Intellectual Output: IO4

Version 1.0.0

Chapter IV.

Introduction to robotics for medical professionals

Funded by Erasmus+ Programme 2014-2020 Key Action 2: Strategic Partnership Projects
Agreement n° 2019-1-TR01-KA203-077662

Project Acronym: WeCoRD

Project Full Title: Open Educational Resources on Enabling Technologies in Wearable and Collaborative Robotics

Call: ERASMUS+

Grant Number: 2019-1-TR01-KA203-077662

Project URL: https://wecord.eu/en

Intellectual Output nature:	Lecture Notes (LN)
Dissemination level:	Public (PU)
Number of pages:	188
Keywords:	Hardware Components, Software Components, Design, Requirements
Authors:	K.Nizamis, M. Vlutters, E. B. Vander Poorten, I. van Nes, UTwente
Peer review:	Barkan Ugurlu, OzU, K.Nizamis, M. Vlutters, UTwente,E. B. Vander Poorten, KU Leuven

IV.1. Introduction

The course "Introduction to robotics for medical professionals" aims to present a common ground for students of medical and engineering disciplines alike. This will pave the way for further disciplinary integration of medical professionals in the research, development, and effective use of medical robotics. Medical professionals and students will better understand the basic robotics principles and can more efficiently contribute to interdisciplinary teams working on the development and implementation of healthcare robotics. The underlying objective of this chapter is to facilitate further adoption of robotics in healthcare environments. As medical professionals will be able to better understand the potential and limitations of robotics, they may provide complementary insights to engineers and roboticists, and actively collaborate in robotic projects.

Robotics will play a progressively important role in medicine, due to ongoing advances in actuators, sensors, materials, and artificial intelligence (AI) algorithms [Niz+21]. The multifaceted use of robotics in healthcare is evident by their various use cases [PL17]: Surgical robots help surgeons perform challenging operations with maximum precision in a safe and minimally invasive way [CCS18]. Wearable robots (or exoskeletons) are achieving new levels of physical interfacing with their human users. They are able to support people with chronic disabilities (assistive), and provide physical therapy to those that seek improvement of their motor functions (rehabilitation) [LRG16]. There are numerous other applications of robotics in healthcare including prosthetic limbs, care and affective robots, nurse assisting robots, collaborative teleoperated robots, futuristic applications include swarm micro-/nano- robots that can deliver medicine inside the body or scan for abnormalities [PL17].

The variety in the applications of robotics in the healthcare sector comes with an ever-expanding ecosystem of disciplines that are engaged with robotics, such as medical professionals for surgery [Gom11] and rehabilitation [Rei+17]. This chapter will be addressing the need for a common language to promote mutual understanding and communication between all disciplines involved, including medical professionals. In this way, medical professionals will be able to extend their knowledge beyond the mere use of medical robots, through a better understanding of their basic principles and applications. Moreover, it will enable people with various backgrounds to discuss the challenges involved in the introduction of robotics in their sectors, the associated constraints, their expectations and concerns, ethical and legal issues involved, and the changes brought to their workplaces [CCS18]. A further goal of this chapter is that medical professionals will be able to assess more effectively what solutions are available for problems they encounter in current practice. Medical professionals and students will receive a more realistic image of the state-of-the-art. Through this course they will receive a better understanding of the feasibility of certain approaches. Approaches that they envision or come up with in a response to challenges they are facing in the OR or in a rehabilitation center.

Modern healthcare is shifting from traditional hospital-centric care towards Healthcare 4.0 [LC21] intertwined with technologies such as AI, home-based healthcare, and the Internet of Things (IoT) combined heavily with robotics. The future surgeon, physiotherapist, or clinician needs to be able to understand the basic principles of robotics, in order to keep up effectively with their use in a medical setting [Weh19].This emerging nature of the field provides inspiring examples of technology applications and of practitioners using combinations of enabling technologies. This chapter aims to prepare medical-oriented students and professionals by training them to implement innovative and relevant healthcare improvements in today's unpredictable, complex and globalized world. In order to achieve this, this chapter focuses not only on providing knowledge and skills in a specific field of study, but also on discussing a broader range of skills and qualities such as systems thinking, logistics and acquisition, stakeholder management.

This chapter is split into 6 sections that together attempt to address its educational aim and are linked to specific Intended Learning Outcomes (ILOs). This section offers an introduction to the chapter and its purpose, together with an overview of the topics discussed. The section illustrates how they link to the previous chapters, how they combine, and how they form a coherent educational path on medical robotics. Section 2 offers the reader information on the learning objectives and on how to read this chapter (Figure IV.1). Section 3, provides a brief history on robotics, and aims to draw parallels between robotic hardware and the human movement control system. These parallels are used to demonstrate the main components of robotic systems[1]. Section 4 treats various robotic working principles based on several state-of-the-art medical robots. More detailed information on such principles can be found in Chapters II and III. In section 5 we highlight the context and main stakeholders surrounding the development of medical robots, and discuss them from a system viewpoint. Section 6, explains the acquisition process for medical robotics, using the description of the process for a real scenario by an organization in the Netherlands. Lastly, in Section 7 the reader can find an extensive list of terms used in this chapter (and the other chapters) and their definition. Sections 3-7 are content-focused and aim to offer information on medical robotics as denoted by the little 'i' icon (see Figure IV.1). Together these 7 sections can help the reader reach the learning objectives that are described in the following section.

[1] Chapter I gives a comprehensive view on such components, and Chapters II, and III elaborate on sensing and control

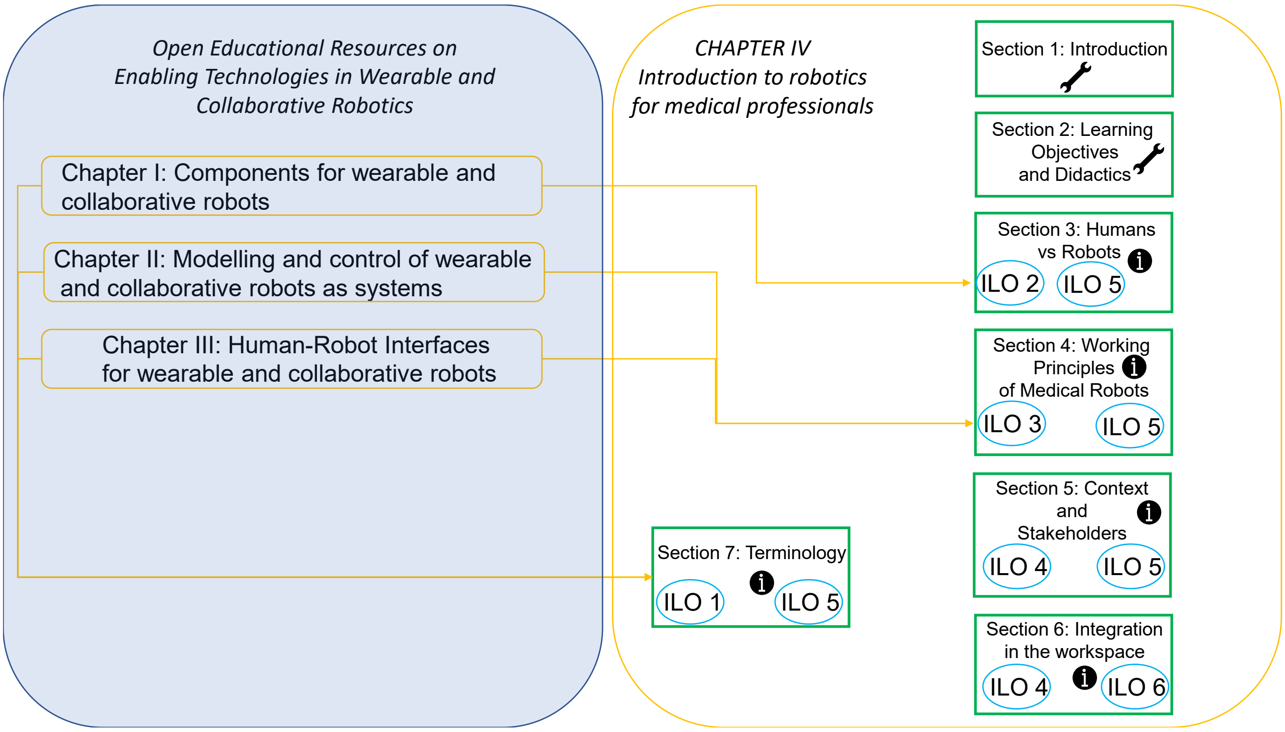


Figure IV.1.: *Here we present the internal structure of this chapter and its connection to the other chapters in the reader. Chapter IV (red rectangle in the middle) has 7 sections (green larger ellipses), and each section is linked to two Intended Learning Objectives (ILO, blue smaller ellipses), as presented in section 2. Some of the sections of this chapter discuss topics that are addressed in more detail in other chapters of the textbook (orange rectangles) .* Some sections are not content-focused but aim to act as a tool on how to use this chapter (denoted by a wrench icon) and some are content-focused and aim to offer information on medical robotics as denoted by the little '*i*' icon.

IV.2. Learning objectives and didactics

How to read this chapter?
This chapter provides material on the basics of robotics, intended specifically for students, graduates, trainees, and professionals with a medical-oriented background. It treats the basic principles of actuation, sensing, and control, and attempts to establish a common ground with which medical professionals/students and roboticists can communicate and understand one another with regard to these matters. The material will draw parallels between a robotic system and the human body. It may therefore also be interesting to readers already familiar with robotics, but looking to know more about those similarities.

Learning objectives
After following the course Robotics for Medical Professionals, the student will be able to:

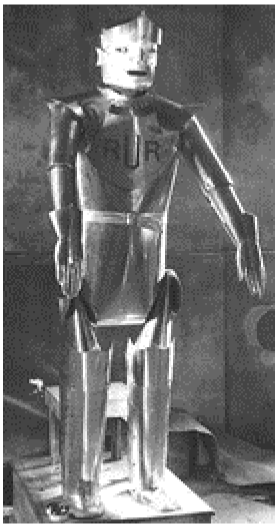
1. Describe and discuss the basic terminology encountered in robotics.
   (Section 7)
2. Describe and discuss the main components of a robotic device (sensors, actuators), and draw parallels between these and the human body.
   (Section 3)
3. Critically assess the use of robotics and their main components in a medical professional context. Specifically, discuss and reflect on (Section 4):
4. Critically assess the use of robotics and their main components in a medical professional context. Specifically, discuss and reflect on (Section 4):
   a) The main applications of state-of-the-art robotics in the medical domain
   b) The main underlying working principles of these robotic systems and their components with respect to a medical application.
   c) The physical interaction between the robotic device and its environment, in particular with a human.
5. Identify and reflect on the context and main stakeholders involved in the research, development, and application of robotic systems in the medical domain. (Section 5)
6. Evaluate and critically reflect on the usability challenges of medical robotics (logistics, acquisition, embedding in workplace, safety, access) (Section 6).
7. Use a common language to communicate with a multidisciplinary team and stakeholders in medical robotic systems.
   (Sections 3-7)

IV.3. Humans and Robots

IV.3.1. A brief history of robotic technology

IV.3.1.1. Teleoperation the cradle of robotics

To acquire a better idea of how robots work, it can be helpful to get some insight on where they come from and how they inspired people over the years. This section provides a short overview of some key milestones in the history of robots. From the dawn of robots until modern days where robots have become commonplace.



The word "robot" can be traced back to a theater play in 1912 by the Czech director Karl Capek. In the show a human-like structure was referred to as "Rossums Universal Robot" (RUR). The robot was static, but at least it looked like a human and the audience could imagine that such robot could come in handy especially if it were as universally deployable as the name suggested. Since that period quite a few people tend to associate robots with such human-like structures ("humanoids"), but clearly that is a rather narrow scope as the vast majority of robots has little resemblance to human anatomy. A major boost in the popularity of robots arose thanks to Isaac Asimov's books that appeared in 1942. Asimov's short stories were welcomed with great interest and are still commonly read these days. Possibly the most notorious impact Asimov enjoyed was through the introduction of "the three laws of robotics".

> A robot may not injure a human being, or, through inaction, allow a human being to come to harm.
> A robot must obey the orders given it by human beings except where such orders would conflict with the First Law
> A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Still, it took six more years before the first physical incarnation of a "robot" appeared. It was 1948 when Ray Goertz introduced a mechanical teleoperator. After the second world war various applications of nuclear radioactive materials were being explored for civil purposes, but researchers soon discovered that handling these materials with lead gloves and a sort of forceps (as in below figure) was not a healthy undertaking. Goertz developed a so-called "mechanic master-slave" system to safely handle hazardous material from a plutonium reactor at Hanford in south-central Washington. The system consisted of two pairs of mechanical manipulators in which the joints of the slave manipulator where connected through cables to the joints of the master manipulator. As a result, the joints at the slave side would move if the operator would move the joints at the master side. With a direct view on the distal side, about three to four meters away, separated by a protective transparent leaded glass wall, the operator would then adjust the master handles to steer the slave toward the targeted direction. The figure below shows several pictures of the scene. Although friction in the cabling did disturb motion, in practice this approach whereby a distal device copied manipulators at the proximal site closely was found highly intuitive. Operators quickly got experienced with this system and managed to produce complex motion commands with this configuration. This gave them the confidence to handle even dangerous substances. The figure aside shows

		
Manual handling of radioactive material by leaden gloves	Handling of waste at a safe distance, by steering a pair of so-called master handles	Distally a pair of so-called “slave devices” are actually manipulating the radioactive material.

a better overview of such a mechanical system. The picture shows a woman steering a match to light a cigarette for a colleague who confidently bends over to get lit. The figure shows the cabling that is routed between the master and the slave via a set of bellow tubes that are routed above the shielding window. These first teleoperators were purely mechanical and did not possess any actuator or sensor. They were very robust as due to their simplicity only few components could fail. A limitation of these first systems is that they could only be used for teleoperation over relatively short distances. It took Goertz a few more years to realize that it would also be possible to electrically connect the two manipulators. By putting actuators and sensors in the joints of the mechanisms and by steering commands to the actuators such that the corresponding joints would take on the same angles, the configuration of the master could be copied at the slave side. Not a steel cable, but an electrical communication cable formed now the connection between the master and the slave manipulator. This was the first electrical master-slave manipulator system. By 1954, a modified version of Goertz’ master-slave manipulator entered commercial production. The overall architecture of such a setup is sketched in the below picture. The systems contains two mechanical devices, the so-called “master robot” is a sophisticated joystick that is directly controlled by the human operator and the so-called “slave robot” which is the robotic system that operates in the remote environment. Nowadays these terms are often replaced with “leader” and “follower”. The follower controllers receive commands from the leader that are sent to it via some form of communication channel. If the two robots are not separated too far from each other, both controllers may be connected to the same computer that runs the overall control loop and spawns commands to both robot controllers. If the distance increases this does not become feasible and a separate computer will then be used per robotic system. The communication between both systems can take place through a serial connection. If the information flows in both directions, i.e. from operator to the environment and back, one speaks of a

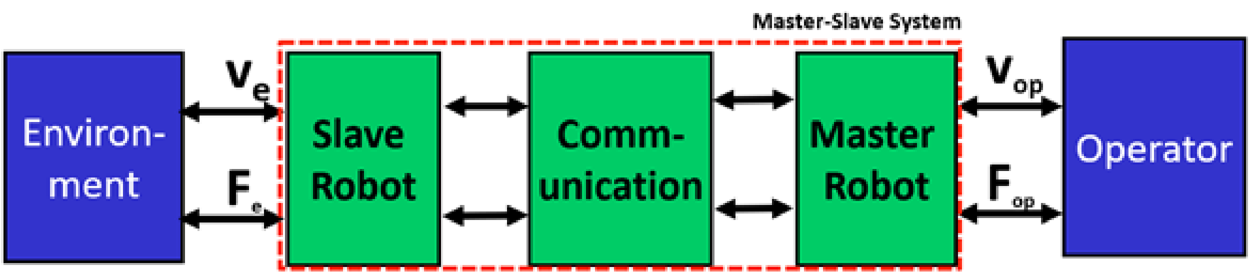


Figure IV.2.: *Overall tele-operation architecture. Velocity (V) and force (F) commands from operator (op) are relayed to the environment (robot) through the master (leader) - slave (follower) system.*

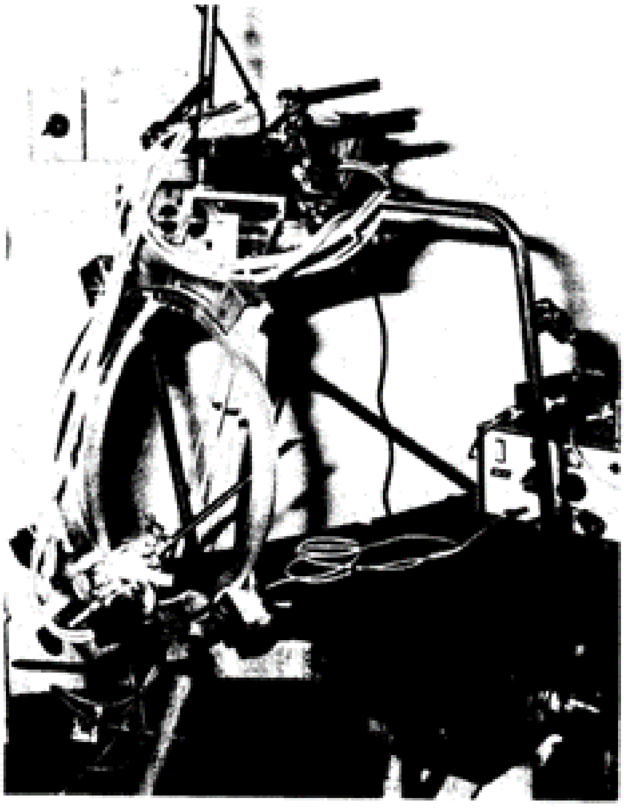
bilateral setup. This in contrast to systems where this information flows only in one direction, which are referred to as unilateral systems. In practice, even unilateral systems are not truly unilateral as all teleoperation systems are equipped with some means of visualization. A camera mounted remotely is positioned to observe the slave's motion. These images are sent to the operator who gets to see them on a monitor. The ideal behavior that one wants to establish in such a system is a so-called 'mechanical transparent' behavior. This means that the operator should have the feeling as if they are operating directly in the remote environment. The operator would in such an ideal case forget the presence of the intermediate mechanical (mechatronic) system which hence would appear transparent to them.

IV.3.1.2. The first stand-alone robots

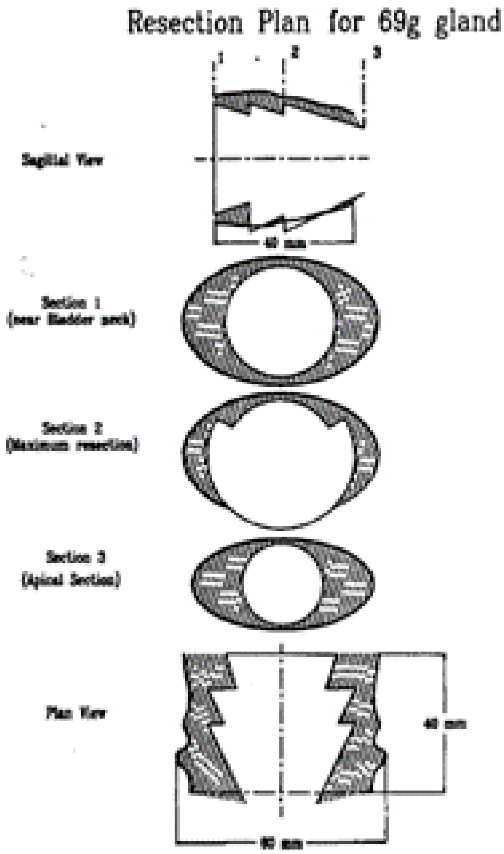
In the '50s, George C. Devol, an inventor from Louisville, Kentucky, realized that the commands that were sent to the follower's robot controller did not necessarily need to come from a proximal system controlled by a human operator, but could equally be generated at the distal system. The system could simply read out a file of motion commands or accept its commands from an algorithm that was computing appropriate commands to achieve a certain goal on-the-fly. He found that it was thus possible that such systems could operate independently without direct human input. The system could follow pre-programmed trajectories reliably and tirelessly. Compared to other machinery the trajectories could be easily adapted and re-programmed to a specific need or task. This was the onset of factory automation. The earliest of such stand-alone robots were created in the early 1950s. G.C. Devol invented and patented a reprogrammable manipulator that he called the "Unimate" which was a shortcut for "Universal Automation" referring to the flexibility and independence of these stand-alone robots. From the onset people were tremendously fascinated by these electro-mechanical systems that appeared in tv-shows that were broadcasted world-wide. Industrially, it took until the 1960s before these devices were truly commercialized. Joseph Engleberger, who acquired Devol's patent, successfully launched an industrial robot with his company called Unimation.

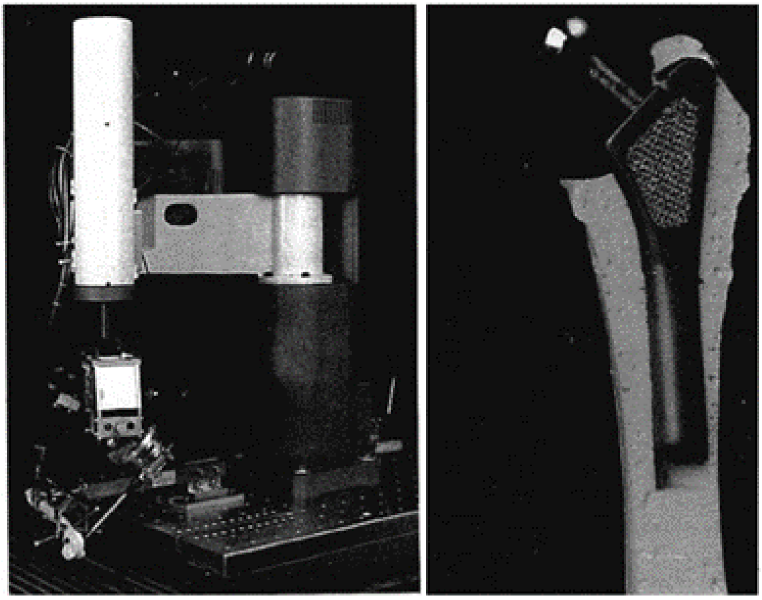
IV.3.1.3. The first medical robots

The first surgical robots appeared only a few decades later. In 1985 on April 11, the first surgical robot was employed on a 52-year-old man. The 1988 paper by Kwoh et al. reports on the use of an Unimation PUMA 200 robot that was used in stereotactic brain tumor biopsy [Kwo+88]. Relying on CT images the calibrated robot was able to accurately point towards a target in the patient's brain. The surgeon would then perforate the skull and insert the biopsy probe relying on that guidance. The surgical intervention itself is actually not executed by the robot. The robot merely serves as a highly accurate precise guide for the operator. This is a configuration that is quite common even in today's surgical robotic systems where the final surgical act is conducted by the operating surgeon. More complex motions were guided by a robotic system that was developed by Davies et al. for prostatectomy [Dav+91]. Initially Davies et al. departed from a mechanism that was mounted on a PUMA robot (partially similar to Kwoh). The mechanism was developed to constrain motion of a cutter in transurethral resection of the prostate (TURP), but the researchers quickly realized that the size and mobility of such large-scale industrial robots imposed unnecessary risks upon patient and surgeon, and therefore continued to develop a dedicated structure for this particular application. Along the history of surgical robotics this motivation for designing dedicated devices resurfaces countless times, where researchers balance the advantages of a general-purpose and more affordable device versus a dedicated and more costly device that is more suited to accomplish a single task with superior quality. The figure below shows a dedicated device: a so-called safety-frame that was developed by Davies et al. Similar to Kwoh's approach, the system guided the surgeon to execute precise tissue ablation tasks. Rather than simply pinpointing a single particular region, the system allowed the surgeon to move over an entire volume whereby the system constrained motion to safe regions and imposed boundaries such that only targeted tissue could be removed. While this system did not find commercial use, a similar approach was followed by researchers in Integrated Surgical Systems and IBM for robotic preparation of the femoral canal to allow placement of a prosthesis in cementless total hip arthroplasty [Pau+92]. The so-called ROBODOC system was built around a Japanese industrial five-axis robot from Sankyo-Seiki. This robot was expanded to offer six degrees-of-freedom and was equipped with a force-torque sensor as well as a standard high-speed surgical cutting tool. During the intervention this robot was rigidly connected to the femur and programmed to execute a highly precise milling action according to a pre-operatively determined surgical plan. To ensure proper execution of the pre-operative plan the location of the femur is taught by a registration procedure prior to milling. From this information the robot can then compute the location where the milling action needs to be performed.



. Motorised Prostatic Safety Frame.

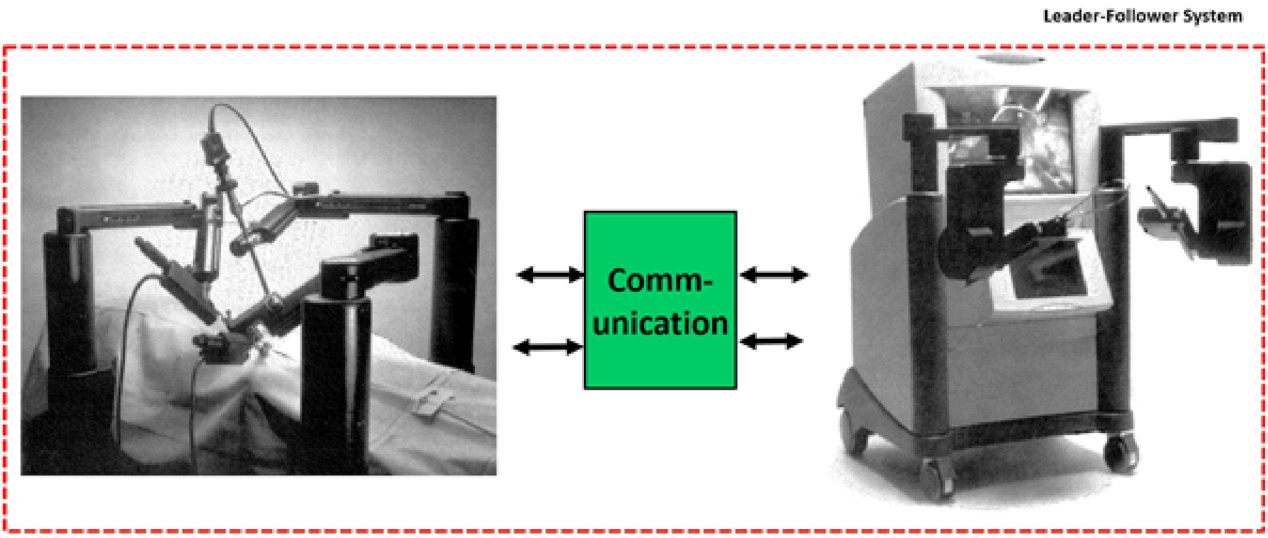




The figure above shows the robotic setup and the femur with inserted implant after robotic milling. In contrast to the previous examples, the surgeon's role is restricted in this approach to a mainly supervisory role. Once the registration has been done and a registration tool is replaced for a cutter the robotic milling is initiated. During the milling the surgeon observes the process through an online display whereby the progress is visualized relative to pre-operative plan. If discrepancy is detected the clinician can halt the system. With a claimed +/- 0.4mm the overall accuracy of this robotic approach is extremely good. Despite this system being the first FDA-approved 'active' robotic system, i.e. a system where the robot itself executed the surgical action, the ROBODOC did not receive further traction in the clinical field due to conflicting results (positive and negative) [Rus07]. Whereas the robotic system was experienced to show greater accuracy, it was not clear whether this led to an improved clinical outcome. Furthermore the procedure was found to take longer than a non-robotic approach and the extra procedure to implant registration pins prior to pre-operative scans was found disadvantageous. Furthermore a higher revision rate was reported in a study by Honl et al. [Hon+03]. This was caused by a higher rate of postoperative dislocation. Complications due to registration failures in up to 10% of the cases, increased exposure to radiation, elevated costs and learning curves by the operators were other factors that hindered the uptake of the ROBODOC systems [Sub+19]. From the alternative systems that were developed, CASPAR, Athrobot and MAKO, only the latter was successful and became popular [Sub+19]. The MAKO system is a semi-active system that provides guidance to the surgeon who then executes the intervention with the help of the robotic system.

IV.3.1.4. The breakthrough surgical robots

The breakthrough in surgical robotics nevertheless did not take place with active or semi-active robots, but rather through teleoperated leader-follower systems. In minimally-invasive surgery the AESOP, the Automated Endoscopic System for Optimal Positioning (AESOP) was introduced. This system essentially allowed surgeons to control the position of a laparoscopic camera system via voice control or joystick control in a typical leader-follow scenario. As the number of robotics arms grew, the system was re-branded to the ZEUS robotics system. In this ZEUS system the main surgeon was dislocated to a remote console from where he/she could control the robotic instruments via a dedicated pair of joysticks. The figure below shows the layout of this system. In practice all devices were connected and controlled by a single controller and the distance between patient and surgeon remained limited to several meters.



While this ZEUS system enjoyed great success, through a number of legal disputes its company Computer Motion was taken over by its main competitor Intuitive Surgical who currently is still dominating the field in surgical robotics with its Da Vinci Robotic systems which will be elaborated in greater detail in the following sections. At the end of 2021 the Da Vinci Robotic Systems had a worldwide install base of 6730 devices[2]. These 'robots' are being used in gynaecology, urology, general surgery, but also other applications. Figure IV.3 shows the worldwide use across surgical disciplines of the Da Vinci Surgical System over the period from 2011 to 2016 [Azi+18]. Since then the popularity of this robotic system has only grown.

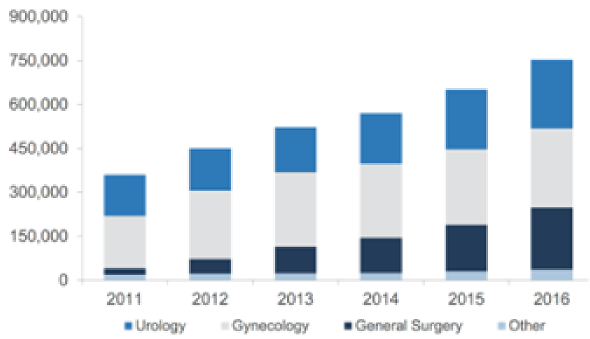


Figure IV.3.: *Worldwide use surgical disciplines*

IV.3.2. Human vs Robot

Humans plan motions using their brain, which is informed by a multitude of physiological sensors via the peripheral nerves. Those sensors can generally be divided into exteroceptive sensors and proprioceptive sensors. Exteroceptive sensors obtain information about the environment, and include tactile skin sensors and the eyes. Proprioceptive sensors signal information about the body itself, and include muscle spindles to signal muscle length and velocity, golgi tendon organs to signal tendon force, and the vestibular system to signal head orientation and acceleration [Vel99]. These sensors come with limitations (uncertainty, noise, neural delay) that need to be taken into account by the brain. The motion planning results in a motor command that activates the muscles through the peripheral nerves to execute the planned movement. The muscles act as the actuators for the human body that move the human skeleton. The movement results in sensory feedback to the brain that will be used to plan the next movement with the use of the same control loop. An overview of this loop is shown in Figure IV.4. A similar architecture may be created for an artificial movement system. Generally, robots perceive their environment with the use of **sensors** to **control** their **actuators**. Sensors acquire hardware-related signals coming from the robot, or signals from the environment of the robot, including for example physiological signals from a potential human user coupled to the robot. A controller utilizes this sensor information to compute and execute a control plan, and generate command signals for the actuators to generate motion. These actuators subsequently move (parts of) the robot in order to achieve the control objective. The system being controlled is often referred to as the plant. [Vel99; Lob+14]. Some common sensors and actuators are treated below. Information on various

[2] https://www.opportimes.com/the-da-vinci-surgical-system-from-intuitive-surgical/

control principles can be found in section 4.
When a human and robot are coupled, such as in a wearable robot, it is possible to identify three interactions between the human movement control system and the artificial movement control system (Figure: IV.4). Those are **motion intention detection**, **user-robot interaction** and **provision of sensory feedback**.

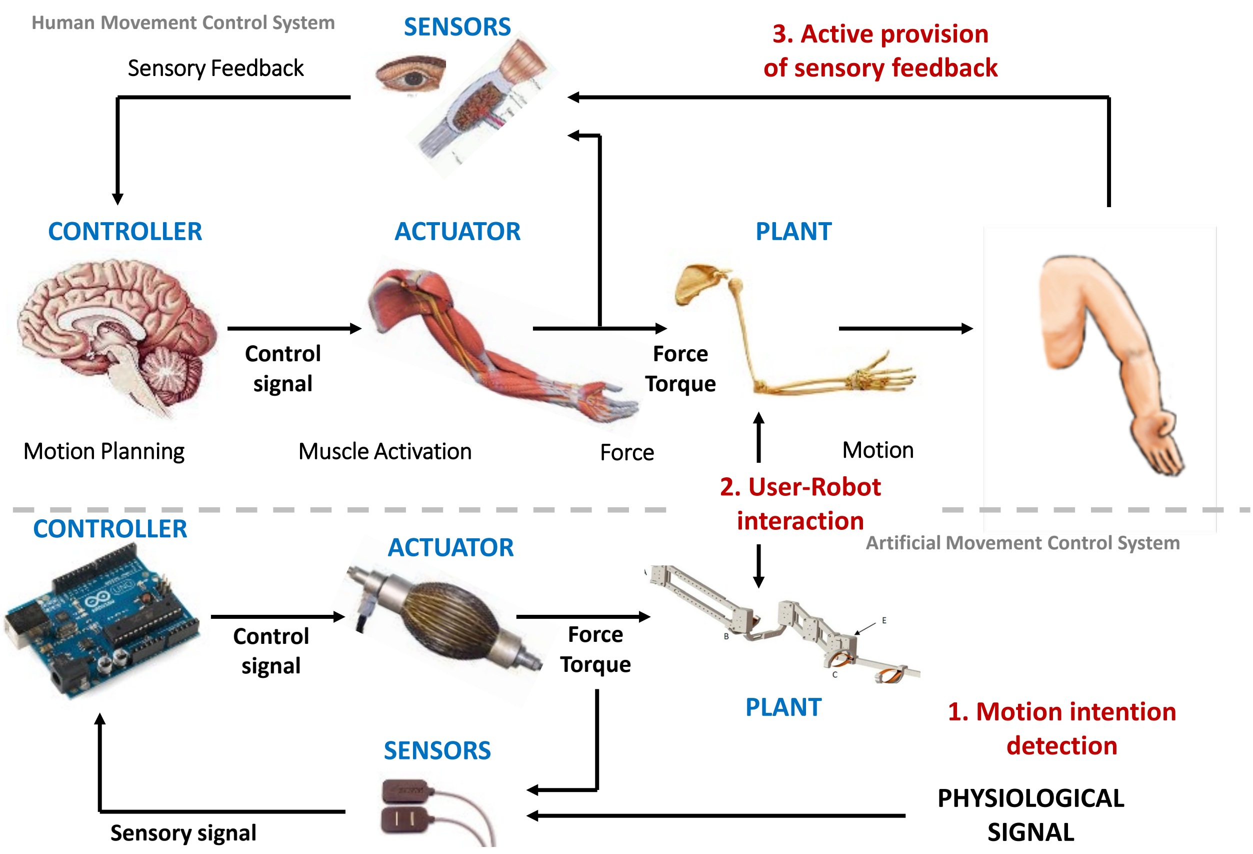


Figure IV.4.: *The figure shows in parallel the two movement systems (artificial movement control system and human movement control system.). The two system architectures are symmetric and decomposed to the same basic elements. Additionally, the three forms of interaction between the two systems are shown. Modified from [Sensory feedback in artificial control of human mobility Peter H. Veltink].*

Motion intention detection refers to the acquisition of the intent of the robot's user through sensors. Of course, sensing is not limited to just intention detection, as a robot might also sense something from the environment that is not the user, and act accordingly. The example that is used here would be a patient with Duchenne muscular dystrophy (DMD) or stroke, who is trying to control a robotic exoskeleton in order to perform an intended movement [Niz+21]. The method of acquiring, processing, and transmitting control commands based on user intention is also referred to as a control interface. For each component of the human movement control system, there are several methods of interfacing [Lob+14]. Brain activity (interfacing with the controller) can be recorded via the means of electroencephalography (EEG), magnetoencephalography (MEG), functional magnetic resonance imaging (fMRI), and near-infrared spectroscopy (NIRS). These are also commonly referred to as Brain-Computer Interfaces (BCIs). One level down the neural path we find the muscles (interfacing with the actuators). Common ways to interface with the muscles include electromyography (EMG), magnetomyography (MMG), and sonomyography (SMG). Subsequently, the limb (interfacing with the plant) can be used for acquiring motion information (position, velocity, or acceleration), or force/pressure, in order to interface with a robot.

User-Robot Interaction refers to the physical interaction between a robot and the user. In the case of an exoskeleton (example of Figure IV.4), the exchange of mechanical power happens between the robot and the limb of the user. Depending on the robot and on the application this exchange may differ. In the case of a surgical robot, we have teleoperation of the tools that perform the surgery by a surgeon sitting in front of a screen (see example in the following section). Depending on the type of exchange, there are different methods for interacting with a robotic device and ensuring stable and safe control. More details on this part in Section 4, and chapters 2, and 3 of this textbook.

Provision of sensory feedback refers to the sensory feedback that loops back to the human due to the exchange of mechanical power between the human and the robot. Depending on the condition of the human user this may be impaired and in need of enhancement (for example in the case of amputation, there can be a restoration of the sensory pathways via Targeted Sensory Reinnervation (TSR)[Gar+21]. This sensory feedback closes the human control loop and assists the human controller (brain) in planning the next motion.
In the same way, a robot might experience a malfunction in any of the components identified in Figure IV.4. Humans may also experience impairment to one or more of the components of the human movement control system [Niz+21]. Examples of that include impairments at the controller level (i.e stroke), at the actuator level (i.e. muscular disorders), at the plant level (i.e. skeletal deformities or trauma), and at the sensory level (i.e.blindness or hearing loss). Such impairments may lead to adaptations in the controller due to brain neuroplasticity and different control strategies of human motion.

Common robot sensors
Like humans, robots can contain a plethora of sensors that can be used to obtain information about the state and environment of the robot. Three frequently-used sensors will be covered in this section: rotary encoders, force-torque sensors, and inertial measurement units (IMUs). Each of these can be used to provide information about the state of the robot, and/or its interaction with the environment.

Rotary encoders
Human muscles contain sensors that signal information about a muscle's relative length and change in length (i.e. velocity). With that, they relay information about the state of the body and the orientation of our joints to the brain (controller). In a similar way, robots may also require information about their own "body" state in order to operate. A rotary encoder provides information on the amount of rotation of a joint or a motor shaft. This information may be either incremental or absolute [Bor+96]. An incremental encoder provides information on rotations without any particular point of reference. That is, the encoder does not provide information of the orientation of the shaft that is being measured, but only on changes in that orientation. As a result, it might be required for the robot controller to set a point of reference with respect to which the changes are tracked. If not, it remains unknown to the robot what the absolute orientation of the shaft is based on the encoder signal. To measure both the amount of rotation as well as the direction of rotation, an incremental encoder must provide two output signals. These are often two binary signals ("high" and "low") that are 90 degrees out of phase, see Figure IV.5. Each change in signal ("high" to "low" or vice versa) corresponds to a certain amount of rotation. The phase difference is required to obtain directional information, which can be determined from which channel (A or B) falls when the other channel is high.

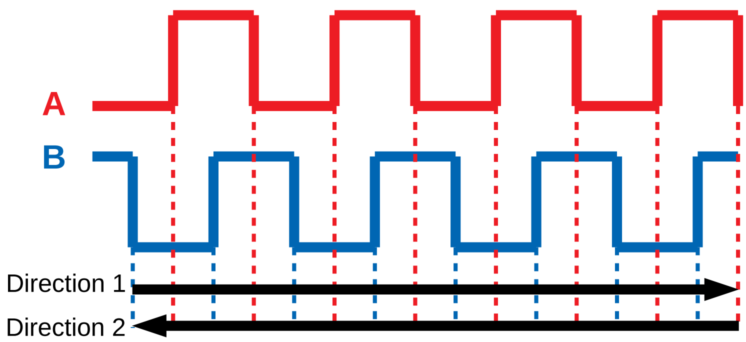


Figure IV.5.: *Example output signals of an incremental encoder. Two signals are required to detect both the amount of rotation as well as the rotation direction. Each signal change (dashed lines) corresponds to the smallest measurable rotation. Rotation direction 1 occurs when A falls or rises when B is high or low, respectively. The opposite rotation occurs when B falls or rises when A is high or low, respectively.*

In contrast, an absolute encoder provides a unique output signal for each possible (discrete) angle of rotation. The sensor's internal mechanical construction often contains a code disk that helps generate an optical or electrical signal that is unique with every rotation angle. The advantage of an absolute encoder is that it maintains its unique output signal after a loss of power, whereas an incremental encoder would need to redetermine the point of reference. Disadvantages are the limited amount of rotation for which a unique signal can be produced (e.g. 360 degrees), as well as the higher costs of production relative to incremental encoders, due to their more complex internals.

Force-torque sensors
A force-torque sensor generates an electrical output signal that depends on an applied mechanical load. Such a sensor generally contains force-sensitive resistors such as strain gauges, which deform upon experiencing load. When a strain gauge deforms its electrical resistance changes, which in turn can influence an electrical output signal [Yan91]. A force sensor often has a sturdy metallic body that can slightly deform under a load, and return to its original shape when the load is removed [SF19]. Integrated strain gauges are then used to measure the deformation. The electrical output signal can be related to an exerted force and/or moment by means of sensor calibration. If a known load is applied to each axis of the sensor and the output electrical signal is measured, a relation can be derived between the two. Ideally this relation is linear, such that a calibration matrix can be obtained that can be used to transform the measured electrical signals into applied forces and moments. To calibrate moments, not only the amount of applied force must be known, but also the distance between the point of force application and the origin of the sensor. *Inertial measurement units*
An inertial measurement unit (IMU) consists of a collection of sensors, being accelerometers to measure acceleration, gyroscopes to measure angular velocity, and magnetometers to measure the orientation relative to the earth's magnetic field [Ahm+]. The information that each of these sensors provide can complement each other. For example, to determine linear acceleration without influence from the acceleration caused by the gravitational component, it is required to know the orientation of the sensor relative to the gravitational vector, such that this component can be subtracted. A Kalman filter-based approach is often used to realize fusion of these sensory signals. An IMU placed on a robot segment may provide information about how this segment is positioned in global space.

Common actuators
In the most general sense, an actuator converts some form of energy into linear or rotational motion. There are three common actuator modalities in robotics, being electric, hydraulic, and pneumatic. Both hydraulic and pneumatic actuators work by applying pressure to a piston, or by causing a shape change in material through which the pressurized substance flows. For hydraulic actuators the pressurized substance is often oil, whereas it is air for pneumatic systems. As of such, these actuators require a compressor and a tank. Electric actuators may use a combination of spindles and magnets to create a spinning motion, such as occurs in common direct-current (DC) motors. By moving current through a spindle wire a magnetic field can be created. When a magnet is present in the vicinity of the spindle, a magnetic force will be exerted on it. This can be used to create a rotating motion of an axis onto which the spindle is attached. Motion would stop once e.g. the north pole of the spindle is close to the south pole of the magnet. By changing the direction of the current flow through the spindle, its polarity can be changed and the motion may continue. DC motors are mechanically constructed in such a way that the current flow changes direction at the correct timing, allowing the rotation motion to continue. The benefit of electric actuators compared to hydraulic and pneumatic ones is that no compressor tank is required, though there is still a necessity for battery packs to ensure that the robot can move freely through space without being attached to a power outlet. Also refer to section 3.6.1 in chapter 1 for more information on actuation.

IV.4. Working principles in medical robotics - case studies

Introduction
This section explains some of the control working principles behind the state-of-the-art robotic devices used in the medical domain. The robotic devices presented in this section relate to minimally-invasive surgery, gait rehabilitation, and assisted living. The applications in question are used to highlight and motivate the benefit of the treated control methods. In all of the following sections, a human is considered to be able to make physical contact with a robot or its interface.

Use-cases

Minimally-invasive surgery
Minimally-invasive surgery, specifically laparoscopy, involves operating on a patient through one or a few small incisions using specialized surgical instruments. It often involves a camera to see inside the patient through one of the openings. This is contrary to open surgery, where larger incisions may have to be made to reach the surgical area, and to obtain a direct view of it. As laparoscopy results in smaller openings into the body, it may provide better patient outcomes and a faster recovery. Laparoscopic surgery has shown to improve quality of life outcomes such as physical functioning and pain, compared to the open surgery counterpart, for various interventions such as cholecystectomy, splenectomy, and esophageal surgery [Vel00]. Laparoscopic surgery often involves two surgical tools and a camera, the latter being held by an assistant. As a result, the surgeon is not in direct charge of the field of view. The surgeon and the assistant need to communicate with regard to the camera view, and the surgeon needs to deal with unanticipated hand movements of the assistant, for example caused by natural tremor, fatigue, or involuntary patient movement (e.g. heartbeat). Finally, there is a loss of depth perception when the camera images are viewed on a tv screen [SW94]. This motivates the use of a robotic system that replaces the assistant, puts the surgeon in charge of both the surgical tools and the camera, ensures a steady camera, and enables depth vision.

Da Vinci surgical system: motion-scaled position control
The da Vinci surgical system (Intuitive, Sunnyvale, California, USA) is a robotic device for use in minimally invasive surgery. It consists of three main components: 1) a set of robotic arms, of which one is equipped with a camera and an endoscope, and up to three are equipped with other minimally invasive surgical instruments, 2) a console with controls for the surgeon to operate the robot, plus a viewport for the surgeon to view the surgical field through the robot's camera, 3) a vision cart for camera image processing, which converts the camera images from the robot into a 3D view of the surgical area at the console. In the following text, the focus is on the first two components. At the console, the surgeon can grasp and hold two "master arms" that measure the hand and arm motion of the surgeon, see Figure IV.6. Each master arm controls the seven degrees of freedom of a single robot arm: three for instrument translation, three for instrument rotation, and one for gripper opening and closure. Furthermore, various foot switches are available to the surgeon that enable modifying the control coupling between the master arms and the robot arms.

A master arm can be assigned to control one or two robot arms, but not two arms at the same time. When one master arm is assigned to two robot arms, a switch in control between robot arms is required. Upon control switch, one robot arm stops being controlled, and another begins being controlled by a master arm. Whenever the control of one robot arm stops, it will remain stationary. It is furthermore possible to completely decouple the master arms from all robot arms. This allows the surgeon to move without affecting the movement of the robot. This might be required when the surgeon wishes to obtain a more comfortable body pose from which to continue operation [BM03]. When re-engaging the control coupling between master arm and a robot arm, the movement of the robot arm may continue from the position where it was left. Translations of the robot are controlled relative to the point in space on the master (operator) side at which the control was engaged, whereas rotations and the gripper closure [Sur04] are mapped as absolute. For translations, this means that the surgeon can choose any (translational) position of the master arm to engage control, and the robot will track translations from that point onward. For rotations, however, the master arm and the robot arm must have the same orientation when engaging control. Decoupling the control between master- and robot arms may lead to a discrepancy between the master arm's pose and that of the robot arm. To restore this, the surgeon may be told by the system to let go of the master arms, and the system will use actuation to restore the master arm orientation to be the same as that of the robot arm [BM03].

For the gripper, too, a matching is required before the control can be continued. When decoupling a master arm from a robot arm, the hand pose at the master could correspond to an opened gripper, whereas the actual gripper at the robot arm is closed. If the gripper would behave like the translational degrees of freedom, relative to the point of the

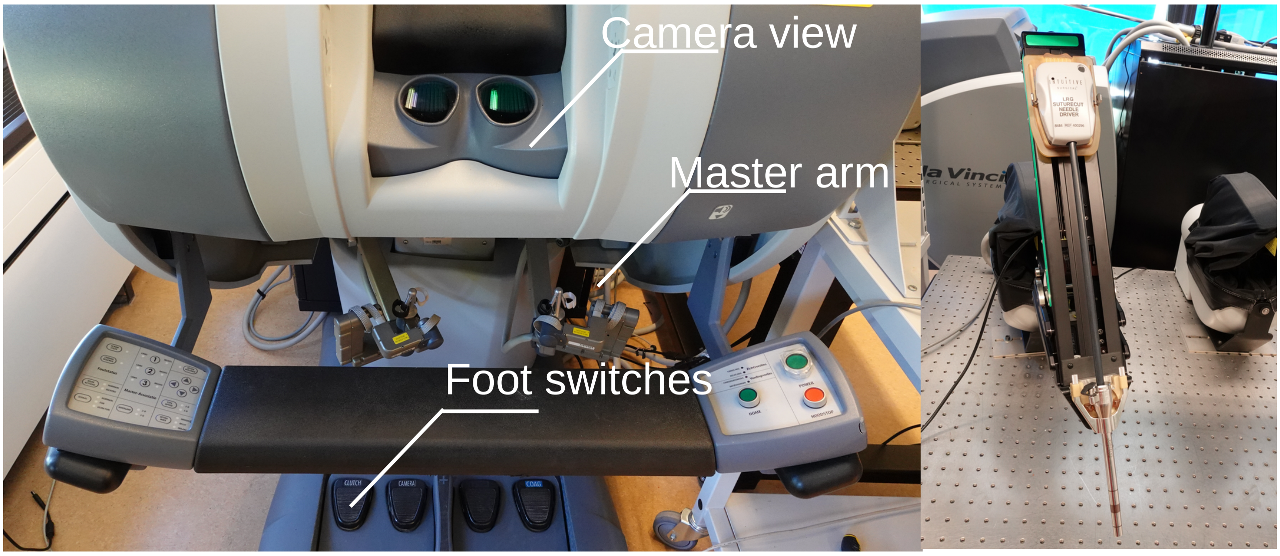


Figure IV.6.: *Layout of the DaVinci robot. Left: surgeon interface with the stereoscopic viewport, the master arms, and the foot switches for controlling the coupling with the robot arms ("clutching"). Right: One of the robot arms (no gripper attached).*

master at which the controls were re-engaged, then the surgeon will likely meet the end of the hand range of motion. That is, it would be difficult for the surgeon to open the robot's gripper by opening their hand, when the surgeon's hand is already open. The mapping is therefore absolute. However, if the system would allow re-engaging the controls at any time under an absolute mapping, and the gripper would move to the same position as indicated by the master arm. In such a case, re-engaging the controls could lead to an unintended opening or closure of a gripper. To prevent such potentially dangerous situations, the surgeon is required to first match the gripper state at the master arm with the true gripper state, before control of the robot arm can be continued. In case the master- and gripper poses accidentally match, a finger movement has to be made by the surgeon that goes all the way to the end of the range of motion of the finger joint at the master arm, before moving to the matched state. This ensures that an intentional voluntary action of the surgeon has been made before the control of the robot arm is continued.

To ensure a natural control of the instruments held in the robot's arms, the surgeon should not have to worry about controlling each of the robot's joints individually. Instead, it should be possible to think about moving the instruments as though they were in their own hands. In other words, the control signals of the surgeon to the robot should occur in the end-effector space, rather than in the joint space. However, this still requires the robot joints to be controlled in such a way that it results in the desired end-effector motion. This is taken care of by algorithms running on the robot, that transform the endpoint motions as intended by the surgeon to joint specific movements of the robot, to realize those endpoint motions.The techniques to move from end-effector space to joint space and vice versa are called inverse and forward kinematics, respectively.

Inverse kinematics involves the calculation of the joint angles in a kinematic chain that are required to obtain a certain desired end-effector position and rotation, relative to the start of the kinematic chain. The start of the kinematic chain is often the base of the robot arm in global coordinates. If the kinematic chain of the robot would contain more than 6 joints, there are infinitely many joint configurations possible that result in the desired end-effector position and orientation. This is because the robot would then have more degrees of freedom than there are present in physical space (3 translational axes, 3 rotational axes). However, for a 6-degrees-of-freedom robot, such as each individual arm of the da Vinci robot (ignoring the gripper), a single analytical solution can be found.

Forward kinematics is the reverse process. It involves the calculation of the position and rotation of the end-effector, given the joint angles of a kinematic chain. This is relevant because sensors in a robot often measure information from the robot's individual joints, and not directly from its end-effector. Forward kinematics are therefore required to obtain a robot's actual end effector position and orientation using sensor measurements at a joint level. A simple example in two-dimensional space using Euler angles is given in Figure IV.7. Other principles exist that are better suited for inverse- and forward kinematics in three-dimensional space, including quaternions and screw theory, as they prevent the rotation order dependencies and Gimbal lock that may occur when using Euler angles in three-dimensional space. The

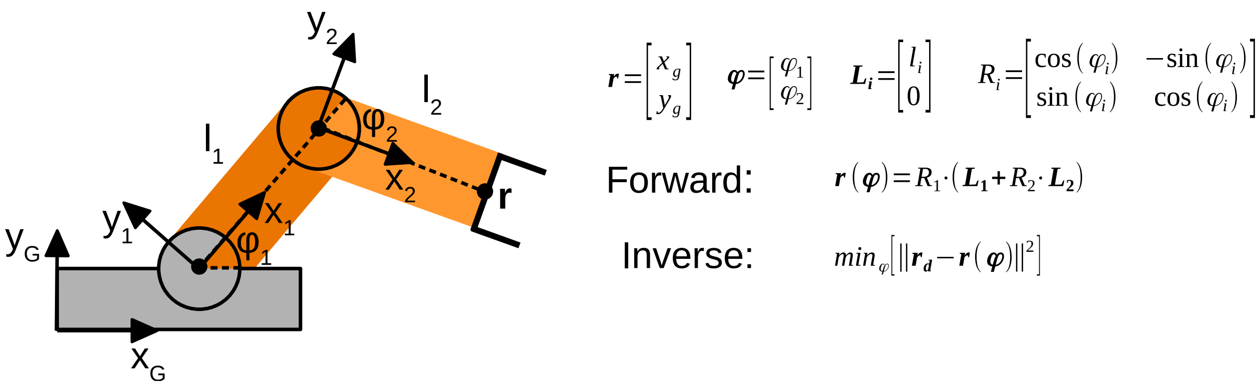


Figure IV.7.: *Two-dimensional robot arm example. The end-effector has a position r in the global coordinate frame located at the robot base. The first joint is rotated by an angle $\phi 1$ relative to the global coordinate frame. The second joint is rotated by an angle $\phi 2$ relative to the coordinate frame of the first joint. The length of the first and second arm segments are l1 and l2, respectively. For each joint i = 1, 2 a rotation matrix Ri can be defined. Forward kinematics involves computing the global coordinates r as a function of generalized coordinates ϕ, by using the rotation matrices to rotate each arm segment. Note that the second segment L2 is not only rotated by R2 but also by R1, as it is serially connected to the first segment. Inverse kinematics involves finding ϕ such that the resulting r is equal to some user-defined desired end-effector position rd. Finding ϕ may involve an iterative minimization procedure, as it is often not straightforward to find an analytic solution.*

following steps provide a way of transforming the motion of the surgeon into the motion of the robot:

1. The surgeon performs a motion while holding the master arms in the console, as if holding the surgical instrument.
2. Sensors in the console measure the translations and rotations of the master arms, as made by the surgeon.
3. Measured motion is used to update the desired end-effector position of the robot arm.
4. Inverse kinematics is used to find the corresponding desired joint angles.
5. The error between the desired joint angles and the true (measured) joint angles can be fed into a controller, to produce motor output in an attempt to minimize this error.

A basic overview is shown in Figure IV.8. The da Vinci robot utilizes motion scaling between the movements of the surgeon and the movements of the robotic arms. This way, a movement of 1 cm at the surgeon's hand may result in less than 1 cm of movement at the robot's end. This may enhance precision, as all motions are downscaled, including the effects of natural hand tremors on the surgical tools.

A controller that can be used to drive the error between the true and desired joint angles to zero is a PID controller. Such a controller takes as input an error term, such as the difference between a desired- and a true (measured) joint angle, and computes a control signal with the aim of minimizing that error. In doing so, it utilizes signals that are proportional, integral, and differential terms with respect to the input error. A basic overview is shown in Figure IV.9.

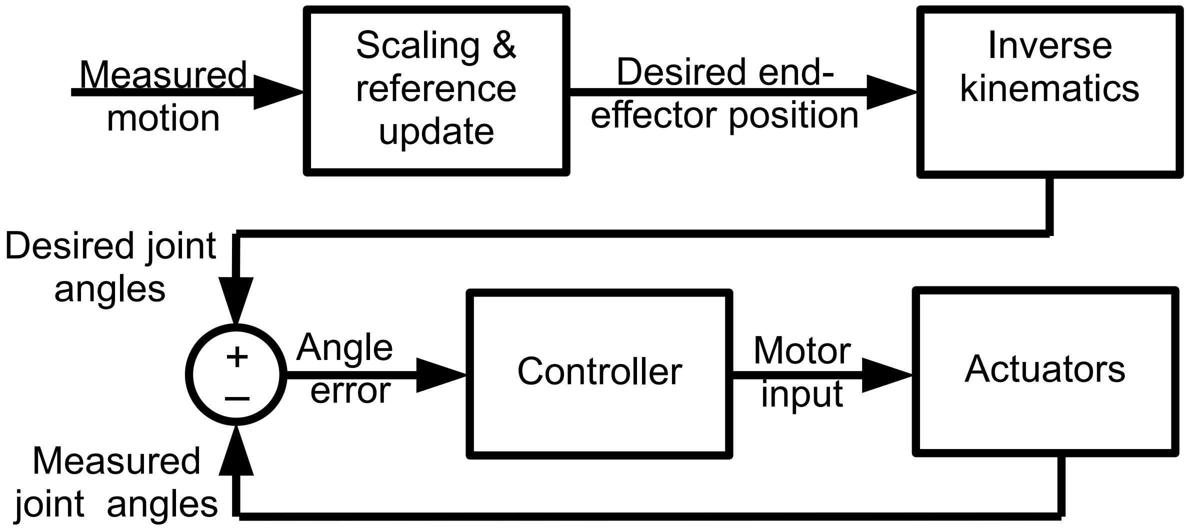


Figure IV.8.: *Hypothetical control overview of a DaVinci robotic arm. Motion performed by the surgeon is measured, scaled, and used to update the desired end-effector position. Inverse kinematics is used to calculate the corresponding desired joint angles. The error between the measured joint angles and the desired joint angles can be fed into a controller that tries to mitigate this error by generating control signals for the actuator.*

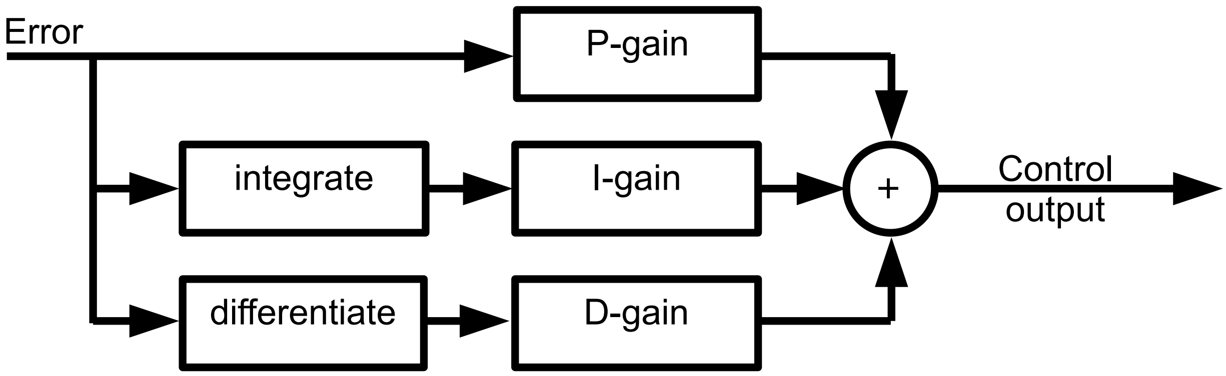


Figure IV.9.: *PID controller overview in so-called parallel form. An input error signal is integrated and differentiated, after which all three signals are multiplied with their own gain, and then summed. The gain values determine the controller behavior. Several variations to the schematic are possible, such as by shifting the P-gain to after the summation, to obtain the so-called standard PID form.*

The values chosen for the gains determine the controller behavior. The proportional gain makes the controller output behave "spring-like" with respect to the error, in the same way a spring stiffness is a proportional constant between spring deflection from its resting length (i.e. the "error") and the resulting force (i.e. the "output"). The larger the error, the more aggressively the controller will respond with its output in opposing the error. In a similar way, if the error is small, the controller might hardly respond. This makes it challenging for a pure proportional control to correct small errors. The integral term may help deal with this, as it causes corrections based on the cumulative error. Even if only a small error exists, its accumulation over time may help the controller drive the error to zero faster than a pure proportional controller. In extreme cases, the integral term may cause a so-called windup. If the error suddenly increases because of a large change in the reference position, the integral term may quickly become large. This may lead to an over-aggressive response of the controller, which in turn may result in strongly driving the error to zero followed by an overshoot, leading to a new error. Finally, the differential term introduces a form of damping, and may reduce oscillations in the control output. However, it is also sensitive to noise in the error, as differentiation is involved which enhances noise effects in the signal. Proper settings of the gain values are important for stable functioning of a PID controller. The values are case specific, and depend on the desired behavior and the dynamics of the robot. Various analysis methods such as root-locus analysis exist to determine appropriate (ranges of) gains and ensure stable system behavior.

Human-robot interaction: who is in charge?
In the da Vinci robot mostly only one-way communication occurs. The user tells the robot what to do, and the robot follows the commands. The user is fully in charge, and the robot's controller does not allow it to autonomously take over from the user. However, some robots need to monitor a person's motion, and possibly influence that motion. This implies two-way communication: not only from human to robot, but also from robot to human. The robot's output may influence the subject who is delivering inputs to the robot controller, and with it influence the input to the robot. As a result, the subject may not be fully in charge. The robot's controller allows it to intervene and (partially) take over the motion of the subject.

One might distinguish varying degrees to which each entity (human, robot) is in charge. In the most extreme cases, either the subject or the robot is fully responsible for the generation of motion, and the other only passively follows. A rehabilitation robot that assists a human wearer in the execution of a motion is such a device in which varying degrees of "leader" and "follower" may occur. If the rehabilitation robot would be fully in charge and the human subject is only passively following the robot's motion, one might question whether this leads to effective rehabilitation. It may be better if the human subject is in charge of the motion, and the robot only intervenes and corrects when it detects that the subject may not be able to complete the walking task on its own. This is also known as an "assist as needed" paradigm [Meu15], which is relevant in the following considered case.

Surgical robots vs traditional industrial robots
Surgical robotic devices were generally developed along two different routes. A first class of robotic systems took inspiration from industrial stand-alone robotic systems, characterized by high-precision, high-power, high-inertia robotic systems that are not backdrivable and often operate autonomously in a well-defined (industrial) environment. The second class of systems evolved from teleoperation systems, which are characterized by lower precision, lower torque, lower inertia robotic systems that are backdrivable, and often operate with a human in the loop in an ill-defined environment. Today's systems can be traced back to either of these classes. One key parameter corresponds to the nature of the environment the system is working in, whether this is a well-defined or an ill-defined environment. For systems that follow the industrial robotic paradigm, the environment is preferably well-defined such that the robotic tasks can be precisely specified and subsequently executed in an automatic fashion. In principle no human is needed in the process.
In practice, from a safety viewpoint a supervisory role for the surgeon is typically assumed. Typically facilities are foreseen (such as a predictive display) for the supervisor to assess the current state of the system such that he/she can easily interpret what is going on and whether the actions are in line with the expectations. In this type of systems the quality of the intervention often relates to the precision with which a predefined path is executed[3]. Hence special care is paid to equip these robotic devices with high-precision sensors and foresee stiff, rigid structures that maintain their pose independent of external forces that apply. Geared transmissions are commonly used in these devices as the transmission helps attain high measurement resolution and allows delivering large forces upon the environment. Applications like milling, drilling or other high-force gestures one can find in orthopedics commonly require such high forces/torques and for these applications one typically looks quite fast at industrial-like surgical robots. The consequence of using stiff structures with large reduction ratios makes that the overall inertia of these robots is typically large. This means that one may restrict oneself to low accelerations and low speeds which, from a safety viewpoint, may need to be enforced even as a lot of kinetic energy can be stored which could cause havoc if unleashed. The large reduction ratios mostly make the robot non-backdrivable. This means that if an external force is applied on the robot structure it most likely will not cause any noticeable motion of the robot. The latter is able to maintain its pose. This is a positive property if one wants to ensure high precision in the presence of external disturbances, but is disadvantageous if the executed trajectory is inappropriate as external forces will not help correcting for it. Also in the case of a power failure it may be difficult to remove the robot from the patient. Dedicated means need to be foreseen to avoid this problem.
Surgical robots that follow a teleoperation philosophy have completely different properties. These systems are designed to be light-weight, backdrivable and low-force. The overall precision that these devices can attain is typically much

[3] Although this assumption is hard to maintain in environments that vary significantly during and due to the surgical act itself. In principle such situations teleoperation-based approaches may be more suitable (in a sense the environment here quickly becomes an ill-defined one).

lower than that of their industrial counterparts. This nevertheless does not mean that these devices cannot be used for precise surgical tasks. On the contrary: teleoperation systems derive their precision from involving a human in the loop. The human expert closes in a sense the control feedback loop. They observe the position of the robot relative to the tissue and will derive appropriate motion commands to reduce position errors. As a result the precision of the combined 'man-machine' system is to be assessed. This can be even higher than for a typical industrial-robot inspired counterpart. This is for example the case for a robotic system that was developed for vitreoretinal surgery [Gij+13] and that allowed cannulation of retinal vessels with diameters below 100 micron and subsequent injection of anti-thrombolytic agent [Gij+18] offering novel treatment options that simply cannot be safely attempted by bare hands.

Rehabilitation of gait

A stroke may occur when blood flow to the brain is impeded. This can happen because of a blood clot that blocks blood flow, or even a rupture in a blood vessel. As a result, the brain does not receive sufficient oxygen to function, which may lead to permanent damage to brain tissue. Among various movement deficits, stroke can lead to difficulties in walking and maintaining balance [PEG11]. Some degree of walking recovery often occurs spontaneously in the months following the stroke [Cra08]. It has been suggested that this recovery may be stimulated by intensive training of movements [Kwa+04; Col+13], especially within the first few months after the occurrence of the stroke [KK13]. A safe gait rehabilitation environment is required in which a stroke survivor can intensively train. Classically, a training session involves a stroke survivor practicing walking while being supported by one or multiple therapists. The therapists help in maintaining balance and in moving the person's legs. This is labor intensive and time consuming for the therapists, such that the training time for the subject may be limited by the therapists' physical capabilities [Sch+07] and by hourly costs [Hes+03]. Furthermore, the assistance provided by the therapists is not standardized, and may be inconsistent across therapists and across training sessions. Finally, for intense training it is the subject who should be attempting to perform the movement, not the therapist. Ideally, assistance is only provided during the gait cycle when needed by the subject. It may be difficult for a therapist to manually deliver such "assist as needed" therapy. A robotic device that takes over the repetitive and labor-intensive work from the therapists could tackle these issues. Here the LOPES-I [Ven+07] and LOPES-II [Meu15] devices are considered. These exoskeletons can be worn by a subject around the hip and legs, such that they can exert forces on the subject's body to provide support and help with the walking motion. To deliver assistance as needed by the subject, the robotic device must be able to:

1. Measure or estimate the interaction that exists on the interface between the robot and the subject. This interaction may be in terms of movement or force.
2. Control the interaction to encourage active participation of the subject, and provide assistance as needed by the subject to successfully execute the walking task [EBR05].
3. Potentially control the interaction to compensate for gravity effects and the robot's inertia. If the subject is in control of executing a movement, they should not feel like they have to drag along a heavy robot construction. This would only impede the movement and hamper rehabilitation.

The LOPES-I and LOPES-II rehabilitation robots achieve the above in different though closely related ways. Specifically, the former uses impedance control, whereas the latter uses admittance control. Whereas impedance control requires measurements of motion, an admittance controller requires measurements of force. These control topics are treated in more detail below.

Haptics

Haptics relates to the study of touch, and to interfaces that introduce a sense of touch (also see chapter 3). Haptics may enhance the user experience when the user is in physical contact with a robotic device. For example, force feedback may allow a surgeon to feel what they are doing while inside a subject's body [Sta+15]. In case of the aforementioned rehabilitation robots, a safe "motion tunnel" may be rendered to guide a subject's movements while inside a rehabilitation robot, and prevent undesired movements. Haptics involves sensing or estimating kinematic and dynamic information of a robotic system, and using that information to manipulate the forces exerted on a human through the contact interfaces with the robot. Control techniques can be used to influence what a human feels from the robot.

LOPES-I: *Impedance control*

An impedance controller takes as input a measured motion, such as a linear displacement or an angular velocity, and responds by generating an output force or torque. The controller allows generating forces that are consistent with certain user-defined virtual dynamical properties (stiffness, damping, and mass). This way, for example, it is possible to generate a spring-like force surrounding a desired reference joint angle. The further a human subject would try to move the robot joint away from the reference angle, the more the robot may resist that motion. The reference angle does not have to be stationary: it could also change over time to create a reference joint trajectory. This may be used to create a guiding force surrounding the legs of a subject who is re-learning to walk. In the following text, it is assumed that the measured input motion is a velocity, and the resulting output is a force. Together, velocity and force conjugate to power, i.e. energy per time unit, which is closely related to the (required) power specifications of the robot's actuators.

Before the robot controller can respond with a force, it first needs to measure a motion between the robot and human subject. As a result, there needs to be some form of measurable compliance on the human-robot interface. This compliance can exist in the robot construction and/or in an actuator. Consider an actuator that is not backdrivable and rigidly coupled to a human joint Figure IV.10A. There is no compliance anywhere in the construction, such that it is impossible for the human to create a measurable motion that can function as input to the impedance controller. If instead a backdrivable actuator is used, the human subject can push against the internals of the actuator Figure IV.10B. If this motion can be measured in the actuator, it may function as an input signal to the impedance controller. However, depending on the actuator construction, it may be undesirable to move against the actuator's internal mechanism, as it may strongly resist motions generated by the human coupled to it. That is, the actuator construction itself may have a high impedance, such as due to internal gear friction. Yet another solution, as utilized by LOPES-I, is to introduce series-elastic actuators (see Section 3). The actuator itself may not be backdrivable, but a compliant mechanism (spring) is present between the actuator and the human Figure IV.10C. The spring deflection can be measured using sensors and function as an input to the impedance controller.

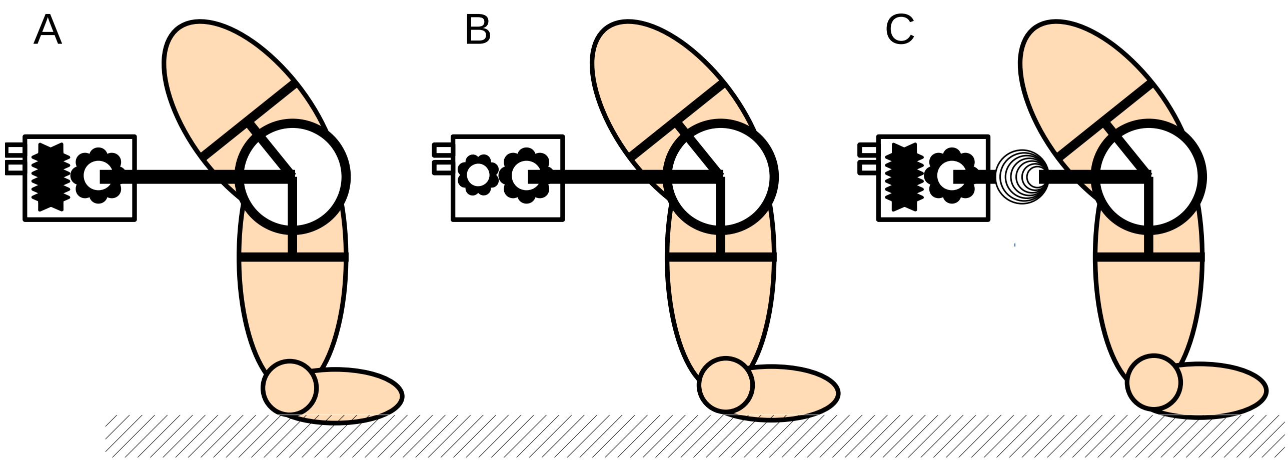


Figure IV.10.: *Actuator coupled to a human joint. A: Non-backdrivable actuator rigidly coupled. No compliance is possible. B: Backdrivable actuator rigidly coupled. Measurable compliance possible in the actuator. C: Non-backdrivable actuator compliantly coupled through a spring. Measurable compliance possible between actuator and human joint.*

In the impedance controller, the conversion of the measured motion to the desired force occurs through a set of parameters that together make up the virtual impedance. This virtual impedance is the impedance to be experienced by the human subject when interacting with the robot. It may be characterized by gains such as a spring stiffness, a damper constant, and a mass constant. A spring constant relates a displacement to a force, a damping constant relates a velocity to a force, and a mass constant relates an acceleration to a force. Though it was assumed only velocity was measured from the compliance, position and acceleration may be obtained through integration and differentiation of the velocity signal, respectively. Finally, a therapist might set reference values that may be included in the virtual impedance, such as a desired joint angle. This leads to a reference force, which may be regulated by a force controller, and executed by an actuator. The force generated by both the actuator and the human work together on the compliant interface between human and robot. This in turn leads to a measurable motion. It may also lead to a measurable (or estimable) force. For example, for a physical spring, the deflection is directly proportional to the force through the spring stiffness. This force may function as a feedback signal for the controller generating the force. The following impedance control diagram may be drawn:

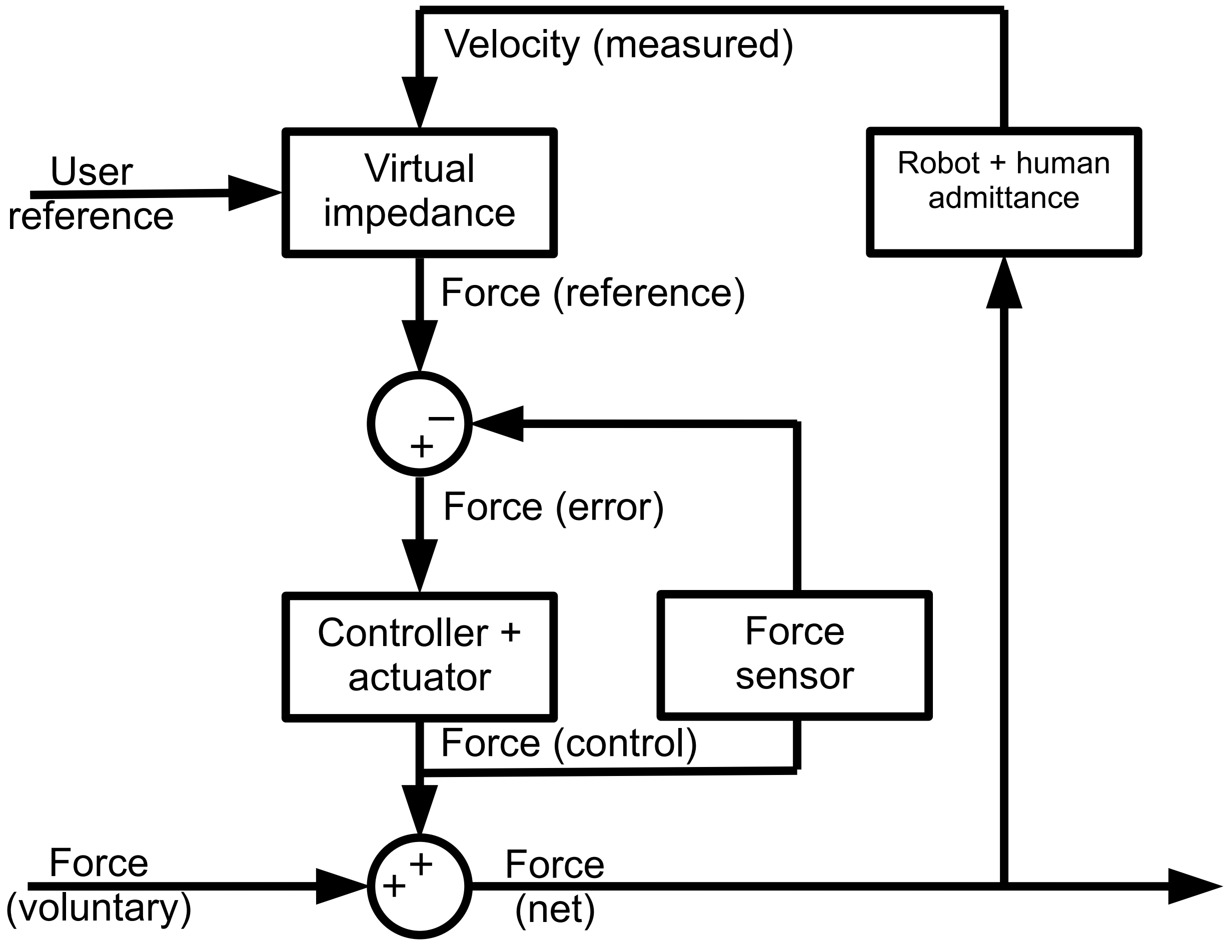


Figure IV.11.: Impedance control overview. A virtual impedance takes as input a measured velocity, and generates a reference net force. This force is to be experienced by the human in contact with the robot. The force is passed to a controller that makes an actuator generate the reference force. The force generated by the actuator and any voluntary force generated by the human in contact with the robot together work on the load that exists between human and robot, such as a spring. The force exerted on this interface results in a measurable movement (velocity), which in turn functions as input to the impedance controller. A user may modify reference parameters in the virtual impedance, such as an angle or velocity around which to operate. Note that the actuator itself might be moved to the robot + human admittance block, in case the actuator itself is the (backdrivable) compliant interface at which the velocity measurement occurs. Also note that in the schematic it is assumed that the controller force can be measured, to serve as a feedback signal. Impedance control without such force feedback is also possible.

Limitations

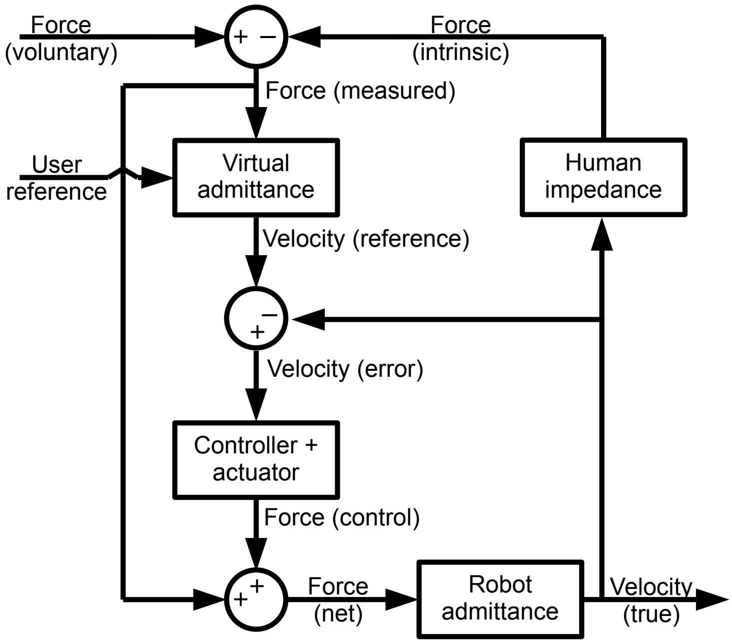
Because an impedance controller attempts to control the force on the interface between robot and human, the controller may have trouble with scenarios in which there is very little interaction, such as free motion. Imagine that the controller attempts to deliver a certain force on the interface, while there is no human body present for the robot to act on. It would be very difficult for the controller to generate the required force, and it may output very high accelerations in an attempt to do so. Furthermore, it can be challenging for an impedance controller to render very stiff interactions [LL03], like a force that makes the user feel as if they were touching a rigid wall. The latter is impossible because of the compliance that naturally exists in the robot construction, which is often lower than that of a solid stationary object. Finally, the forces experienced by the human subject are influenced by the mechanical properties of the robot [AH99]. For example, if the force generated by the actuator would have to pass through a heavy mass in order to reach the human, then this mass affects the force transfer. That is, the impedance of the robot construction itself is also present in the force that is eventually experienced by the human. This may make it challenging for the controller to ensure that the human experiences purely the virtual impedance. For an impedance controller it is therefore beneficial if the robot construction itself is lightweight and has a low impedance.

LOPES-II: *Admittance control*
An admittance controller takes as input a measured force or torque, and responds by generating an output motion such as a displacement or an angular velocity. The controller allows generating motion that is consistent with certain virtual dynamical properties. For example, the person delivering a force against the robot may experience the motion as if pushing against a spring, or against a light mass, even if the robot itself is heavy. Such virtual dynamics may be used to prevent the legs of a walking subject from moving too far or too fast in a certain direction. The admissible motions may change over time, making it possible to create reference trajectories.

Before the robot controller can respond with a generated motion, it first needs to measure an interaction force on the interface between the robot and the human subject. It is therefore beneficial if there exists a rigid coupling between the user and the force sensors used to measure that force. Any (unknown) compliance between the two may hamper the measurement of the contact forces. This is in contrast with the aforementioned impedance control setup, where some form of compliance is required between human and robot. It is furthermore beneficial if there exists a rigid coupling between the force sensors and the actuators of the robot, as any compliance would limit the maximum forces that the robot can deliver onto the subject. A serial chain is only as stiff as its weakest (most compliant) link. The setup may be as in Figure IV.12, but with the compliant spring replaced by a non-compliant force sensor. As long as the interaction force can be measured, an admittance controller may realize motion even if the actuator is non-backdrivable, and the coupling between human and robot is non-compliant.

In an admittance controller, the conversion of the measured force to the desired motion occurs through a set of parameters that together make up the virtual admittance. This virtual admittance is the admittance to be experienced by the human subject when interacting with the robot. As in impedance control, the virtual admittance, too, may be characterized by gains such as a spring stiffness, a damper constant, and a mass constant. For example, dividing the measured force by a virtual mass may yield a desired acceleration. This in turn may be integrated to yield a desired velocity. A therapist may also set reference values in the virtual admittance, such as a reference joint angle around which the controller should operate. The output of the virtual admittance is a reference velocity, which may be regulated by a force controller. The force generated by both the actuators and the human together work on the robot construction, which may move as a result, leading to a measurable motion (velocity). This motion is applied by the robot to the human body, which in turn results in an intrinsic (passive) interaction force between robot and human. This force, together with the voluntary force generated by the human, is in turn measured by the force sensor, and functions as an input to the virtual admittance. The following admittance control diagram may be drawn:
Admittance control vs impedance control
Admittance control always requires force sensing, and preferably stiff mechanics of the robot to ensure proper sensing of the interaction force. Impedance control does not strictly require such a setup. This may impact the system design and the costs of the device, for example due to the costs of high-quality force sensors.
Impedance-controlled devices are often designed to be lightweight, to prevent the dynamics of the construction itself to be reflected in the forces that the user experiences. An impedance controller may compensate for such device dynamics by incorporating a force sensor, but if a force sensor is to be used regardless, one may also consider a construction suited for admittance control.
In general, the range of possible dynamics that can be experienced by the human in contact with the robot is higher for admittance control than for impedance control [Fau+08]. For example, admittance control may be used to render very stiff interfaces, which can be challenging using impedance control. However, if the required dynamics are obtainable by either controller, then impedance control may be more straightforward to implement successfully.

LOPES-I and LOPES-II: *Minimal impedance mode*
The goal of a minimal impedance mode is to make a robot maximally compliant ("transparent") to the person interacting with it. By using the actuators to compensate for the inertial effects of the robot, the robot might be made "invisible" to the human in contact with it. That is, if a human would physically interact with the robot, minimal force is required to move it around.
Perhaps confusingly, a minimal impedance mode can be achieved with both impedance- and admittance control. Assuming that the human coupled to the robot can experience the virtual dynamics as set in the controller, then the dynamics may be chosen such that e.g. the experienced mass approaches zero. There are limits to what can be achieved



]

Figure IV.12.: *Admittance control overview. A virtual admittance takes as input a measured force, and generates a reference velocity. This velocity is to be experienced by the human in contact with the robot. The velocity is passed to a controller that makes an actuator generate a force. The force generated by the actuator, plus any voluntary force generated by the human in contact with the robot, together work on the robot construction. This results in a motion (velocity) of the robot, with the human coupled to it. Because a human is coupled to the robot construction, movement of that construction also affects the intrinsic (passive) interaction force between human and robot, through the human impedance. This intrinsic force together with the voluntary force generated by the human is what is measured by the force sensor, and again functions as input to the virtual admittance. A user may modify reference parameters in the virtual impedance, such as an angle or velocity around which to operate. Note that in the schematic it is assumed that the generated velocity can be measured, to serve as a feedback signal to the controller. Also note that the schematic does not consider the human controller. That is, in practice the human may also adjust the voluntary force in response to a controller action.*

by modifying the virtual dynamics parameters, however. For example in the case of a virtual admittance, the human-robot system will become unstable as the virtual mass approaches zero. The division of an input force by a zero mass leads to an infinite acceleration. This would make the actuators respond aggressively, which in turn leads to high intrinsic interaction forces, which in turn are fed back into the virtual dynamics, leading to another opposite response. The result is a vibration that rapidly becomes more aggressive.

Assistance in daily care

In a classical sense, robots are specialists. They operate best in well defined environments with little unknown factors, and they work best on well-defined tasks that require minimal additional instructions and contextual cues. Robots are often designed specifically for executing a single task, and they may provide super-human performance during such operations. Compared to robots, humans are not specialists but generalists. We may not be able to deliver the precision or power of an assembly-line robot, but we are able to handle a broad range of task variations, and solve various problems using skills and knowledge obtained in previous challenges. If robots are to operate in the everyday world, and assist humans not in one, but in a variety of tasks, then robots also need to be able to generalize to some extent. As the world around us is designed by- and for humans, a generalizing robot may benefit from human-like features, such as an upright posture, locomotion, and robotic hands. A human-like appearance furthermore has benefits on a social human-robot interaction level [Pin+19].

The EVE-r3 robot [Gup+19] is a robot meant for safe human-robot interaction, for example in a care setting. The robot's tasks may include serving, feeding, moving objects, or going for a stroll with a wheelchair-bound person. The robot stands 1.80m tall, has a weight of approximately 80 kg, and has a wheeled base instead of legs.

A first step towards human-like robots in daily life is balance control. In the everyday world a robot cannot be protected from all possible uncertain disturbances. If a robot cannot prevent itself from toppling over following such a disturbance it will not be able to function. Humans can use discrete steps to recover balance during locomotion. The advantage of a wheeled base is that continuous adjustments can be made. The EVE robot maintains balance by use of model predictive control [Gup+19].

Model predictive control
Model predictive control (MPC) utilizes a model of the robot to be controlled. The model can be used to predict how the robot will respond to (known) disturbances, as well as to self-generated actions, over a predetermined future prediction interval. These predictions of the future may be used to determine what the best action is to take "now".
In balance control of the EVE robot, the model is a (linearized) inverted pendulum on top of a movable cart, see Figure IV.13. This serves as a simplified representation of the robot's total mass on top of its wheeled base. The objective of the model predictive controller is to generate optimal actions that prevent the pendulum from toppling over in the future. This may be realized by applying a moment around the rotational point of the pendulum, or by driving the cart in a certain direction.
The controller objective is reflected by a so-called objective function or cost function. This is a (quadratic) mathematical expression defined by the person designing the controller, which describes (un)desired robot behavior. The expression outputs a cost value as a function of model states and (motor) actions. For example, for balance control it is undesired that the pendulum topples over. This might be prevented by high motor output to stiffen up the pendulum's point of rotation and cart wheels, to keep everything in place. However, this may require high motor current at all times, which is undesired as well. A cost for pendulum deflection angle and a cost for motor action may together take part in the total objective function. In addition, expressions may be defined to put constraints on variables, such as the motors that cannot deliver infinite power.
The controller may then try to find an optimal solution with respect to the objective function, such that pendulum balance is maintained while delivering minimal motor output. This optimal solution involves finding the inputs to the objective function that minimize the output cost. This optimization is performed over a finite future time horizon, such that model-based predictions of the future can be taken into account in determining what is best to do at the current time step. Even though optimal controls may be determined over a future prediction horizon, computing the controls over the horizon only once does not guarantee that balance is maintained. This is because the controller cannot take into account unknown disturbances that have not yet occurred, simply because they are unknown. If the computed controller plan is not adjusted after an unexpected disturbance has occurred, a fall is still likely to occur. To deal with this, a model predictive controller only executes a single time step of the optimal controls that were computed over the prediction horizon. After that, the entire control plan is discarded, and a new plan is computed in the next time step. This makes the method adaptive to unforeseen disturbances, but also computationally intensive.

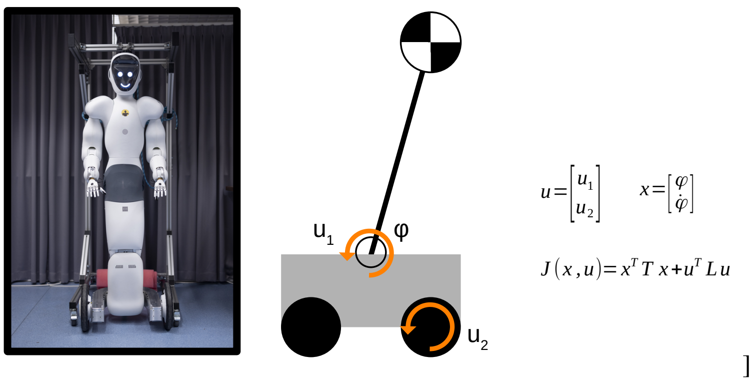


Figure IV.13.: *Left) An EVE-r3 robot, with a wheeled base and a single "ankle" joint to balance itself. Right) Pendulum on a cart to model the fore-aft balancing of the EVE-r3 robot. In a model predictive controller a quadratic cost J may be defined in terms of state x and actions u that needs to be minimized in order to realize the objective of keeping the pendulum upright. This cost determines the behavior of the robot, and how actions are distributed between for example driving and using the pendulum joint.*

Open-loop force control and backdrivable motors
Generalization in the everyday world requires frequent making and breaking of contact with the environment. While an assembly-line robot may also frequently have to come into contact with the environment (e.g. the product being assembled), it can often be assumed that there is very little variability in this contact. The interaction is always the same. As of such, a position-controlled robot may suffice, without the robot sensing anything from the interaction. The more uncertain the environment, the more relevant it becomes to know the interaction force, for example using a force sensor. However, a challenge for controllers trying to achieve a certain interaction force is contact instability. For example, when a force sensor makes contact with a rigid environment, the measured force signal will suddenly increase. A controller that tries to regulate a certain level of interaction force may respond vigorously to this sudden change. An alternative, as in EVE, is to have actuators that can reliably generate a certain output torque in an open-loop manner, without a feedback signal from a force sensor ("direct force control"). In such a case, there is no reliance on force sensors and related contact instability, under the assumption that the motor is capable of tracking the desired torque. In various scenarios such as in a healthcare setting, an assistive robot like EVE may need to frequently make and break contact with humans. If such a robot would contain actuators that are non-backdrivable, it would result in rigid and non-compliant contact with a person, and a human would not be able to push against the actions of the robot. With safety reasons in mind, the EVE robot is equipped with highly backdrivable motors that allow countering the motions of the robot.

IV.5. Stakeholders, Context and System View

The field of medical robotics is a multidisciplinary field which involves the combination of numerous technologies and stakeholders [RD16; Rei+17; Pol+13], therefore it is of great importance to involve all of them as early as possible [Cre+21]. A major point of interest in this early involvement is the need for the stakeholders to have a clear understanding of the costs and benefits of medical robotic exoskeletons and their implementation. This way, a proper mapping of their key interests and goals can be performed and the type of information they need can be defined (see also section I.2.6. User Acceptance). This is usually achieved through a stakeholder analysis. Additionally, the convergence of all these disciplines, technologies, and stakeholders increases complexity and makes medical robotics increasingly complex systems [Niz+21]. In this section we will try to explore the broad and immediate context of medical robotics as well as present a system view. Lastly, this section will discuss system design tools that incorporate the stakeholders and the system view and also facilitate multidisciplinary collaboration. With regards to the terminology used, please also refer to section 2 of this chapter.

IV.5.1. Placing Robotics within their context

The ISO 26000 (guidance on social responsibility), defines a stakeholder as an ”individual or group that has an interest in any decision or activity of an organization/project citeiso.The process of analyzing the stakeholders is iterative and dynamic, therefore, in many cases the first attempt may be incomplete (but that is ok!). The first step in analyzing the stakeholders for a medical robot,or any other device such as occupational exoskeletons [Cre+21] or wheelchairs [Alq+19], is to properly map and identify relevant stakeholders. The importance of identifying, analyzing, and managing stakeholders is multilevel. Firstly, it serves as a map of the existing key players and their competencies, influence, and impact. This way an early stakeholder analysis can help the designer leverage the capacity of these key players. Secondly, the needs and goals of these key players can be timely aligned in order to achieve maximum focus and commitment to the final outcome of the design.Lastly, potential conflicts of interest can be properly managed or even completely avoided.

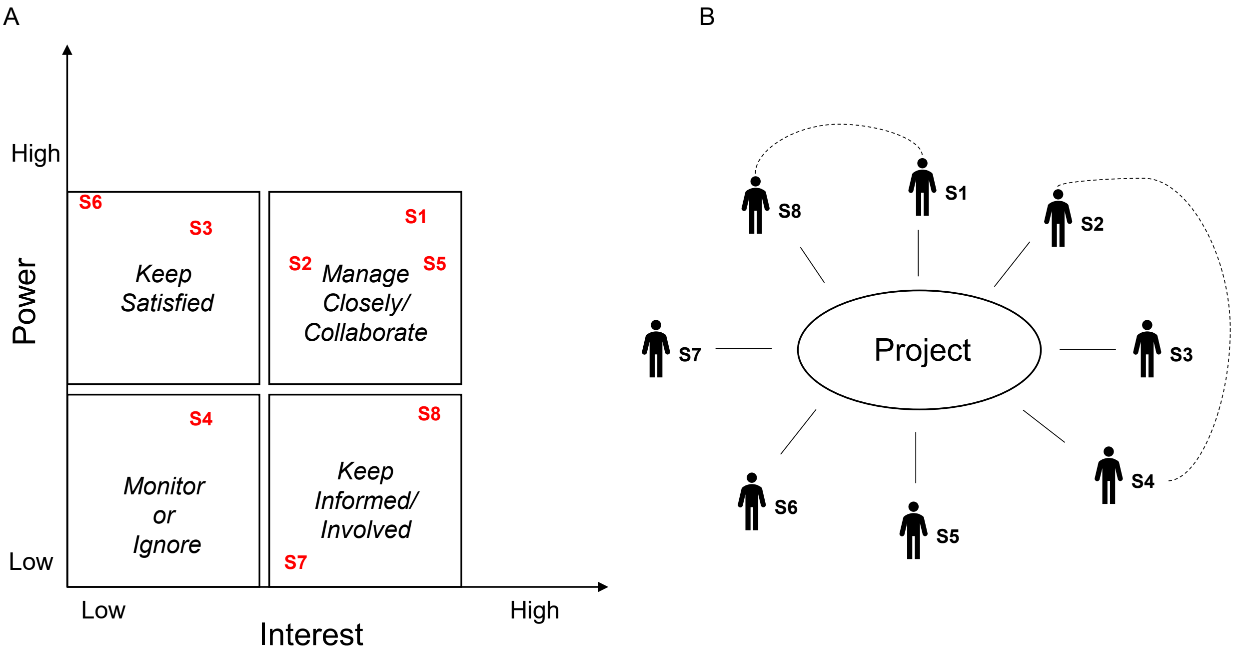
IV.5.1.1. Stakeholders - Identify

Usually the identification process starts with a brainstorming session within the developing organization. This brainstorming has as purpose to identify the most obvious stakeholders that influence- or have interest in the activity of the organization [BAO16]. The first step is to identify categories of stakeholders [Sch+13]. For example, for a medical robot, those could potentially be policy makers, governmental agencies, insurance companies, society, users, subjects etc. (see also Figure IV.14). As a second step, try to think of the networks and collaborative groups of the already identified stakeholders. For example, the subjects of a robotics device are stakeholders, but what about their families and close friends? What about support groups and local private businesses? In many cases these networks could be considered relevant stakeholders but their interest may vary. Lastly, talk to experts from inside and outside of the organization, in order to identify any remaining stakeholder.

IV.5.1.2. Stakeholders - Analyze

Commonly, stakeholders are categorized into direct primary, indirect primary, and secondary stakeholders [May05; BAO16]. Direct primary stakeholders are those that stand to benefit directly from the project under consideration, while indirect primary stakeholders are merely affected by it, without being direct beneficiaries. Secondary stakeholders are all remaining stakeholders. In order to define which of the identified stakeholders belong in these categories, we can perform a power-interest analysis, sometimes also referred to as importance-influence analysis. In this case “power” illustrates the ability of the stakeholder to influence policies or organizations related to the project, for example someone who can influence insurance policies in the case of medical robots. Interest shows how much a specific stakeholder is affected by the project or interested in its outcomes. This way, it allows you to prioritize the stakeholders in essentially four categories:

- High Power - High Interest: These are the primary stakeholders (S1, S2, and S5 in Figure IV.14A). They need to be fully engaged and kept satisfied with the project
- High Power - Low Interest: These are secondary or indirect primary stakeholders (S3, and S6 in Figure IV.14A). These people should be kept satisfied, and potentially defended against if a conflict of interest arises.
- Low Power - High Interest: These are secondary stakeholders (S7, and S8 in Figure IV.14A). They need to be kept informed to ensure that no major issues will arise. They can prove to be very helpful.



]

Figure IV.14.: *A. The figure shows a common tool to perform stakeholder identification and analysis (Power-Interest plot). The stakeholders can be categorized based on their interest in the project and the power of impact they have over it. B. This is a simple example of a context diagram. Here the stakeholders that are related to the project are plotted (rigid lines), as well as the connections between them if any (dotted lines).*

- Low Power - Low Interest: These are secondary stakeholders (S4 in Figure IV.14A). They need to be monitored, but with minimum effort and in some cases they can even be ignored.

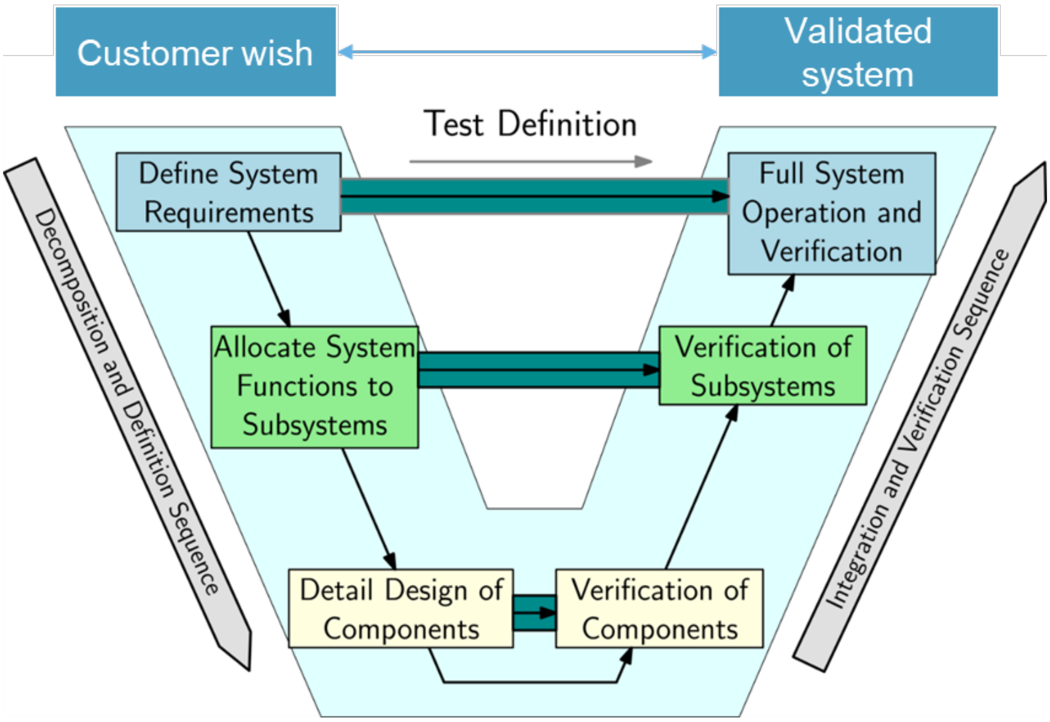
After prioritizing the stakeholders, all of the gathered information can be used to develop a strategy to manage stakeholders. Poor stakeholder management can easily cause a project to fail. A large portion of stakeholder management focuses on communication. A very simple and effective way to visualize the stakeholders and their communication around a project is given in Figure IV.14B, and it is called a context diagram [BVB16]. The context diagram shows the project in its context. The project is in the center and the stakeholders are placed around it with their names. More details can be added, such as what they require from the project, what they can add to the project, and their relationships with each other (for example in Figure IV.14B). In addition to the context diagram that can be used for communication and brainstorming, more elaborate documents can be drafted that include raw information useful to stakeholder management. An example of this can be seen in Figure IV.15, which is adapted from James Mayer [BVB16].

IV.5.2. The system view

Robotics is an increasingly complex field that integrates not only multiple technologies [Niz+21], but also multiple disciplines [Rei+17]. This increase in complexity indicates that we should start thinking of robots not as products, but as systems [Bon18]. According to Bonnema, there are considerable differences in design that are considered a product and those that are seen as systems. A product is usually shaped by its context (one-way interaction) and has only a few users/stakeholders. It is focused on optimization and is usually developed by a relatively small team with a few disciplines. On the other hand, a system can be viewed as something that is impacted by its context but also impacts it in return (two-way interaction), and often has multiple users and stakeholders. Its development is a matter of taking care of trade-offs and achieving good balance between different sub-systems and it is usually carried out by large multidisciplinary teams. This realization brings forth many challenges. A complex system is usually developed by decomposing it in smaller, manageable parts that are in many cases developed separately. However, despite the fact that the decomposition phase (Figure IV.16) is usually successful, the composition/integration phase is rarely taken into account and almost never from the beginning of the project. A logical consequence of decomposing the system into smaller components is the creation of interfaces between what are now sub-systems. The more of those are created the more interfaces need to be managed (Figure IV.17). So despite the fact that sometimes decomposing a complex system to many less complex components seems to reduce complexity, in fact it may do the opposite, if interfaces are not properly identified and managed. Also, the number of different disciplines needed to deal with the now complex systems is expanding, for example in medical robotics those can include, among others, from any kind of engineer to movement scientists, physiotherapists and doctors. This may create challenges in the communication between these disciplines and can easily result in incoherent and incompatible project outcomes. In section 5.3, some tools are briefly explained that can help designer-teams and systems engineers to deal with the aforementioned challenges (for detailed information see Systems Design and Engineering: Facilitating Multidisciplinary Development Projects by Maarten Bonnema [BVB16]).

Stakeholder	Key Interest	Power	Interest	Role in the project
Primary Direct				
S1	• Interest 1 • Interest 2	• High. This stakeholder can influence the project heavily	• High. This stakeholder has high stakes in the final outcome of the project	• Is responsible for tasks 1, and 2
Primary Indirect				
S3	• Interest 1	• High. This stakeholder can influence the project heavily	• Low. This stakeholder has low stakes in the final outcome of the project	• Will perform task 1
Secondary				
S4	• No interest	• Low. This stakeholder cannot influence the project heavily	• Low. This stakeholder has low stakes in the final outcome of the project	• No role

Figure IV.15.: *The table shows the categorization of stakeholders into primary and secondary. This table is an example of a tool that can be used for analysis of the stakeholders by writing down their interests, power, and role in the project. For brevity we did not write the table using all the example stakeholders from Figure IV.14 (S1-S8), but we picked three of them to represent all three categories (S1, S3, and S4).*



]

Figure IV.16.: *The figure shows an adaptation of the Vee model that is used to drive the development process of complex systems. The Vee model has a decomposition phase (left) and an integration phase (right). However, it also shows that those need to be thought of in parallel, and integration needs to be addressed early on. For example when system requirements are compiled, the process to test and validate those requirements has to be thought of already. During the decomposition phase the system is broken down into subsystems and components. However, it is important to keep in mind also the relationships between the decomposed subsystems and components, called interfaces (see Figure IV.17). The figure is adapted from [BVB16].*

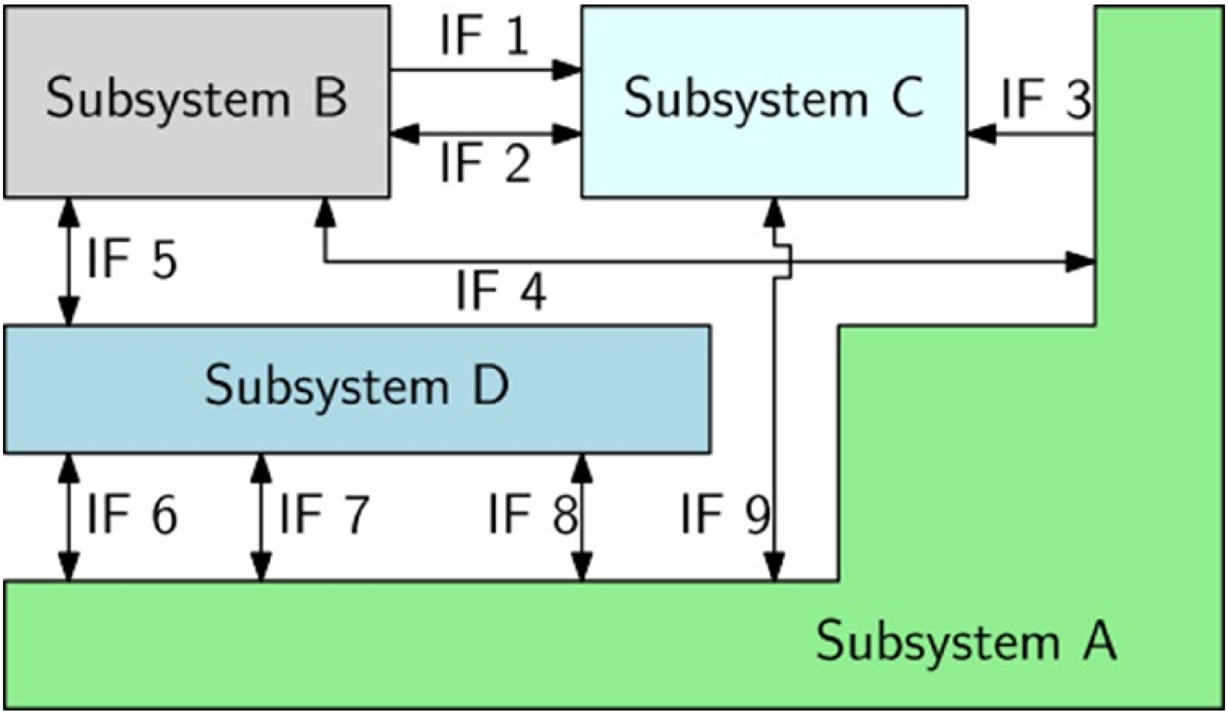
IV.5.3. System Design Tools

System design tools are practical tools designed by experienced systems engineers and support the implementation of the systems engineering process. A comprehensive guide on systems engineering and tools is the Guide to the Systems Engineering Body of Knowledge [SeB] (SEBoK). Such tools can be used to some extent by any designer, in order to facilitate proper design of complex systems. The above context diagram is one of those tools. In this section a few tools are discussed that address the challenges mentioned above (complexity, integration, multidisciplinarity, communication, etc.). The tools presented here can be found, among other sources, in the book Systems Design and Engineering: Facilitating Multidisciplinary Development Projects by Maarten Bonnema [BVB16]). Those tools are usually in practice complemented by systems thinking (even though they can be applied independently). This type of thinking will not be elaborated further in this book, but for the interested reader more can be found in the 12 thinking tracks developed by Bonnema et al [BVB16].

IV.5.3.1. Nine-Window Diagram

The nine-window diagram was originally derived from the theory of inventive problem solving (TRIZ)[Alt97]. This tool is used to place the system under design (SUD) into perspective with regards to its temporal and hierarchical context. An example of such a diagram can be found in Figure IV.18

This tool helps describe the past, the present, and the future situation at different hierarchical levels of the SUD. This way, one can depict and consider the implications of the SUD in different levels and how those relate to each other and promote discussion of the consequences within a multidisciplinary team. Another way of using the tool is to help apply scenario-based design (more about that in the sub-section called Scenarios). Several versions of the SUD can be developed for the middle (now) and right (future) columns. These can be discussed and evaluated based on criteria that come from the customer wish and the stakeholder management. The best version is selected and elaborated into a system



]

Figure IV.17.: *The figure illustrates the different interfaces that are created every time we decompose a complex system to less complex subsystems. Therefore, it is important to keep in mind the tradeoff between creating less complex subsystems that together compile our system of interest, however, this happens at the expense of creating more complexity due to the number of new interfaces that are created every time we break down our system. Therefore, decomposition may simplify design up to the point that interfaces are still manageable. The figure is adapted from [BVB16].*

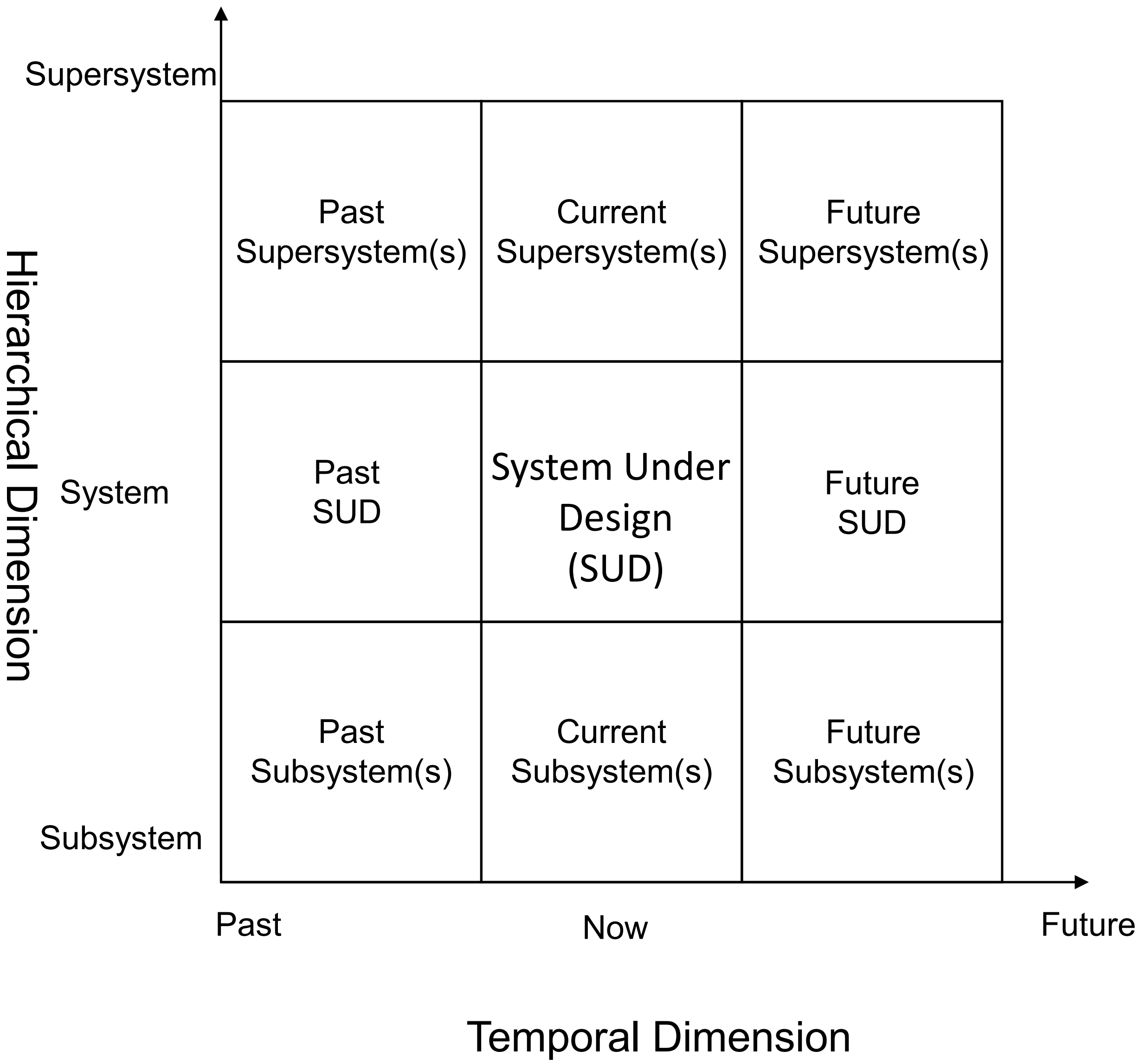
design. Therefore, it becomes clear that this tool is also very well suited for discussions with diverse stakeholders (users, subjects, medical specialists, engineers, people from marketing, etc.). This tool can also be found also in other variants (i.e. time/market).

IV.5.3.2. Scenarios

Storytelling and scenario making is commonly used in the process of designing a complex system. A scenario or story can cover a textual or pictorial description of the use of the SUD. For example:

"The physiotherapists enter the physiotherapy room 20 minutes before the patient arrives. They turn on the lower limb robotic exoskeleton and perform a quick diagnostic to ensure everything works properly. After checking the function of the robot, one of them programs the robot to the specific needs of the following patient, while the other prepares the necessary coffee and the mandatory paperwork. Once the patient enters they greet each other and have a discussion about the rehabilitation progress over coffee. Afterwards the patient is strapped to the robot and a safety trial is performed before the robotic exoskeleton is fully active. The physiotherapists calibrate the robot to the biometrics of the patient. The robot starts moving the legs of the patient slowly at first and faster as the patient seems to gain confidence, relying on AI algorithms. The same algorithms adapt the amount of support offered to the changing needs of the patient over time. The physiotherapists are observing the process, while one of them is holding the emergency stop button. "

This artificial story is an example of the use of a lower limb robotic exoskeleton such as the LOPES to perform lower extremity rehabilitation. The same story could have been made with or in combination with a series of pictures/-doodles /cartoons in order to communicate the same information. Such a story can help the designing team to discuss the SUD regardless of their disciplines and even manage to include the end users and other stakeholders. Discussions with the latter groups are possible because the use of jargon is kept to a minimum in such scenarios. This story can be transformed into diagrams (for example state transition diagrams or functional block diagrams) that can help the



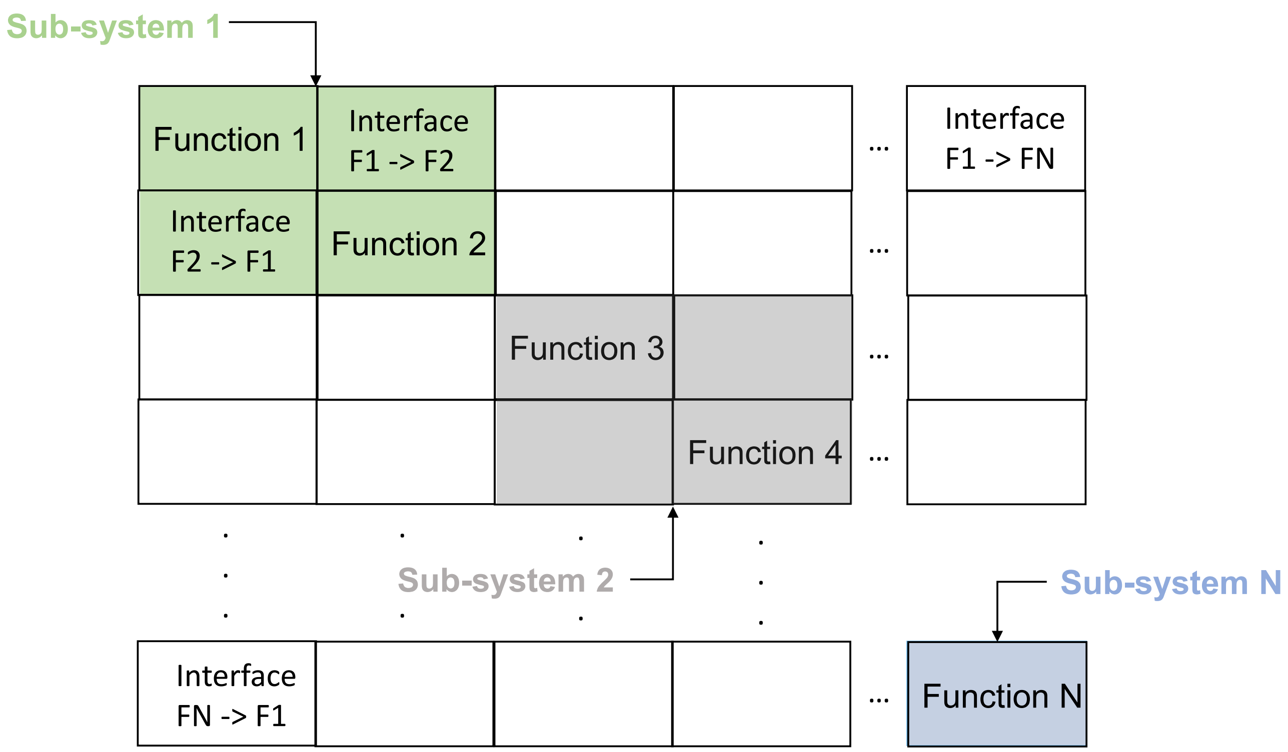
]

Figure IV.18.: *The figure shows the structure/form of a time-hierarchy nine-window diagram. Each box encompasses different combinations of systematic hierarchy and temporal specifications for the system under design.*

engineers understand the design. Additionally, such a story can help with the creation of the initial requirements for the design and help identify potential use bottlenecks and additional challenges that can only be identified during such an example of operational thinking.

IV.5.3.3. N^2 Diagram

Before talking about the N2 diagram, we need to briefly explain what interfaces are. In many cases and in practice, the total system is divided in its main subsystems. A subsystem is defined as a smaller system that is largely independent from the other subsystems, and it performs a set of functions. An interface is a point of interaction between functions of the system or sub-systems [BVB16]. Additionally, when we refer to functions of the system we mean the main actions performed by that and their subcategories. For example the main function of a surgical robot is to assist in surgery. From there, we can split this main function into sub-functions or try to define more high level functions. A few examples can be: make incision, insert clamp, remove tissue, make suture, etc.. The N2 diagram [Nas] is a useful tool that helps with proper interface management. In practice it can be used to:



]

Figure IV.19.: *This is an example of an N2 diagram, which is a square matrix, with the functions of the system placed in the diagonal (F1-FN) (functional N2 diagram). Those functions can be grouped into subsystems or modules (modular N2 diagram). The rest of the boxes show the interfaces between the different functions or sub-systems (read clockwise). The outputs of a function listed on the row of that function and the inputs to a function listed in the column of that function. Where a blank appears, there is no interface between the functions.*

- Inventory all interfaces between functions
- compare different designs regarding the number and type of interfaces
- Monitor the interfaces when the system is developed further

It is a square matrix (see Figure IV.19), with the functions of the system placed in the diagonal, the outputs of a function listed on the row of that function and the inputs to a function listed in the column of that function. Where a blank appears, there is no interface between the functions. In addition to defining the interfaces, the N2 diagram also pinpoints areas where conflicts could arise in interfaces, and highlights input- and output dependency assumptions and requirements [Nas]. The interfaces refer to types of Information, energy , materials, and persons that are transferred between the subsystems/functions.

Once we plot the functions and their interfaces in the N2 diagram, the functions can be grouped and allocated to subsystems. Then another N2 diagram is made: the modular N2 diagram (see shadowed areas in Figure IV.19). Here, the subsystems (shaded areas) are on the diagonal, and the interfaces between the subsystems are represented off diagonal. Then, for each subsystem the interfaces inside this subsystem are worked out in a separate N2 diagram.

The tools presented in this chapter are powerful systems engineering tools, used commonly in the high-tech industry to help with the development of complex systems. They can be applied in different phases of the system design process, but of course their use is dependent on the context and the specific requirements of each system. Their main aim is to help the designer to be able to keep an overview of the project at all times, and more importantly, align that overview to the wishes, needs, and interests of the various stakeholders, while keep a clear communication between the different disciplines involved. What is important to remember, is that the best way to understand and master these tools is to apply them in practice and in real projects.

IV.6. Integration in the workspace

For the implementation of robotic technologies in the medical domain, it is important to be aware of all the possible opportunities and threats. As the implementation often is a precious trajectory, both in terms of costs for the device itself, but also in terms of personal investment in time, it is important to make a good plan. In this chapter we will give a short overview about the topics one has to be aware of. We also give an example of a successful implementation.

IV.6.1. Starting a project

When you start with the orientation on the introduction of a new robotic device in the clinical field, it is first important to ask yourself a few questions. First of all, you have to wonder which problem you would like to solve. And subsequently, whether the device will be able to solve the problem and if it is worth starting any further investigation. If the answers are 'yes', you have to collect some people around you who are motivated to start working with the device. All these people should support the device's introduction. If you will have a lack of support, there is also a high probability of failure.

Stakeholders involved in the implementation of robotic technologies (in random order of importance)

- Patient: it is important to know the wishes of the end-user of the device. Therefore, it is important to involve them from the start of the project.
- Doctor: as the doctor is ultimately responsible for the medical treatment and should therefore support the new device.
- Physical/occupation therapist: these are the ones with the most contact with the patient and are most often involved in the therapy.
- Manager of the clinic: to be able to do the investment for the new device, the manager of the department should give his approval too.
- Researcher: to determine the added value it is important to define some research questions before you start with the use of the device. The research department can help.
- Innovation manager: to bring new devices to the clinic, you often need someone who is aware of all the options and who has contact with all the companies.
- Insurance company: for the reimbursement of the device it is important to talk with the insurance company and to see if you can arrange any type of reimbursement.
- Technical department: for the use in the clinic every apparatus has to be checked by the technical department on a regular basis.

IV.6.2. Tips

- Check if the device has the right approvals (CE-mark for use in Europe) for both clinical- and home use
- Start in time with the whole trajectory and schedule enough time for the introduction
- Check if the users (physical therapist and patients) need some education and certification before they can use the device in clinical practice.
- Make a clear protocol with all the necessary information (who schedules the therapy sessions, who is allowed to use the technology?).
- Ensure the availability of an easily accessible project room.
- If the device does not meet your expectations, stop the project.
- Enjoy the project!

IV.6.3. Example of a successful implementation

The Sint Maartenskliniek has a long history of specialized gait training for patients with spinal cord injury. This gait training was mainly focused on incomplete spinal cord injury patients. However, in order to help people with a complete spinal cord injury (that do not have the option to regain gait capacity), we were looking for a device that could help them to compensate for the loss of function of their legs. One of the prerequisites was that they could use the device in their home environment, as this is the place where they will need standing and walking. Also one of their questions was if it

would be possible to regain standing and walking.
When exoskeletons came on the market with the right certification, we wanted to explore the options for the use in our clinic and in the home environment of the patient. Questions from a clinical point of view were: 'is it safe to use an exoskeleton at home', 'what kind of training program is needed', 'what are the effects on secondary health complications like bone density, spasticity and pain'? Our technical department was also involved and they wondered how reliable the technique and electrical parts would be. Together with the innovation manager, general manager, research department, physical therapist, doctor and patient we started a project. After a pilot with a few patients we decided to buy an exoskeleton, because of the enthusiastic reactions of both the pilot patients and physical therapists. At the same time we wrote a research proposal for a PhD student. We received a grant and could start with a study. We looked both at the training period and at home use. Through the several studies we did, we were able to collect a lot of valuable data for clinical practice. We know what training program to prescribe and which patient characteristics help in gaining the exoskeleton skills. We are also aware of the risks and wrote a safety protocol. At the moment we offer patients with a complete spinal cord injury a training program so that they are able to use an exoskeleton in the home environment. The only thing we were not able to solve is the reimbursement. Despite a few meetings with the insurance company, the exoskeleton is still not reimbursed. Since the exoskeletons did become commercially available, we did feel it was important to gain a better understanding of the application, despite the fact that they are not covered by insurance. If research will show a reduction of secondary health complications and thus a reduction in related costs, this may lead to a coverage of the insurance companies.

IV.7. Basic terminology

Actuator
Something that delivers a torque or force to drive a mechanical device into motion.

Ball-screw

Actuator mechanism to transform rotational motion into linear motion using a screw thread.

Direct-drive

Actuator mechanism in which motor torque is directly transmitted to the output device, without gears.

Series-elastic

Actuator mechanism in which a spring is placed in series between the motor and an output device.

Accelerometer
A sensor that measures linear acceleration.

Assist-as-needed
Principle in human-robot interaction in which a robot coupled to a human only assists the human in a task when "needed" as determined by a set of conditions, and is compliant otherwise.

Bandwidth
The range of frequencies over which a device can operate, or the amount of information that can travel over a communication channel within a certain timespan.

Backdrivable
A mechanism is considered backdrivable if motion at the output side can result in motion at the input side.

Backlash
A mechanism has backlash if a loss of motion can occur between input and output sides due to gaps in the mechanism, such as two gears not tightly connecting to each other.

Co-bot
A cooperative robot.

Compliance
A mechanism or controller is compliant if it gives way to external force or torque.

Controller
Regulator that observes a certain (sensory) input, and produces a certain (actuator) output.

Admittance
Controller architecture in which a motion is regulated to admit to a given input force.

Impedance
Controller architecture in which a force is regulated to impede a given input motion.

Position
Controller architecture in which a position is the controller output.

Torque
Controller architecture in which a torque is the controller output.

High-level
Often refers to control on a system level, involving multiple actuators and related sensors.

Low-level
Often refers to control on an actuator level, involving a single actuator and potentially its related sensors.

Control Interface
Commonly used term to refer to the modality responsible for the acquisition of user intention, its processing, and the creation of the final control signal that facilitates communication between a user and a robot. Electromyography (EMG) is a common control interface used in the control of robotic exoskeletons

Damping
The relation between velocity and force.

Degrees of freedom
The number of independent parameters required to describe the state of a system.

Disturbance-observer
Software paradigm that attempts to observe a (known) disturbance source, such that disturbance information can be used by a controller to reject it.

Dynamics
The field of mechanics concerned with forces and their resulting motions.

Forward dynamics
The use of forces on a system to compute the resulting body motions.

Inverse dynamics
The use of body motion of a system to compute the forces that cause them.

Encoder
Sensor used to encode a signal, such as the amount of rotation in a joint.

End-effector
A device attached to the endpoint of a robot body, such as a gripper at the end of a robot arm.

Energy-storage-and-return
The principle of storing energy in a system, and returning it at a later point in time.

Exoskeleton
A device, either passive or actuated, that is worn in parallel to the human body.

Gear-ratio
The rotation ratio between a motor shaft and an output shaft, caused by the drivetrain (e.g. gears) in between. A gear ratio of 1:2 implies that the motor shaft has to spin once to make the output shaft spin twice.

Gimbal lock
The loss of one degree of freedom when two rotational axes align.

Gyroscope
A sensor that measures angular velocity.

Haptics
Field concerned with the sense of touch as part of an (mechanical) interface.

Human-in-the-loop Control paradigm in which a human is actively influencing the signals in the control loop of a controller, often through sensors that measure movement of the human body.

Inertia
The relation between (angular) acceleration and (torque) force.

Inertial measurement unit (IMU)
A sensor that often consists of an accelerometer, a gyroscope, and a magnetometer, and fuses their outputs to perform estimates of linear acceleration, angular velocity, and orientation.

Joint
A point in a system where two bodies are linked together through zero or more degrees of freedom.

Ball-socket

A joint that allows three-degree-of-freedom rotation, consisting of a cup-like
socket that holds a ball-like sphere.

Prismatic

A joint that allows one-degree-of-freedom pure translation.

Revolute

A joint that allows one-degree-of-freedom pure rotation.

Kinematics
The field of mechanics concerned with motions of bodies, without considering the forces that cause them.

Forward kinematics

The use of a system's independent coordinates position and motion information to compute global position and motion, such as using joint angles to compute an end-effector position.

Inverse kinematics
The use of global position and motion information to compute position and motion in the system's independent coordinates, such as using an end-effector position to compute the joint angles required to obtain that position.

Kinematic chain
A set of kinematic bodies linked together through joints.

Magnetometer
A sensor that measures its orientation with respect to the earth's magnetic field. *Manipulator*
(Part of) a robot that manipulates (objects in) the environment.

Mechatronics
Field concerned with the integration of mechanical, electrical, and software engineering systems.

Range of motion
The amount of movement allowed by a joint.

Robot

Parallel robot

Serial robot

Stakeholder
Person or organization that has an interest in any decision or activity of an organization.

System

Stiffness
The relation between position (deviation) and force.

Strain (mechanics)
Displacement between particles in a body relative to a reference length

Stress (mechanics)
Forces that particles within a continuous material exert on each other.

Torque
Moment of force

Workspace
All positions that a robot or its end-effector can reach.