stainless Steel
243	48.0	74.0	22.8	14.4	2.1	2509.0	AISI 302 Stainless Steel
244	36.0	57.5	17.1	10.7	2.1	8158.9	AISI 302 Stainless Steel
245	90.0	222.0	42.7	14.9	2.1	1207.8	AISI 302 Stainless Steel
246	26.0	56.0	12.3	12.4	2.1	7256.3	AISI 302 Stainless Steel
247	31.3	58.0	13.0	12.6	2.4	10968.8	AISI 302 Stainless Steel
248	115.0	155.0	47.7	11.6	2.4	3944.0	AISI 302 Stainless Steel
249	101.0	135.0	41.9	19.6	2.4	929.7	AISI 302 Stainless Steel
250	110.0	154.0	45.6	21.6	2.4	637.6	AISI 302 Stainless Steel
251	123.0	160.0	51.0	19.6	2.4	763.4	AISI 302 Stainless Steel
252	65.5	108.5	27.1	23.6	2.4	820.9	AISI 302 Stainless Steel
253	89.0	127.0	36.9	22.6	2.4	688.0	AISI 302 Stainless Steel
254	72.5	105.5	30.0	19.6	2.4	1295.1	AISI 302 Stainless Steel
255	72.5	115.0	30.0	23.6	2.4	741.6	AISI 302 Stainless Steel
256	80.5	110.0	33.4	15.1	2.4	2552.4	AISI 302 Stainless Steel
257	30.0	52.0	12.4	16.1	2.4	5649.4	AISI 302 Stainless Steel
258	110.0	145.0	44.0	17.5	2.5	1428.7	AISI 302 Stainless Steel
259	73.7	100.0	29.5	12.5	2.5	5851.4	AISI 302 Stainless Steel
260	47.5	75.0	19.0	12.5	2.5	9078.9	AISI 302 Stainless Steel
261	55.0	80.0	22.0	12.5	2.5	7840.9	AISI 302 Stainless Steel
262	42.0	70.0	16.8	12.5	2.5	10267.9	AISI 302 Stainless Steel
263	37.5	65.0	15.0	12.5	2.5	11500.0	AISI 302 Stainless Steel
264	34.5	60.0	13.8	12.5	2.5	12500.0	AISI 302 Stainless Steel
265	27.5	55.0	11.0	12.5	2.5	15710.4	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
266	22.0	50.0	8.8	12.5	2.5	19602.3	AISI 302 Stainless Steel
267	18.5	45.0	7.4	12.5	2.5	23310.8	AISI 302 Stainless Steel
268	15.0	40.0	6.0	12.5	2.5	28750.0	AISI 302 Stainless Steel
269	82.5	127.0	29.8	27.2	2.8	842.8	AISI 302 Stainless Steel
270	108.0	157.0	39.0	27.2	2.8	643.8	AISI 302 Stainless Steel
271	66.0	103.0	23.8	19.2	2.8	2991.1	AISI 302 Stainless Steel
272	39.0	59.0	14.1	15.2	2.8	10188.8	AISI 302 Stainless Steel
273	69.0	98.0	24.9	15.7	2.8	5227.0	AISI 302 Stainless Steel
274	91.5	123.5	33.0	16.2	2.8	3588.5	AISI 302 Stainless Steel
275	125.0	154.0	45.1	14.7	2.8	3513.7	AISI 302 Stainless Steel
276	60.5	109.0	21.8	15.2	2.8	6568.0	AISI 302 Stainless Steel
277	41.5	84.0	15.0	13.2	2.8	14606.6	AISI 302 Stainless Steel
278	63.5	88.5	22.9	13.2	2.8	9546.0	AISI 302 Stainless Steel
279	50.0	80.0	16.7	14.0	3.0	15276.1	AISI 302 Stainless Steel
280	43.9	75.0	14.6	14.0	3.0	17398.7	AISI 302 Stainless Steel
281	37.5	70.0	12.5	14.0	3.0	20368.1	AISI 302 Stainless Steel
282	33.0	65.0	12.2	14.3	2.7	12825.0	AISI 302 Stainless Steel
283	28.0	60.0	9.3	14.0	3.0	27278.7	AISI 302 Stainless Steel
284	25.3	55.0	8.4	14.0	3.0	30189.8	AISI 302 Stainless Steel
285	595.0	610.0	198.3	17.0	3.0	717.0	AISI 302 Stainless Steel
286	84.5	127.5	27.7	23.0	3.0	2220.8	AISI 302 Stainless Steel
287	132.0	182.0	43.3	24.5	3.0	1175.8	AISI 302 Stainless Steel
288	127.5	170.0	41.8	21.0	3.0	1934.8	AISI 302 Stainless Steel
289	500.0	500.0	164.0	12.0	3.0	2657.9	AISI 302 Stainless Steel
290	500.0	500.0	164.0	17.5	3.0	853.7	AISI 302 Stainless Steel
291	500.0	527.0	164.0	25.0	3.0	292.1	AISI 302 Stainless Steel
292	51.0	81.0	16.7	15.0	3.0	13309.6	AISI 302 Stainless Steel
293	79.0	120.0	25.9	19.5	3.0	3902.2	AISI 302 Stainless Steel
294	113.0	155.0	33.2	23.1	3.4	2831.5	AISI 302 Stainless Steel
295	56.5	88.0	16.6	16.1	3.4	16730.3	AISI 302 Stainless Steel
296	120.0	152.0	35.3	16.3	3.4	7590.7	AISI 302 Stainless Steel
297	84.0	185.0	24.7	18.1	3.4	7919.2	AISI 302 Stainless Steel
298	154.0	240.0	45.2	22.1	3.4	2372.7	AISI 302 Stainless Steel
299	500.0	530.0	146.9	27.1	3.4	396.3	AISI 302 Stainless Steel
300	50.0	84.0	14.3	16.5	3.5	20168.7	AISI 302 Stainless Steel
301	164.0	220.0	43.6	31.7	3.8	1234.3	AISI 302 Stainless Steel
302	90.5	155.0	24.1	35.2	3.8	1634.3	AISI 302 Stainless Steel
303	79.0	148.0	21.0	17.2	3.8	15988.4	AISI 302 Stainless Steel
304	77.5	122.0	20.6	25.2	3.8	5194.0	AISI 302 Stainless Steel
305	113.0	164.0	30.1	23.2	3.8	4563.3	AISI 302 Stainless Steel
306	180.0	220.0	42.9	21.3	4.2	6403.0	AISI 302 Stainless Steel
307	116.0	187.0	27.7	35.8	4.2	2093.7	AISI 302 Stainless Steel
308	132.0	178.0	31.5	21.8	4.2	8144.5	AISI 302 Stainless Steel
309	88.5	124.0	21.1	30.8	4.2	4308.9	AISI 302 Stainless Steel
310	500.0	532.0	119.3	30.8	4.2	762.7	AISI 302 Stainless Steel
311	193.0	242.0	46.1	21.3	4.2	5971.7	AISI 302 Stainless Steel

Spring ID	L [mm]	L_0[mm]	N_b	D [mm]	d [mm]	K [N/m]	Material
312	170.0	257.0	40.6	18.8	4.2	9858.3	AISI 302 Stainless Steel
313	101.0	150.0	24.1	24.8	4.2	7231.0	AISI 302 Stainless Steel
314	500.0	500.0	119.3	25.8	4.2	1297.4	AISI 302 Stainless Steel
315	92.0	145.0	22.0	25.8	4.2	7050.9	AISI 302 Stainless Steel
316	95.0	228.0	22.7	24.3	4.2	8171.9	AISI 302 Stainless Steel
317	103.0	208.0	24.6	24.3	4.2	7537.2	AISI 302 Stainless Steel
318	118.0	235.0	28.2	24.3	4.2	6579.1	AISI 302 Stainless Steel
319	168.0	210.0	36.7	21.4	4.6	10424.0	AISI 302 Stainless Steel
320	131.0	205.0	28.7	35.4	4.6	2957.9	AISI 302 Stainless Steel
321	105.0	135.0	23.0	28.4	4.6	7142.7	AISI 302 Stainless Steel
322	72.0	215.0	15.7	31.4	4.6	7709.2	AISI 302 Stainless Steel
323	500.0	500.0	97.0	24.8	5.2	4099.1	AISI 302 Stainless Steel
324	131.0	185.0	25.4	28.8	5.2	9997.4	AISI 302 Stainless Steel
325	130.0	250.0	25.2	22.8	5.2	20279.8	AISI 302 Stainless Steel
326	110.0	215.0	21.3	24.3	5.2	19804.1	AISI 302 Stainless Steel
327	105.0	170.0	18.8	36.9	5.6	8899.2	AISI 302 Stainless Steel
328	350.0	435.0	57.9	44.0	6.0	2342.3	AISI 302 Stainless Steel
329	450.0	535.0	74.4	44.0	6.0	1821.8	AISI 302 Stainless Steel
*330	-	-	-	-	-	2468.6	AISI 302 Stainless Steel
*331	-	-	-	-	-	3702.9	AISI 302 Stainless Steel
*332	-	-	-	-	-	4937.2	AISI 302 Stainless Steel
*333	-	-	-	-	-	6171.5	AISI 302 Stainless Steel