Fast and Strong Lightweight Robots based on Variable Gear Ratio Actuators and Control Algorithms Leveraging the Natural Dynamics

by

Alexandre Girard

B.Ing., Université de Sherbrooke (2010)
M.Sc.A., Université de Sherbrooke (2013)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2017

Author .....
Department of Mechanical Engineering
March 31th, 2017

Certified by .....
H. Harry Asada
Ford Professor of Engineering
Thesis Supervisor

Accepted by .....
Rohan Abeyaratne
Chairman, Department Committee on Graduate Theses

Fast and Strong Lightweight Robots based on Variable Gear Ratio Actuators and Control Algorithms Leveraging the Natural Dynamics

by

Alexandre Girard

Submitted to the Department of Mechanical Engineering
on March 31th, 2017, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

In many applications, robots have to bear large loads while moving slowly and also have to move quickly through the air with almost no load. These type of bimodal tasks, with conflicting requirements in terms of operating speeds and desired impedances, often lead to the use of oversized and inefficient actuators which are inhibitory particularly for mobile robots. Multiple gear ratios, like in a powertrain, address this issue by allowing an effective use of power over a wide range of output speeds, by enabling significant changes to the reflected intrinsic actuator impedances and by making possible the leveraging or attenuation of the natural load dynamics. This thesis aims to develop the technological solutions needed to use variable gear ratio actuators and exploit the advantages of variable transmissions in a robotic context. First, by addressing the issue of how to make fast and seamless gearshifts between two very different reduction ratios under diverse load conditions, with a solution based on a dual-motor actuator architecture and a control scheme using the null space. Second, by developing control algorithms that select optimal gear ratios dynamically based on state feedback, to move with minimal motor torques and to adjust the output impedance appropriately given a task. The proposed approach exploit variable transmissions not merely for increasing maximum torque and speed, but also to significantly alter the dynamic properties, including load sensitivity, robustness, and backdrivability. Simulations and experiments using a novel lightweight robotic arm using three custom-built dual-speed dual-motor actuators are presented. Results demonstrate very fast gear shifting in highly dynamic situations with dual-speed dual-motor actuators, and show that actively changing gear ratios using the proposed control algorithms can lead to an order-of-magnitude reduction of necessary motor torque and power.

Thesis Supervisor: H. Harry Asada
Title: Ford Professor of Engineering

Acknowledgments

I would first like to thank my thesis advisor, Professor Asada, who gave me the opportunity to work on unique robotic research projects in the d'Arbeloff laboratory. His guidance was invaluable, always thinking ahead to guide my efforts in a direction that would generate innovative and relevant research. I hope that I have learned some of his wisdom during my years at MIT. Also, special thanks to Professor Hogan and Professor Slotine for been members of my thesis committee: Their advices and feedback contributed to strengthening my thesis. Many thanks to all my labmates at the d'Arbeloff laboratory. The lab was a very dynamic and exciting place to work. Special mention to Lluis and Kosuke for spending precious time helping me reviewing technical drawings of gearbox prototypes. I would also like to thank the whole MIT community, for being a constant source of inspiration. The education I have received during my PhD studies was of outstanding quality, and I learned a lot spending time with members of the MechE and the CSAIL community.

I would also like to thank The Boeing Company, Sumitomo Heavy Industry, *Le Fond Québécois de Recherche sur la Nature et Technologie* (FQRNT) and The Natural Sciences and Engineering Research Council of Canada (NSERC), who contributed to funding my PhD studies at MIT.

I must also acknowledge the strong support I have received from my friends and family. My wife Catherine has given outstanding support, and I could not have accomplished everything I did without her. I must also thank all my friends who came to visit me in Boston; their visits provided me with breaks from the frenetic lifestyle of MIT studies. I would also like to thanks my parents for always encouraging me in my studies, and Jean-Sébastien for giving me the push I needed to pursue my ambitions. Without them I would never have been able to pursue my dream at MIT.

Contents

List of Figures

List of Tables

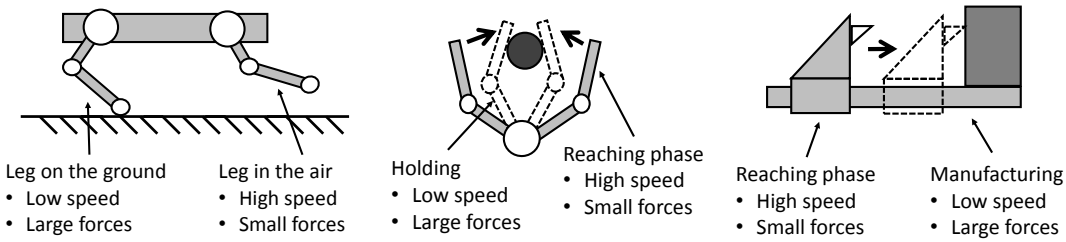
Chapter 1

Introduction

"To have a great idea, have a lot of them."

– Thomas A. Edison

This thesis proposes an actuation technology and control schemes addressing the fundamental problem of efficient power transmission in diverse situations. In many robotic systems, actuators are often required to operate in distinctively different torque-speed load conditions. Machine tools, for instance, are usually either moving at high speed unloaded during reaching phases, or moving slowly applying large forces during manufacturing operations (see Fig. 1-1c). Also a legged robot, for example, has to move its leg forward quickly through the air and, once touching the ground, it has to bear a large load (see Fig. 1-1a), or a gripper needs to reach the part quickly and then has to apply large holding forces (see Fig. 1-1b).



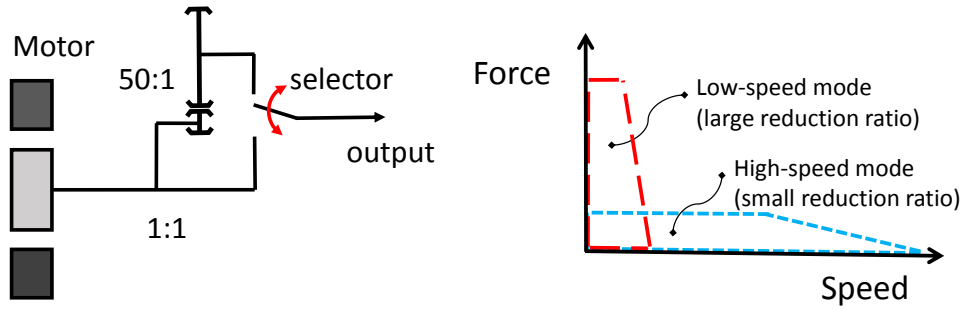
(a) Legged robot (b) Gripper (c) Machine tool

Figure 1-1: Robotics system encountering very different load situations

These two operating conditions, high speed at low torque vs. high torque at low speed, are often an order of magnitude different, while the required output power is similarly low. This discrepancy in requirements is problematic as most actuators will be operating far for their optimal conditions with a gear ratio picked from a middle ground. Electromagnetic actuators are not optimal in term of efficiency and power output at extremum torque-speed conditions. This often leads to the use of over-sized and inefficient actuators, when designing for such bimodal operations, which is inhibitory particularly for mobile robots.

1.1 Proposed approach: variable transmissions

To meet the power requirement of all operating points with small actuators, it is proposed to use electric motors coupled to a gearbox where the reduction ratio can be drastically changed online, see Fig. 1-2. Using such Variable Gear-ratio Actuator (VGA) on the many joint of robotic systems, leads to offering a wide range of properties in terms of speed, force and impedance, see Fig. 1-3.



(a) two drastically different gear ratios (b) discrete operating modes

Figure 1-2: Variable gear ratio actuator with two discrete options

The two main advantage of the VGA approach are: **good power output and efficiency for a wide range of output speeds** and **radical changes of intrinsic impedance** (goes with the square of the reduction ratio).

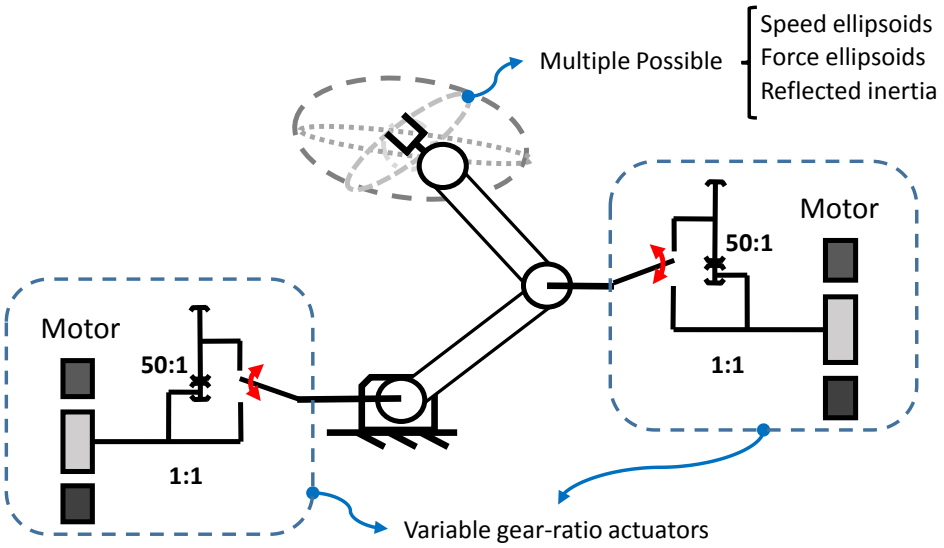


Figure 1-3: Robotic arm equipped with variable gear ratio actuators

1.1.1 Features of gear shifting in a robotic context

Power output and efficiency over a wide range of speed In many situations, using multiple gear ratios allows for the downsizing of the motors while still meeting required forces/speeds capabilities. A small lightweight actuator can generate large torques and move at high speed if equipped with both a large and a small gear ratio. Furthermore, by actively selecting gearing ratios to keep motors in efficient regimes, the energy consumption of a robot can be greatly reduced.

Radical changes of reflected impedance The transmission has a radical effect on the output impedance of a robot; the motor inertia and viscous damping are reflected to the output proportionally with the square of the reduction ratio. Gear shifting can thus also be used as an alternative approach to variable impedance actuation. A robot joint could be made backdrivable by selecting a small reduction ratio, to interact safely with the environment. Alternatively, a joint could be made non-backdrivable by selecting a very large reduction ratio, to resist easily external disturbances.

Exploitation or attenuation of the external load dynamics Changing the gear ratio has a radical effect on the natural dynamics of a system. By changing dynamically the gear ratio, it is possible to select the natural dynamic behavior that is most advantageous for a task. For instance, selecting a small reduction ratio to exploit gravitational forces pulling the robot in a desired direction, or on the other hand, selecting a large reduction ratio to hold a heavy weight with small actuator torques.

Directionality of properties For a multiple degrees-of-freedom (DoF) mechanism, the speed/force properties are directional. Typically the Jacobian (transformation from motor coordinates to task-space coordinates) of a mechanism is only a function of the configuration $J = J(\boldsymbol{q})$. For a mechanism using m variable gear ratio actuators with l possible gear ratios, the Jacobian would also be function of the selected gear ratios R and thus dependent on control inputs $J = J(\boldsymbol{q}, R)$. Hence, for a given configuration there is m^l manipulability ellipsoid options. Fig. 1-4 illustrates situations where gear ratios would be picked to meet the task requirements in terms of load bearing, speed and impedance.

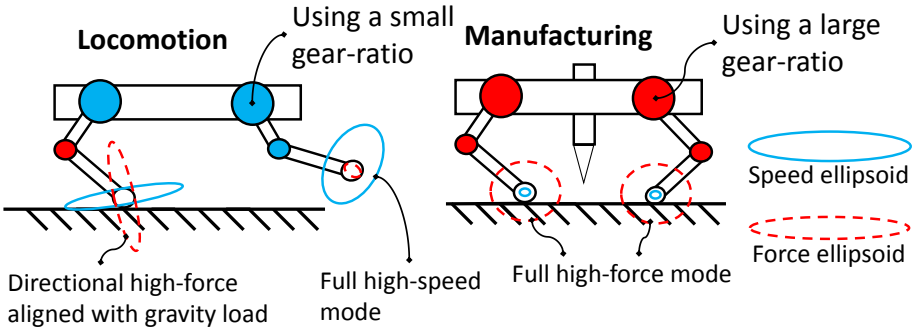


Figure 1-4: Examples of advantageous gear selections with a multi-DOF robot

1.1.2 Differences from vehicle powertrain transmissions

Variable transmissions have been mostly explored in the context of vehicle powertrains. Their usage in robotics raise different issues but also new opportunities, and those distinctions are briefly presented in this section.

Range of gear ratios For car transmissions, a typical gear ratio range start with a 3:1 reduction for the first gear and the last gear ratio is an overdrive of about 0.8:1. For robotic applications, the explored idea in this thesis is a much wider range between gear ratios (one order-of-magnitude and more). For instance, this thesis presents an actuator prototype having a first gear with a 474:1 reduction and a second (and last) gear with a 23:1 reduction.

Electromagnetic transducer vs. internal combustion engine characteristics Internal combustion engines have an efficient power-output on a narrow range of speed, hence transmissions require many gear ratio options (modern cars use up to eleven [17]) to keep the engine at its optimal velocity for any vehicle speed. Moreover, internal combustion engines cannot produce torque at low speed and car transmissions must be equipped with a disengaging clutch. Electric motors are much more flexible and there is less constraints for the design of adapted variable transmissions.

Car dynamics vs. robot dynamics For car powertrains the driven load is always a large inertia and is thus naturally attenuating any discontinuity in torque during a gear shift. However, this is not the case in a general robotic context.

Mutli-DoF systems Last and most fundamental difference, this thesis explores the control of VGA in multi-DoF systems like robotic arms, while for vehicle powertrains the system is always single-axis.

Solutions concepts All in all, as exemplified at eq. (1.2), this thesis explores variable transmissions in multi-axis systems where gear ratios exhibit drastic variations. This differs from typical car transmissions with small jumps between successive gear ratios for a single-axis system, see eq. (1.1).

$$\text{Car transmission:} \quad R \in \{\, 3\,,\ 2\,,\ 1.4\,,\ 1\,,\ 0.8\,\} \tag{1.1}$$

$$\text{Proposed robots:} \quad R \in \left\{ \begin{bmatrix} 474 & 0 \\ 0 & 474 \end{bmatrix}, \begin{bmatrix} 474 & 0 \\ 0 & 23 \end{bmatrix}, \begin{bmatrix} 23 & 0 \\ 0 & 474 \end{bmatrix}, \begin{bmatrix} 23 & 0 \\ 0 & 23 \end{bmatrix} \right\} \tag{1.2}$$

1.2 Main challenges

How to make fast and seamless gearshifts? Gear shifting is more technically challenging in robotics applications than in vehicle applications. For powertrains, the load is mostly a large inertia, while for robots, the loads may exhibit a rich range of dynamics including spring-like and damper-like loads. Hence, unlike vehicle applications, leaving the load free momentarily during transitions (from one gear ratio to another) is not acceptable in the context of robotics. Moreover, many robotic applications would benefit from having order-of-magnitudes difference between the possible gear ratios, i.e. a wider range of ratio than what is typical in vehicle powertrains. Hence, an effective gear shifting methodology adapted to robotics is needed, allowing for fast and seamless transitions between very different gear ratios under diverse load conditions.

When to use what gear ratio? From the control perspective, automating the gear ratios selection in a robotic context is a new and challenging problem. Gear shifting is a very non-linear process and the plant becomes a hybrid dynamical system if the usable gear ratios are a set of discrete values. Hence, no classical control approach can be applied directly to handle the additional gear ratio selection control input. In simple scenarios, the gear ratio selection can be based on simple principles. However, to handle the generalized problem of the gear ratios selection for multi-DoF robots, that experience diverse types of forces acting simultaneously and coupling between each axis, new methodologies are needed to generate trajectories and feedback laws that would use effectively all the gear ratios options and exploit their advantages.

1.3 Original contributions

1.3.1 A gear shifting methodology adapted to robotics

The first major contribution of the thesis is an actuation technology capable of fast and seamless gearshifts between two discrete order-of-magnitude different gear ratios. This technology consist of a mechanical architecture, that will be refer as DSDM (dual-speed dual-motor), used in conjunction with novel gear shifting control algorithms. The key idea is exploiting the internal degree-of-freedom (DoF) of the actuator to make possible transiting for one gear ratio to another while also always fully controlling the output load.

1.3.2 Control algorithms to select gear ratios dynamically

The second major contribution of this thesis, is the development of intelligent automatic gear ratio selection schemes for robotic systems. The key idea is using a model to estimate intrinsic and extrinsic forces, to compute if it is more advantageous to attenuate extrinsic forces with large gear ratios or alternatively to leverage them with a small gear ratios. The method can be applied to arbitrary n-DoF fully-actuated non-linear robotic systems using variable transmissions. Also, to the knowledge of the author, the work in this thesis is the first exploration of closed-loop selection of gear ratios for multi-DoF robotic systems.

1.3.3 A robotic arm using variable gear ratio actuators

This thesis also presents a novel 3-DoF robotic arm using a variable gear ratio actuator at each joint, the first of its kind to the best knowledge of the author. This very lightweight robotic arm can move at high speeds, apply large forces and exhibit a wide range of impedance.

1.4 Results

This thesis focused on demonstrating the viability and exploiting the advantages of robotic systems using variable transmissions, with a vertical exploration of related topics: actuator design, actuator controllers, robot controllers and motion planning algorithms.

Prototypes Two generations of DSDM actuator prototypes were designed, manufactured and tested. The first generation DSDM prototype consists of a linear actuator using a ball-screw output. The second generation consists of two actuated revolute joints equipped with the DSDM technology. A custom lightweight 3-DoF robotic arm using the DSDM actuator prototypes was also built.

Software A *Python* library was developed for providing tools for planning trajectories, controlling and simulating the behavior of robots using VGA. This library also includes all the controllers proposed in this thesis. Additionally, ROS wrappers to use those algorithms online to control prototypes were also developed. Both library are open-source and available on *Github* at github.com/alx87grd/alexrobotics and github.com/alx87grd/dsdm_robotics_ros.

Analytical Control laws exploiting the null space of DSDM actuators, leading to independent control of the internal DoF are synthesized. A modeling approach and representation for multi-DoF system with variable transmissions is proposed, with a special emphasis on intrinsic vs. extrinsic dynamics. A closed-form solution of optimal gear ratios for a class of n-DoF robotic systems on a known trajectory is derived. Model-based control laws and optimal gear ratios selection algorithms are proposed, and guarantees in terms of stability and chattering behavior are derived.

Simulations The advantage of dynamic gear-selection is demonstrated using simulations of multi-DoF robotic arms. Comparisons to equivalent robotic systems using fixed gear ratio actuators to accomplish the same motion show drastic improvement

in terms of maximum necessary motor torque and integral cost metric.

Experiments Multiple experiments with actuator prototypes demonstrate the salient features of VGA actuators and the ability of the DSDM technology to change gear ratio quickly and seamlessly even in very dynamic situations. Experiments usign 1-DoF and 2-DoF of the robotic arm prototype reproduce the advantageous behaviors obtained in simulations and demonstrate the viability of the technology and control schemes in real-world conditions.

1.5 Organization of the thesis

First, chapter 2 discusses manufacturing applications that would benefit from the developed technologies in this thesis. Chapter 3 then presents a variable gear ratio actuator technology, referred to as DSDM, using a gear-shifting methodology adapted to robotics. Chapter 4 explores the generalized problem of dynamic selection of gear ratios for a robotic systems equipped with variable transmissions, and present control algorithms. Chapter 5 presents a novel robotic arm using three custom built DSDM actuators, used in all experiments presented in this thesis, and discusses the mechanical design and control system implementation.

Chapter 2

Aircraft Manufacturing Automation: Concepts and Challenges

"We need men who can dream of things that never were and ask why not?"
– John F. Kennedy

This chapter presents robotic concepts to automate the manufacturing of airplanes, and discuss challenging actuator requirements motivating the work of this thesis. Automating aircraft production requires robots going inside the fuselage, unlike automobile production where the car can be on an assembly line and surrounded by robots. The focus is on the production of commercial aircraft, but issues discussed are relevant in the context of manufacturing any large objects (ships, buildings, etc.).

2.1 Current situation

Currently aircraft manufacturing is dependent on highly qualified manual workers because of the complexity of tasks involved and the difficulty for accessing manufacturing sites. Currently, many temporary assemblies such as scaffolds are used to assist human workers accessing the manufacturing sites. Hence, traditional robotics systems hardly fit this type of environment; industrial robot arms are too heavy and bulky to be used effectively inside the fuselage [44] [42].

2.2 Solution concepts

Many concepts have been proposed to address this complex problem. Here, three class of solutions are described: long snake-like articulated arms 2.2.1, wearable robots 2.2.2 and mobile robots 2.2.3.

2.2.1 Lightweight long manipulator arms

The first solution is a direct extension of the approach used in the automotive industry: using robotic arms to reach inside the part. However, because of the size of the fuselage and the highly constraint environment, robot arms needs to be very long and highly articulated to be able to reach manufacturing sites. This type of robot arm is usually refer to as snake-like robots [8]. Fig. 2-1 illustrates a concept of a long serial robotic arm reaching inside the fuselage through a window hole.

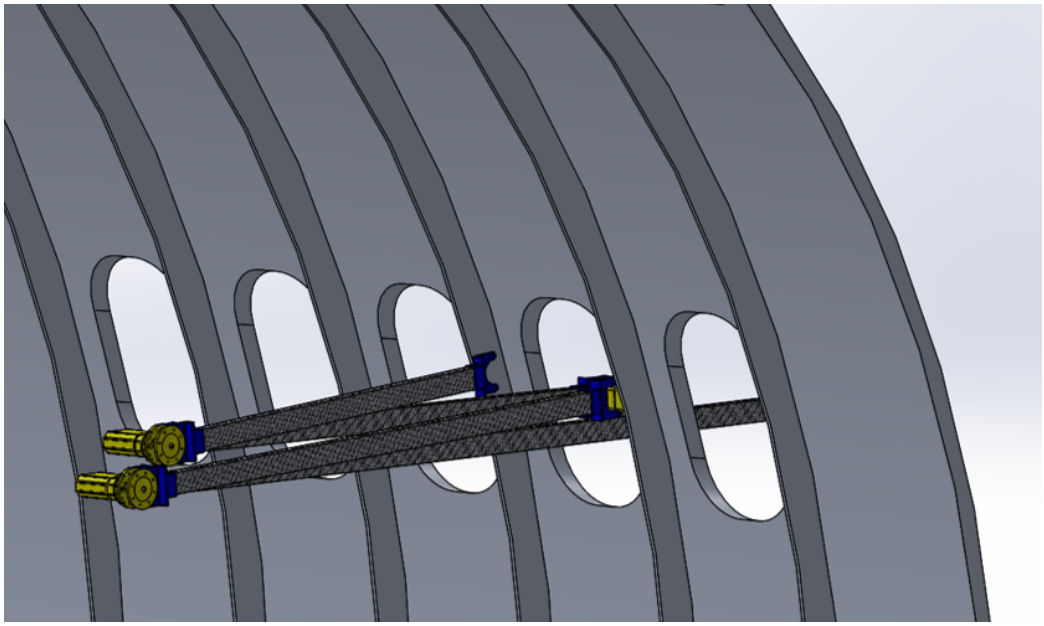


Figure 2-1: Long lightweight arm concept for interior access

One of the main bottleneck of such concept is the weight and volume of the actuation system [53]. With a highly-articulated serial arm, many actuator are required and each of them needs to bear the weight of the payload and posterior links. Moreover, with a highly articulated arm, it is hard to design transmission mechanisms to

displace actuators from the joint toward the base of the robot where their mass would be a lesser issue. Hence, typical snake-like arms using electric motor usually struggle just to overcome their own weight and have very limited payload capabilities.

2.2.2 Wearable robots

Another possible solution to bring robots on site easily is to use the help of humans, which unlike robot would have no problems navigating and moving inside a manufacturing site. The idea is to augment human capabilities with a wearable robotic system. One approach is using exoskeleton to improve the strength and precision of worker. An alternative, illustrated at Fig. 2-2, is supernumerary robotic limbs that can be used to brace workers, assist them in complex tasks and others.

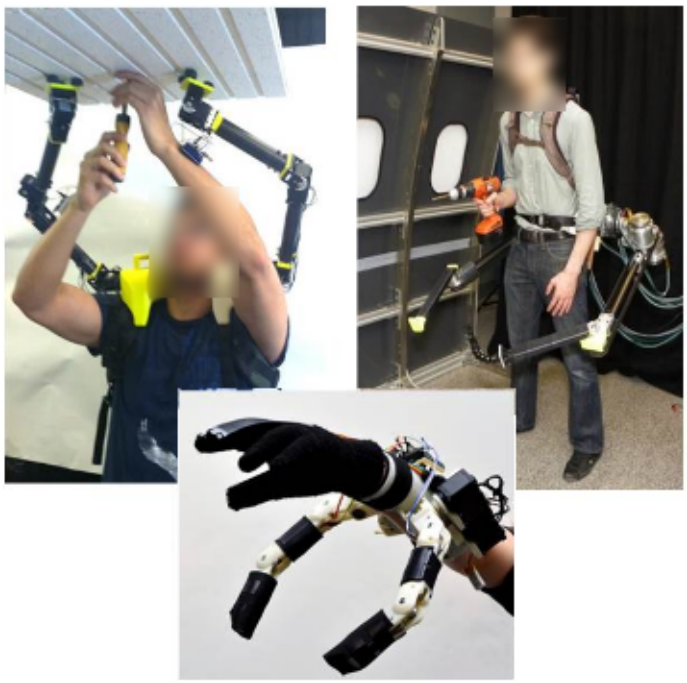


Figure 2-2: Wearable robot concepts and prototypes [6] [45] [62]

Because this type of robots is carried by a human, weight is also a critical characteristic. It is still a challenge to design wearable robots sufficiently light to be an asset and not a burden to the human wearer.

2.2.3 Mobile climbing robots

Another approach, aiming at a higher level of automation, is to have mobile robots walking or climbing inside the aircraft fuselage to reach manufacturing sites automatically. Fig. 2-3 illustrates a spider-like mobile robot. The idea for this concept is using local bracing for reaching the force and stiffness required for some manufacturing tasks, and using the same legs for site-to-site locomotion inside the fuselage. Fig. 2-4 illustrates a climbing robot prototype that was built to demonstrate the idea.

Figure 2-3: Mobile climbing manufacturing robot concept

This type of robotic system would face many challenges regarding reaching the required autonomy level. However one fundamental issue is still the weight, since the robot would have to fully bear its own weight as it climbs around the fuselage. Moreover, meeting the requirement of both locomotion and manufacturing tasks with the same actuators, with mass and volume constraints, is also a big challenge.

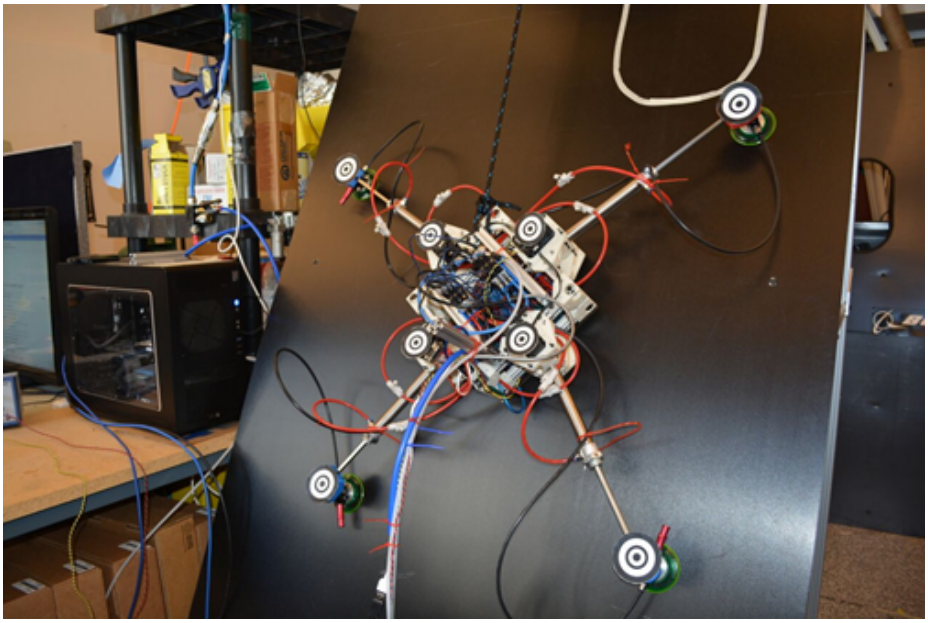


Figure 2-4: Climbing robot prototype

2.3 Technical challenges

All those concepts share the same difficulty regarding very challenging actuator requirements. It all comes from the fact, that for any system that needs to reach manufacturing sites inside a fuselage, volume and weight are highly constrained. Moreover, tasks related to reaching/transportation and manufacturing operations have very different requirement regarding force, speed and impedance. Hence, designing actuation systems meeting a wide-range of requirements, when weight and volume are highly constrained, is not trivial. The proposed idea, of using robot with actuators equipped with variable transmission, directly addresses those practical challenges limiting many robotic concepts. All the concepts presented in this chapter could hugely benefit from the technology developed in this thesis.

Chapter 3

A Variable Gear-ratio Actuator with Fast and Seamless Transitions

"Simplicity is the ultimate sophistication."

–Leonardo da Vinci

This chapter presents an actuation technology, consisting of a mechanical architecture called DSDM (dual-motor dual-speed) used in conjunction with novel gear-shifting control algorithms, that make possible fast and seamless transitions between two radically different gear-ratios. Fig. 3-1 illustrates a DSDM actuator prototype.

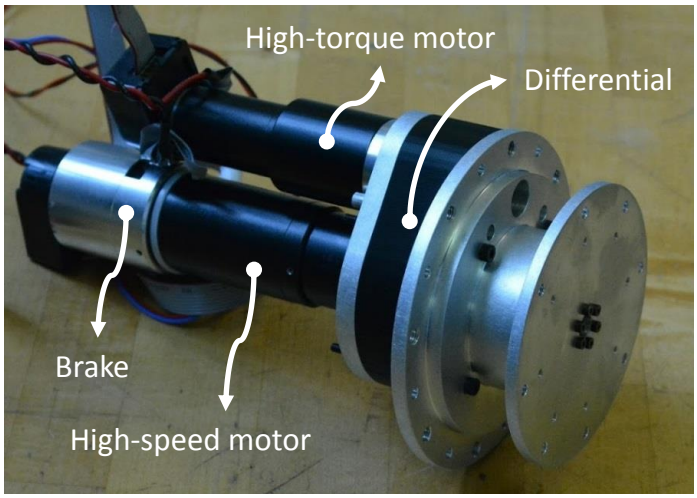


Figure 3-1: DSDM actuator prototype

This technology allows for improved power transmission over a wide range of output speed, and reflecting radically different impedance at the output. Unlike alternative variable transmission approaches, this is achieved without requiring complex or novel components, only proven technology (motor, brake and gears), which can be greatly advantageous from a product development point-of-view.

3.1 Motivation

In many robotic systems, actuators are often required to operate in distinctively different torque-speed load conditions. As illustrated on Fig. 3-2 for a typical electromagnetic (EM) actuator, extremum torque-speed conditions are not optimal in term of efficiency and power output. This often leads to the use of oversized and inefficient actuators, which is inhibitory particularly for mobile robots.

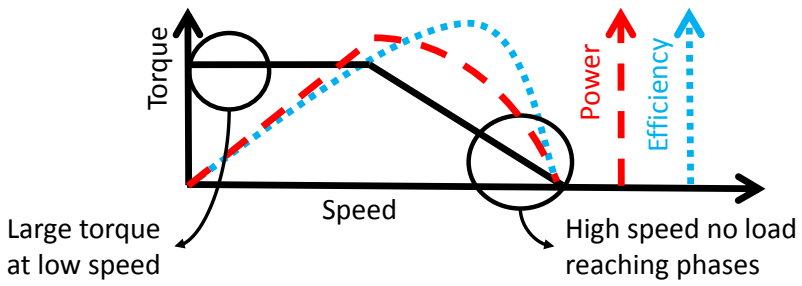


Figure 3-2: Limitations of EM motors for extremum torque-speed operations

Automobiles with internal combustion (IC) engines use transmissions with multiple gear-ratios to match torque-speed conditions. IC engines have a very narrow speed range in which they can effectively deliver power; a transmission with multiple gear ratios is a necessity for the engine to work effectively for a wide range of output speed. EM motors are more flexible than IC engines, but still far from ideal sources. EM motors cannot output high power at low speed because of thermal dissipation and magnetic flux limits related to material properties; are limited in speed by the supply tension and others; and are very inefficient when producing large forces at low speed [25]. In robotic, since it is often the extremums, i.e. maximum torque and speed, that determine the actuator design instead of the power requirement, much

can be gained with multiple gear ratios.

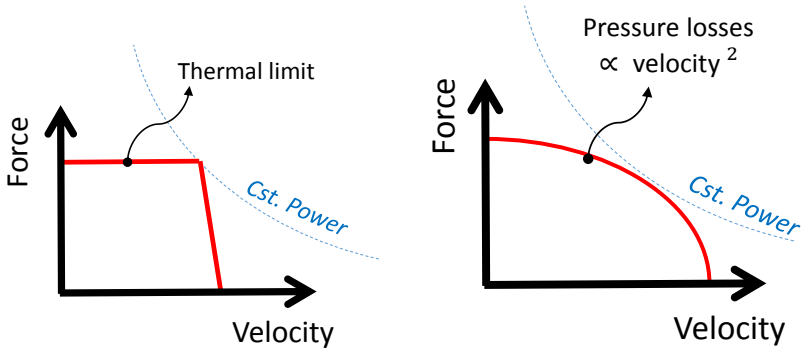
It will be a significant breakthrough if a type of multiple speed transmission can be used effectively in robotics. Even a small, lightweight actuator can generate large torques and move at high speed if equipped with both a large and a small gear-ratio. Moreover, a multiple gear-ratio transmission can allow an actuator to work closer to its optimal operating conditions, improving overall efficiency significantly. Furthermore, gear shifting significantly changes the intrinsic impedance of an actuator, since the impedance is proportional to the square of the gear-ratio. The actuator may be made back-drivable while using its small reduction ratio, an important property in many applications where the robot physically interacts with the environment [23]. Also the same actuator may be made non-back-drivable while using its large reduction ratio, allowing the actuator to support loads without consuming energy and enabling high-stiffness position control.

3.2 Actuator and powertrain research

Classical Actuators Traditional robots generally use actuators that behave as displacement-sources because of their high intrinsic impedance. These include geared EM motors and hydraulics cylinders. Using a force sensor, it is possible to control the output force with this type of actuators, but the bandwidth is rather limited. To guarantee the stability of the force-feedback scheme only half the intrinsic inertia can be canceled [23]. Since 70's, roboticists have been attempting to build actuators that can behave naturally as a force-source such as series-elastic actuators, pneumatic cylinder and air-muscles [19] [49]. However, because of the physical limitation of compliant transmission materials, the achievable bandwidth is limited and precise position control is hardly achievable. Direct drive EM actuators are the best force-source actuators with high fidelity, high bandwidth, and have been used for high-speed robots [2] and more recently small legged robots [27]. However, the very low force density [25] and low efficiency at low speeds make them impractical for most mobile robot applications, just holding a payload with static torques require

continuous currents in the motors leading to a large energy consumption even though no mechanical work is done.

Regarding power-throughput, as briefly discussed before and illustrated at Fig. 3-3a, electromagnetic actuators are typically characterized by a flat force curve for most of their range of speed, leading to maximum power been only available at high velocity [14]. Fluidic actuators are typically characterized by a force curves dropping quadratically with velocity (related to pressure losses in valves orifices), as illustrated at Fig. 3-3b. All in all, EM and fluidic actuators are not perfect power sources and could benefit from using variable transmissions to have their maximum power available on a much wider range of speed.



(a) Electromagnetic transducer (b) Fluidic transducer

Figure 3-3: Typical force-speed curve of actuators

Some advance electric motor systems can extend their operation at high-speed by weakening the magnetic flux. Such motors can thus transmit their maximum power over a wider range of speed than basic DC motors, see Fig. 3-4. However, this clever electromagnetic scheme cannot go around the fundamental force saturation at low-speed, which is limited by material properties [25]. Hence, there is still a big advantage of using multiple gear-ratio even for motor using advance flux control schemes, see Fig. 3-5.

Variable Impedance Actuators Force-source type of actuators are desirable for interaction tasks, for instance grasping, manipulation and locomotion, since the interaction force can be controlled. On the other hand, actuators with non-back-drivable

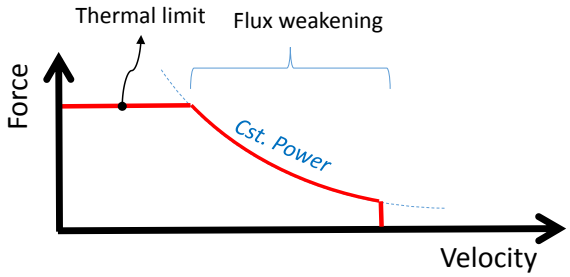


Figure 3-4: Force-speed curve of an electric motor using flux weakening

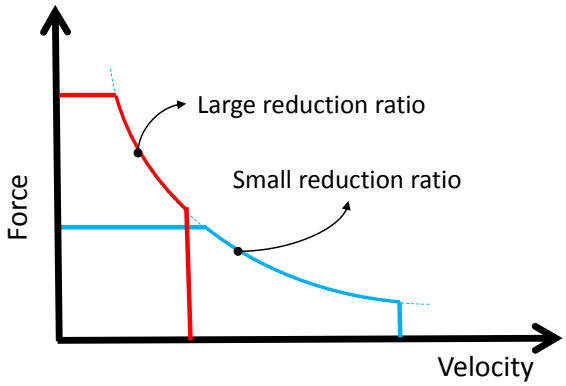


Figure 3-5: Force-speed curve of an electric motor using flux weakening with two different reduction ratios

mechanisms have the advantage for pure position controlled tasks, disturbance rejection and statically bearing large load without any power consumption. Since both small and large intrinsic impedances are advantageous in different scenario, several group have developed variable intrinsic impedance actuators, such as based on variable stiffness spring [60], antagonist non-linear devices [31], a series-compliance that can be locked with a brake [35] and dual-motors in serial configuration [29]. Furthermore, so-called macro-micro actuators, can improve the bandwidth of force-source type of actuators by exploiting the high-bandwidth of a small actuator in parallel, allowing for wider-range impedance control and improved position control [43]. Regarding power throughput however, all these technologies are still limited by force-speed characteristic of their main transducer (generally a geared electric motor). Hence, those designs do not solve the problem of efficient power transmission over a wide range of speed.

Vehicle Powertrains While the actuator work in robotics have been focused on impedance and bandwidth issues, in the powertrain field the torque-speed matching issue is predominant, since power density and efficiency are critical for mobile systems. The idea of using multiple gear ratios with electric motors has been explored occasionally, to improve efficiency and power density [40] [32] [48]. A twin motor configuration has been proposed for smooth gear shifting, where each motor shifts at a different timing [5]. Also, a dual motor configuration using a planetary coupling and non-back-drivable worm-gears was proposed for a mobile robot powertrain [36]. Multiple gear-ratio powertrains provide effective solutions for torque-speed matching, but are not adapted to the robotic context. First because powertrain shift mechanisms are not adapted to make gear shifts while interacting with dynamic environments, and second because they are designed to make shifts between gear-ratios much closer to one another than what is investigated in this thesis.

Fig. 3-6 shows typical transmitted force profile during gear-shifts with powertrain shifting mechanisms. The simplest manual transmissions using dog clutch do not transmit torque at all during a shift, while more advance systems such as dual-clutch transmission can supply torque during the transition, the fidelity is low compared to what is typically expected from an actuator in a robotic context. For instance, with a state-of-the-art two-speed dual-clutch transmission for an electric car [61], experimental results shows output torque oscillations with amplitudes of about 100% the nominal value during a period of 0.5 sec. However, this torque deviation only lead to an undesirable car acceleration of about 0.05 g, and the effect is barely noticeable on the output velocity curve. As illustrated at Fig. 3-7, force fidelity requirement for vehicle powertrain are not very severe since the load is always a very large vehicle inertia which will act as a very strong low-pass filter. However, robotic system can be interacting with all kind of load without filtering characteristics. For instance, for a robot fighting a gravitational load or compressing a spring, it would be catastrophic if the force drop during a gear-shift. The output needs to be always fully under control in those situations.

One other aspect is that power-train gear-shifting mechanisms are not adapted to

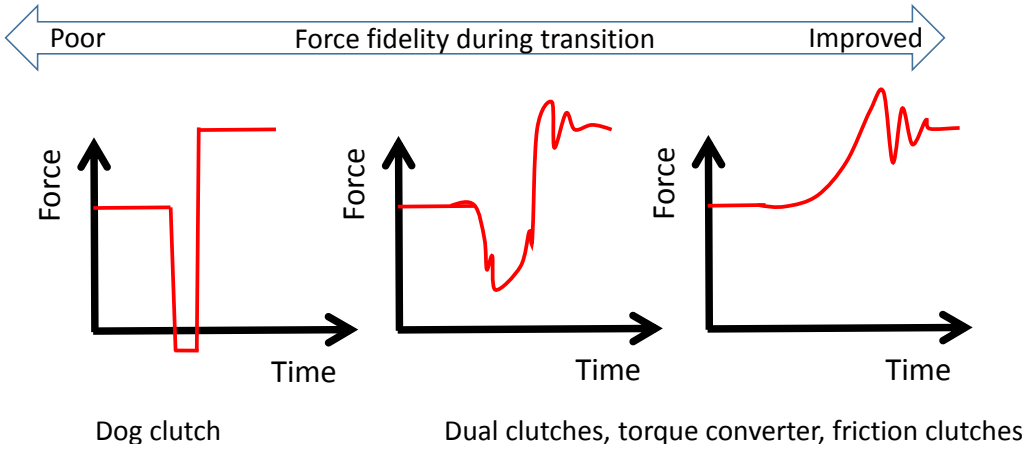


Figure 3-6: Force profile during a gear-shift with powertrain gear-shifting mechanisms

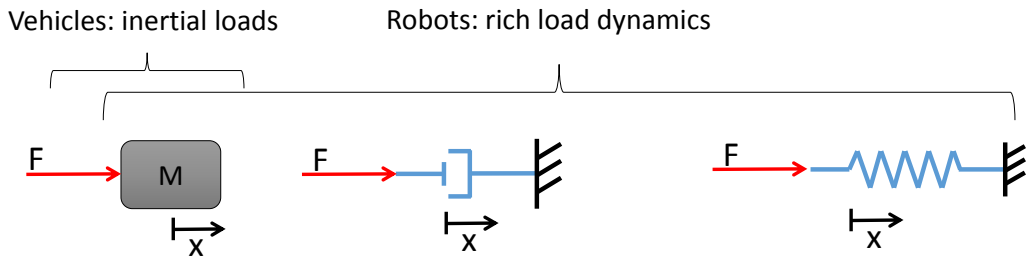


Figure 3-7: Type of loads encountered by vehicle powertrains vs. robot actuators

make transition between drastically different gear-ratios. For instance, for dual-clutch systems the ratio step between adjacent gear ratios should not exceed 1.8 to avoid shift difficulties [11]. Furthermore, from a design perspective, advance power-train systems are very complex machines (wet clutches, dry clutches, synchronizers, etc.) leading to manufacturing, maintenance, wear and reliability challenges. Adding such systems in all the many actuators of a robotic system might be especially hard to justify when balancing all those practical issues. All-in-all, power-train gear-shifting technologies are not adapted directly for use in a general purpose robotic actuator.

Variable Gear-ratio Actuators While variable gear-ratio actuators (VGA) have been studied extensively for automobile power-trains, they have not yet been fully investigated in robotics, despite significant potential gains. A few instance of research in that direction were made for legged locomotion [21], grasping robotic hands [55] and actuators [20] [28] [59]. While those works are promising, no gear shifting methodology

for generic robot actuator in arbitrary dynamic situation are proposed.

Continuously variable transmissions (CVT) Continuously variable transmission have been used sporadically for car power-trains. Most common designs are based on belts with variable-diameter pulleys or toroidal disks. Drawbacks compared to regular transmission include higher transmission losses and complex dynamics [57]. Moreover, typical CVT have limited total ratio variation range (for instance 0.5:1 to 2:1), which makes them un-adapted for very large ratio variation. Some designs have been proposed for infinite range variation, often called IVT. For instance the company *Torotrack* claims to have a CVT that can reach an effective gear-ratio of zero [54]. However, since those designs rely on friction, maximum transmitted torque is limited which limits the effective large reduction range. Moreover, such transmission systems are highly complex and very large (requiring a hydraulics system to control the pre-load forces for instance), which inhibit the potential use for smaller scale actuators in robotic applications.

In the actuator field, lever mechanisms with variable attachment points that lead to a very wide range of effective transmission ratio [59], even infinite range when using a singular configurations [26], have been proposed. However, this type of CVT implementation limits drastically the motion range, thus cannot be used for general purpose actuator transmission. In the literature, those mechanisms are used in VSA between the spring and the output, where their limited motion range is not an issue.

Many clever mechanisms have proposed to be used as CVT. However, all designs have some major drawback regarding either: total variation range, constraints on output motion, ratio-variation speed, allowable shift conditions, efficiency, etc. The best indication of this is the automotive industry. Theoretically, there is a huge incentive to use CVT with internal combustion engined because of their narrow peak of power and efficiency. However, despite a century of development in one of the largest industry, transmissions using many discrete gear-ratios are still the most widely adopted solution. Even with the recent efforts to improve fuel economy, the trend in the industry is to use transmissions with a large number of discrete gear-ratios [47] [17].

Furthermore, compared to the automotive field, the incentive of using CVT in robotics is diminished because of more flexible torque curves of electric motor and the challenges are greater because of the wider desired range of ratios. To conclude, all the limitations of CVT, especially the limited variation range for typical designs, make this technology un-adapted for general purpose robotic actuators. However, a breakthrough in term of IVT technology would be very interesting for the field of robotics.

3.2.1 Novel contribution

The presented DSDM actuator in this chapter, address the issue of improving available power and efficiency over a wide range of operating speeds, which has rarely been addressed in the robotics literature. Also, the actuator enables order-of-magnitude variation of the output impedance, which is also a highly desirable feature. **The main novel contribution is the methodology for gear shifting between two very different gear-ratios seamlessly even in highly dynamic situations, which is a key enabling feature for robotic applications.** A mechanical architecture where two motors are coupled using a 3-ports gearbox and a brake is used in conjunction with novel control scheme to provide full control of the output during gear shifting. The mechanical architecture is not new by itself as similar architectures (using planetary and brakes) are used in hybrid car powertrains, automatic transmissions and special actuators. However, here this architecture is used in conjunction with a novel controller to provide full control of the output during gear shifting. A preliminary version of this work has been published by the author in [14], but this chapter includes a more throughout analysis and new algorithms for fast gear-shift even during impacts.

To the knowledge of the author, no other technology meets all those requirements:

- Fast shifting between order-of-magnitude different gear-ratios
- High-fidelity control of the output during transitions
- Simple mechanical design enabling small and practical implementations

3.2.2 Related works

Many dual-motor actuators have been proposed in the literature [58], with different goals and architectures. So called macro-micro actuators, are essentially series-elastic actuators equipped with an additional small motor directly attached to the output [43]. Variable stiffness actuators (VSA) are also based on series elastic actuator architecture, where a small motor can modulate either the spring directly or the transmission between the spring and the output [26]. Fig. 3-8 shows a generalized bond-graph actuator model illustrating the conceptual differences between many approaches.

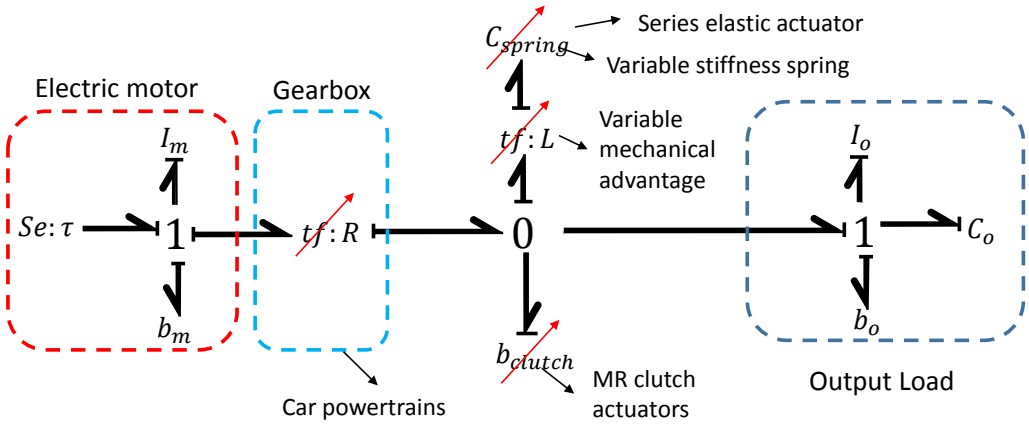


Figure 3-8: Different approaches for variable impedance actuators

Variable stiffness actuators use a variable transmission placed between a compliant element and the load, where it influence the reflected output stiffness but not the steady-state power-transmission characteristics. Macro-micro actuators have a fixed reflected output stiffness, but the advantage is that the additional direct-drive motor on the output makes possible to emulate a wide-range of output impedance. **The proposed VGA actuators in this thesis are fundamentally differents, it is the transmission between the motor and the load that is varied, like in a car transmission.** The main advantage of VGA is regarding efficient power transmission, enabling small motors to make full use of their maximum power at high speed and at low speed. Macro-micro actuators and VSA have no advantages over a regular electric motor regarding the range of speed at which power can be

transmitted. On the other hand, macro-micro and VSA have the ability to store and release potential energy in their compliant element, which is not the case with VGA. Regarding, natural reflected impedance, both VSA and VGA have the ability to change it. One fundamental difference is that VGA can only attenuate or amplify the natural inertia and friction of the rotor, while VSA can only modify its reflected stiffness.

3.3 Dual-Speed Dual-Motor architecture

The proposed architecture, referred to as a Dual-Speed Dual-Motor (DSDM) actuator, consists of a direct drive motor (M1) equipped with a locking brake and an geared EM motor (M2) with a large reduction ratio coupled to the same output through a differential, see Fig. 3-9. The differential can be viewed as a 0-type junction (taking bond-graph terminology) where the speeds add up and the force is shared.

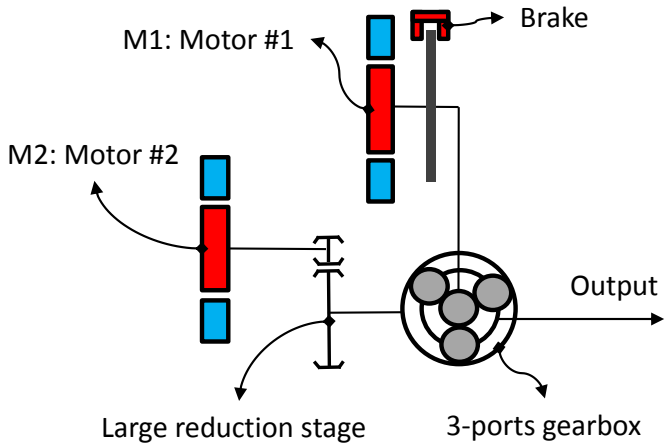


Figure 3-9: DSDM actuator concept

The envisioned implementation of the DSDM concept is to embed all the components into a single compact unit, as illustrated by Fig. 3-10. A lot of weight and space could be saved by combining the reduction and the differential gearing and having all the components inside a single housing.

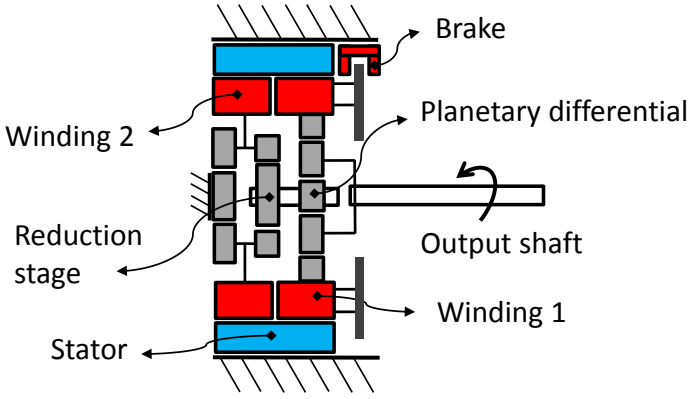


Figure 3-10: Possible architecture of an integrated DSDM concept

3.3.1 Principle

The DSDM can be used in two modes, high-force mode when the brake is closed and high-speed mode when the brake is open. The result is like having two very different reduction ratio you can choose from during operation. Fig. 3-11 conceptually illustrates the principle with a leverage analogy, M1 acts like a force source connected almost directly to the output and M2 acts like a displacement source with a large lever arm relative to the output.

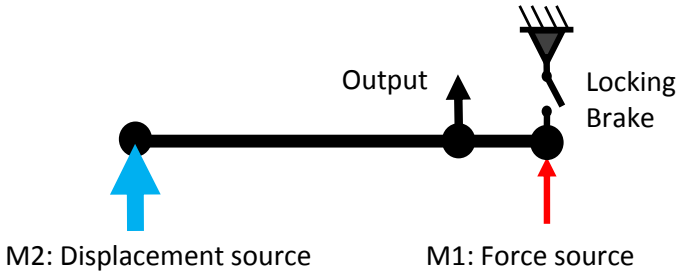
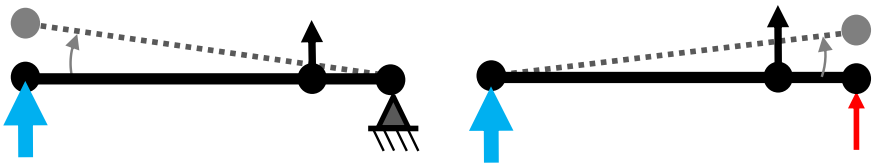


Figure 3-11: Dual inputs system

During the high-force mode, see Fig. 3-12a, the brake is closed and M2 drives the output with a large mechanical advantage. The result is a low-speed displacement-source type of actuation like a geared EM motor. During the high-speed mode, see Fig. 3-12b, M1 drive the output almost directly, creating a high-speed force-source actuator like a direct drive EM motor. Additionally, both motors can be used simultaneously to drive the output even faster.



(a) High force mode (brake closed) (b) High speed mode (brake open)

Figure 3-12: Two modes of operation

Fig. 3-13 illustrates the operating range of the DSDM actuator plotted on the standard torque-speed plane. The high-force mode region is determined by the per-

formance of M2 alone, since M1 is locked. The high-speed mode region can exceed the performance of M1 alone, as M2 can be used simultaneously to increase the output speed. The fail safe zone indicates the guaranteed performance of the DSDM actuator in case of failure in either motor.

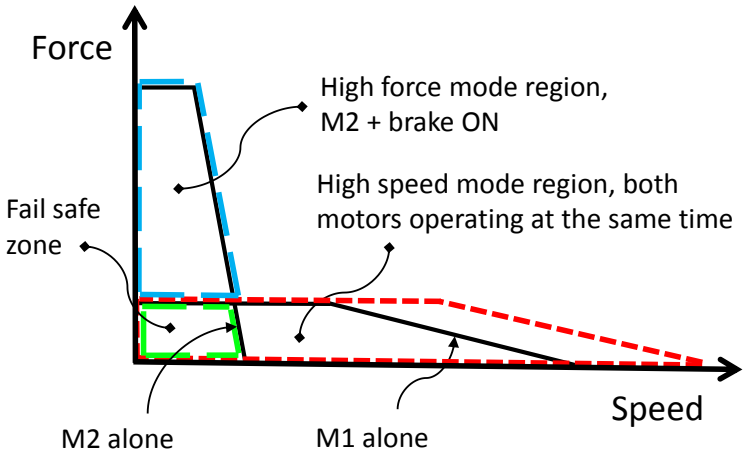
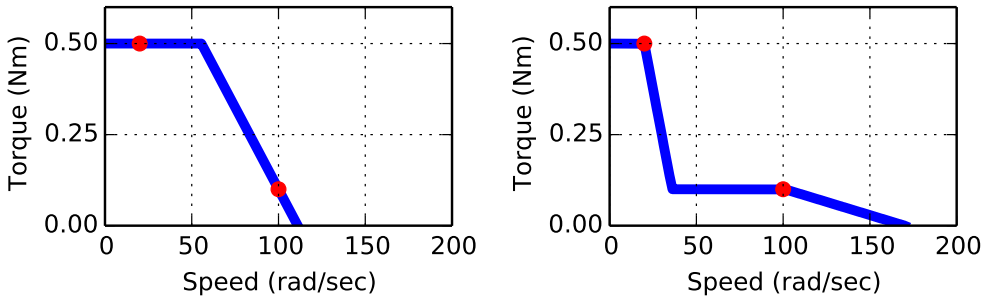


Figure 3-13: DSDM actuator operation regions

3.3.2 Weight advantage

A DSDM actuator will be lighter than a single motor for applications with a wide range of operating speed. Suppose that an actuator must generate 10 W output power at two operating points: 0.5 Nm of torque at a speed of 20 rad/sec and 0.1 Nm at 100 rad/sec. A single EM motor that satisfies these requirements at both operating points tends to be oversized in terms of power, to reach both operating points, see Fig. 3-14a. A DSDM actuator can reach the same operating points using two smaller motors with appropriate gear ratios, see Fig. 3-14b. On the other hand the DSDM actuator uses more components: two motors instead of one, more gearing and an additional brake. The DSDM concept pays-off when the difference in speed between two required operating points becomes larger. Fig. 3-15 shows the estimated weight of actuators in relation to the ratio of operating speeds ($\lambda = \frac{w_1}{w_2}$), while the required power output is kept at 10 W. The actuator weight is computed assuming that the mass of each component is proportional to its maximum output torque, with values taken from commercially available *Maxon motor* components in the 10 - 100 watts

range: 2 kg/Nm for motors, 0.1 kg/Nm for gearboxes and differentials and 0.2 kg/Nm for brakes. As shown in Fig. 3-15, the DSDM concept becomes advantageous when there is a large speed difference between the operating points. This is because only the gearbox and brake need to be scaled up for the DSDM actuator to meet the high torque requirement of the low-speed operating point, while the motor size must be increased for the single motor solution.



(a) One motor solution (b) DSDM solution

Figure 3-14: Case study of two actuator solutions for two 10 W operating points

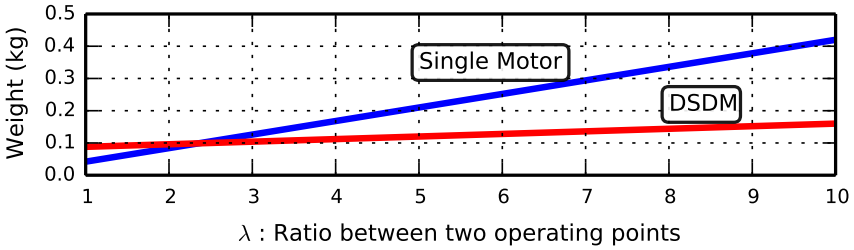


Figure 3-15: Weight of a single motor compared to the DSDM concept for two 10 W operating points at different speeds $w_1 = 100$ rad/sec, $w_2 = w_1/\lambda$

Note that a two-speed actuator using a single motor and a variable transmission similar to the type used in car power-trains could have an even larger weight advantage over a single-gear motor. However, this type of variable transmissions, using components such as dog clutch, synchronizers and friction clutches, would not exhibit the features required to change gear-ratio seamlessly in the dynamic situations encountered by robots, unlike the DSDM architecture.

3.3.3 Efficiency advantage

VGA actuators, including DSDM actuators, can transmit power more efficiently given various operating conditions. Electric motor efficiency is a function of velocity and applied torque. The efficiency map depends on the electric motor type (DC brushed, AC induction, DC brushless, etc.). As a general rule, motors are typically more efficient in the upper-end of their velocity range. Hence, by using multiple gear-ratios, not only motors can be down-sized, but power transmission can be made more efficient. For instance, switching to a large gear-ratio to use a motor at its most efficient operating conditions with high rotor speeds. This efficiency advantage has been studied for electric cars equipped with multiple speed transmissions [51] [24] [64] [40]. This advantage is also of high interest for any mobile robots where on-board energy is limited. Any quantitative analysis of this benefit depends heavily on the specific of the type of motor used, the controller, the robotic system and the task executed by the robotic system. Section 4.7 will offer some quantitative simulation results regarding reduced energy consumption of robotic manipulator achieved using VGA actuators.

Another significant efficiency advantage of DSDM actuators over single-gear motors, is that high-speed mode can be backdrivable for interaction tasks while the high-force mode can be made non-backdrivable (using a irreversible large reduction for M2) in order to be able to hold an object against gravity without having to consume any electrical energy. A backdrivable single-gear motor will always have to supply electrical power just to sustain gravity forces, leading to zero efficiency for holding tasks. Hence, for a robot requiring backdrivability in some task and often holding objects against gravity, the efficiency gain could be huge. Note that some industrial robot arms use brake mounted on actuator outputs to address this type of energy consumption issue [41].

An additional advantage of the DSDM architecture, is that during high-speed operation, many combination of M1 and M2 velocity can lead to the same output velocity. Hence, this internal degree of freedom can be used to further optimize

efficiency by distributing motor speeds to minimize the overall energy consumption.

3.3.4 Reliability advantage

An additional secondary advantage of the DSDM architecture is that some minimum performance, illustrated by the green area at Fig. 3-13, can be guaranteed even if either motor fail. Here failures leading to either jamming (rotor and stator stuck together) or freewheeling (motor cannot transmit any torque) are considered. Table 3.1 details how those failures mode can be addressed with the DSDM architecture.

Table 3.1: Safe failure modes of a DSDM actuator

Failure Mode	**Fail-safe Operation**
Jamming of M1	If M1 jams, then the situation is the same as if the brake would be engaged, and the capability of high-force mode is still available.
Jamming of M2	If M2 jams, then M1 can still be used freely to move the output. The capability of high-speed are almost fully available, only the maximum speed is reduced as M2 is not available to add-up speed.
Freewheeling of M1	If M1 is no longer able to transmit any torque, than if the brake is engaged all the capability of high-force mode are still available.
Freewheeling of M2	If M2 is no longer able to transmit any torque, M1 can still be used to move the output. Assuming the reduction stage of M2 is irreversible or barely backdrivable (meaning its associated moving part in the differential is still fixed and can sustain reaction forces) then the capability of high-speed are fully available.

3.4 Modeling

This section derives mathematical equations describing the behavior of a DSDM actuator for the purpose of designing adequate control laws.

3.4.1 3-ports planetary gear junction

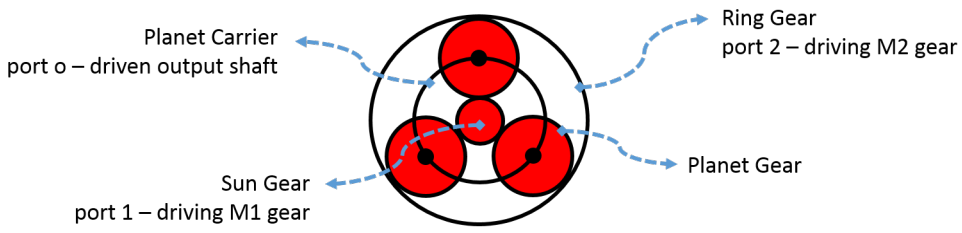


Figure 3-16: Planetary gear-box used as a 3-port junction

A planetary gear box is used to implement the 3-port differential junction that links the two motors to the output. As illustrated by Fig. 3-16, the planet carrier is connected to the output, M1 to the sun gear and M2 to the ring gear. Note that in typical gear-reducers using a planetary, the ring gear is usually fixed and there is a single DoF in the gearing. Here the ring gear is also mounted on bearing and there is 2 DoF in the gearing. The kinematic relation of the system is given by

$$w_o = \underbrace{\left[\frac{1}{N+1}\right]}_{1/R_1} w_1 + \underbrace{\left[\frac{N}{r_2(N+1)}\right]}_{1/R_2} w_2 \tag{3.1}$$

where r_2 is the additional reduction of M2, N is the ratio of gear teeth of the ring gear over the sun gear, and w_o, w_1 and w_2 are angular velocities of the output shaft (port o), M1 input shaft (port 1) and M2 input shaft (port 2). Neglecting internal inertial forces in the gearing, the effort relation of the system is given by:

$$-e_o = \underbrace{[N+1]}_{R_1} e_1 = \underbrace{\left[\frac{r_2(N+1)}{N}\right]}_{R_2} e_2 \tag{3.2}$$

Hence, the 3-ports planetary coupling can be interpreted as a 0-junction, in the bond

graph terminology, with different mechanical advantages (R_1 and R_2) on each input ports.

3.4.2 Dynamics

Fig. 3-17 shows a lumped-parameter dynamic model of a DSDM when the brake is open (high-speed mode). I_i and b_i are the inertia and damping of the respective i-th ports. It will be assumed here that low-level high-bandwidth current controllers are used, and electromagnetic torques τ_1 and τ_2 are going to be considered directly as inputs to the system. An equivalent bond-graph model is illustrated at Fig. 3-18.

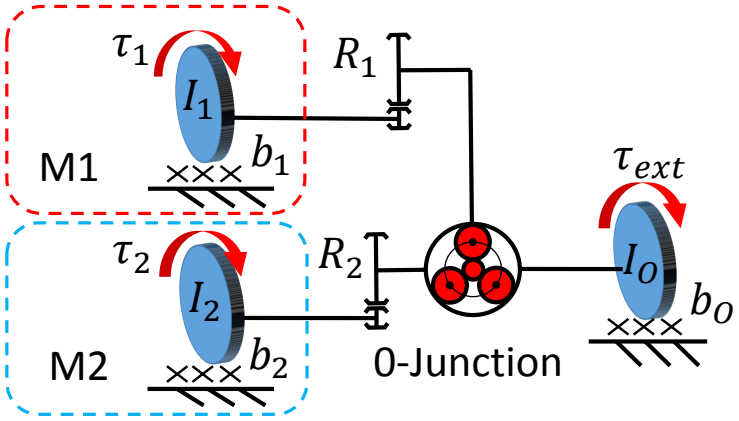


Figure 3-17: Lumped-parameter dynamic model of a DSDM

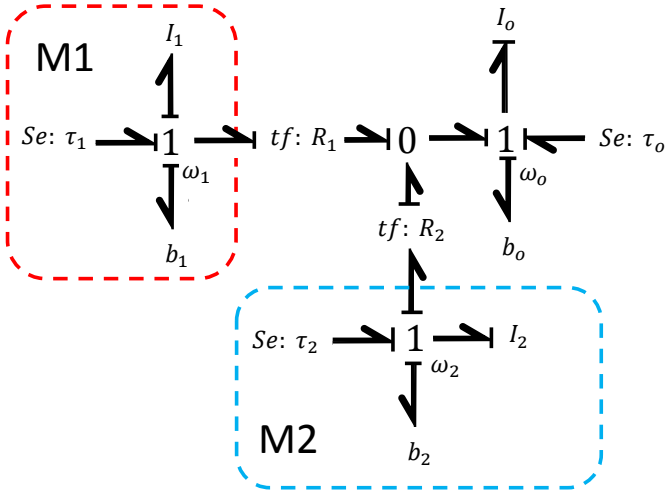


Figure 3-18: Bond-graph dynamic model of a DSDM

Applying Newton's law on each ports yields the following equations of motions:

$$\tau_{ext} - e_o = Z_o(s)w_o \tag{3.3}$$

$$\tau_1 - e_1 = Z_1(s)w_1 \tag{3.4}$$

$$\tau_2 - e_2 = Z_2(s)w_2 \tag{3.5}$$

where $Z_i(s) = I_i s + b_i$ represents the mechanical impedance of the i-th ports in the Laplace domain. Note that the system is coupled due to the constraint given by eq. (3.1) and (3.2), and that there is only two degrees of freedom among the three ports. Fig. 3-19, illustrate the coupled equations motion in block diagram form. It is then

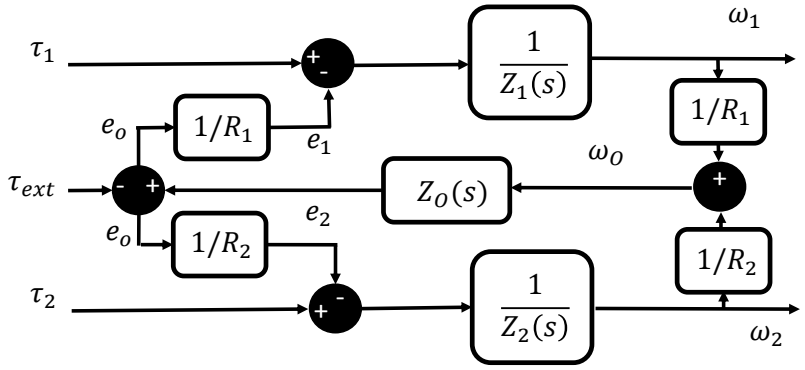


Figure 3-19: Dynamics of a DSDM illustrated with a block diagram

possible to eliminate one variable and express the dynamic as the following system of two equations:

$$\begin{bmatrix} 1 & 0 & \frac{1}{R_1} \\ 0 & 1 & \frac{1}{R_2} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_{ext} \end{bmatrix} = \begin{bmatrix} Z_1(s) + \frac{Z_0(s)}{R_1^2} & \frac{Z_0(s)}{R_1 R_2} \\ \frac{Z_0(s)}{R_1 R_2} & Z_2(s) + \frac{Z_0(s)}{R_2^2} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{3.6}$$

The equation of motion can then be rearranged in the standard manipulator equation form, using as generalized coordinates for more convenience the following:

$$\begin{bmatrix} w_o \\ w_1 \end{bmatrix} = \boldsymbol{w} = \dot{\boldsymbol{q}} \tag{3.7}$$

Note that since there are no conservative forces in the actuator, all displacement coordinates are ignorable and the dynamic equation is independent of rotor and output angles.

$$\underbrace{\begin{bmatrix} I_o+R_2^2 I_2 & -\frac{R_2^2}{R_1} I_2 \\ -\frac{R_2^2}{R_1} I_2 & I_1+(\frac{R_2}{R_1})^2 I_2 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} \dot{w}_o \\ \dot{w}_1 \end{bmatrix}}_{\dot{\boldsymbol{q}}} + \underbrace{\begin{bmatrix} b_o+R_2^2 b_2 & -\frac{R_2^2}{R_1} b_2 \\ -\frac{R_2^2}{R_1} b_2 & b_1+(\frac{R_2}{R_1})^2 b_2 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} w_o \\ w_1 \end{bmatrix}}_{\dot{\boldsymbol{q}}} = \underbrace{\begin{bmatrix} 0 & R_2 & 1 \\ 1 & -\frac{R_2}{R_1} & 0 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_{ext} \end{bmatrix}}_{\boldsymbol{\tau}} \tag{3.8}$$

The inverse of the inertia matrix is given by:

$$H^{-1} = \frac{1}{I_o + I_1 R_1^2 + I_o \frac{I_1}{I_2}(\frac{R_1}{R_2})^2} \begin{bmatrix} 1+\frac{I_1}{I_2}(\frac{R_1}{R_2})^2 & R_1 \\ R_1 & \frac{I_0}{I_2}(\frac{R_1}{R_2})^2 + R_1^2 \end{bmatrix} \tag{3.9}$$

The equations can be converted to linear state space form:

$$\underbrace{\dot{\boldsymbol{w}}}_{\dot{\boldsymbol{x}}} = \underbrace{\left[-H^{-1}D\right]}_{F} \underbrace{\boldsymbol{w}}_{\boldsymbol{x}} + \underbrace{\left[H^{-1}B\right]}_{G} \underbrace{\boldsymbol{\tau}}_{\boldsymbol{u}} \tag{3.10}$$

Leading to the following after eliminating the external torque and the damping at each motor port for brevity:

$$\begin{bmatrix} \dot{w}_o \\ \dot{w}_1 \end{bmatrix} = F \begin{bmatrix} w_o \\ w_1 \end{bmatrix} + G \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{3.11}$$

$$\text{with} \quad F = \frac{1}{I_T} \begin{bmatrix} -b_T & 0 \\ -R_1 b_o & 0 \end{bmatrix} \tag{3.12}$$

$$G = \frac{1}{I_T} \begin{bmatrix} R_1 & R_1 \frac{R_1 I_1}{R_2 I_2} \\ (R_1^2 + \frac{R_1^2 I_o}{R_2^2 I_2}) & -\frac{R_1 I_o}{R_2 I_2} \end{bmatrix} \tag{3.13}$$

$$I_T = \left[I_o + R_1^2 I_1 + \left(\frac{R_1}{R_2}\right)^2 \frac{I_1}{I_2} I_o\right] \tag{3.14}$$

$$b_T = \left[b_o + \left(\frac{R_1}{R_2}\right)^2 \frac{I_1}{I_2} b_o\right] \tag{3.15}$$

3.4.3 Inputs/Outputs equations

The variable of interest is the output w_o, and its dynamics can be expressed, going back to the Laplace domain by:

$$\left[Z_1(s)Z_2(s) + \frac{Z_1(s)Z_o(s)}{R_2^2} + \frac{Z_2(s)Z_o(s)}{R_1^2}\right] w_o(s) = \tag{3.16}$$

$$\left[\frac{Z_2(s)}{R_1}\right] \tau_1(s) + \left[\frac{Z_1(s)}{R_2}\right] \tau_2(s) + \left[\frac{Z_2(s)}{R_1^2} + \frac{Z_1(s)}{R_2^2}\right] \tau_{ext}(s) \tag{3.17}$$

When the brake on M1 of the DSDM is locked, the output equation is reduced, by letting $Z_1(s) \to \infty$, to:

$$\left[Z_o(s) + R_2^2 Z_2(s)\right] w_o(s) = [R_2]\, \tau_2(s) + \tau_{ext}(s) \tag{3.18}$$

When the brake is open, if the gear-ratio R_2 of M2 is large, the equation can be simplified. Assuming the reflected impedance of M1 is much smaller than that of M2, and neglecting motor side damping, the equation is reduced to:

$$\left[Z_o(s) + R_1^2 Z_1(s)\right] w_o(s) = [R_1]\, \tau_1(s) + \left[R_1 \frac{R_1 I_1}{R_2 I_2}\right] \tau_2(s) + \tau_{ext}(s) \tag{3.19}$$

$$\text{with} \quad Z_1(s)R_1^2 << Z_2(s)R_2^2 \quad \text{and} \quad \frac{Z_1(s)}{Z_2(s)} = \frac{I_1}{I_2} \tag{3.20}$$

During high-speed mode the behavior of the output is also dominated by a first-order linear behavior, but interestingly both input torques contributed to the motion through inertial coupling. Note that this differ from a serial architecture, in both case speed adds-up and effort is shared (0-type junction), but inertial properties are different. Input/output differential equations are thus given by:

$$\text{High-speed mode:} \quad \left[I_o + R_1^2 I_1\right] \dot{w}_o + [b_o]\, w_o = [R_1]\, \tau_1 + \left[R_1 \frac{R_1 I_1}{R_2 I_2}\right] \tau_2 \tag{3.21}$$

$$\text{High-force mode:} \quad \left[I_o + R_2^2 I_2\right] \dot{w}_o + [b_o]\, w_o = [R_2]\, \tau_2 \tag{3.22}$$

Note that, when a DSDM is connected to a robotic system, the left-hand side of equations (3.21) and (3.22) would have terms reflecting other extrinsic load-side forces: including coupled inertial effects, gravitation forces and others. However, the right-hand side of the equation would stay the same: the motor torques transmission gains would not be affected:

$$\left[I_o + R_1^2 I_1\right] \dot{w}_o + [b_o]\, w_o - \underbrace{\tau_{ext}}_{Extrinsic forces} = [R_1]\, \tau_1 + \left[R_1 \frac{R_1 I_1}{R_2 I_2}\right] \tau_2 \tag{3.23}$$

$$\left[I_o + R_2^2 I_2\right] \dot{w}_o + [b_o]\, w_o - \underbrace{\tau_{ext}}_{Extrinsic forces} = [R_2]\, \tau_2 \tag{3.24}$$

Also, although neglected in (3.21) and (3.22), motor damping could be reintroduce easily by substituting motor torque τ_i with the effective torque $\tau_i - b_i w_i$ since those forces are collocated.

3.4.4 Hybrid Behavior

Fig. 3-20 illustrates the two different discrete modes of the system and possible transitions. Table 3.2 gives references to the equations modeling each of the discrete behaviors and transitions.

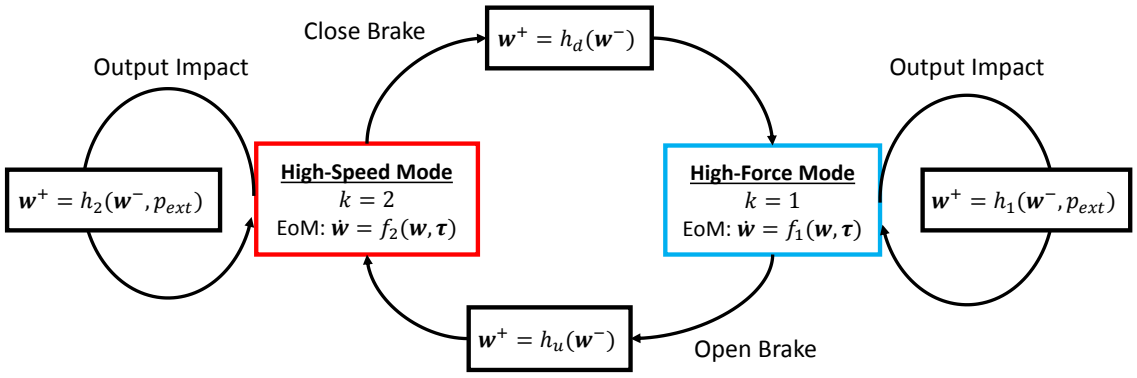


Figure 3-20: Discrete operating modes of the DSDM

Table 3.2: Hybrid model: Continuous equations and discrete jump maps

Situation	Discrete Mode	Mapping	Equation of motions
Continuous differential equations			
High-Force Mode	$k = 1$	$\dot{\boldsymbol{w}} = f_1(\boldsymbol{w}, \boldsymbol{\tau})$	given by eq. (3.18)
High-Speed Mode	$k = 2$	$\dot{\boldsymbol{w}} = f_2(\boldsymbol{w}, \boldsymbol{\tau})$	given by eq. (3.10)
Discrete gear-shift jump maps			
Up-shift	$k : 1 \to 2$	$\boldsymbol{w}^+ = h_u(\boldsymbol{w}^-)$	given by eq. (3.27)
Down-shift	$k : 2 \to 1$	$\boldsymbol{w}^+ = h_d(\boldsymbol{w}^-)$	given by eq. (3.28)
Output impact jump maps			
Impact during HF	$k : 1 \to 1$	$\boldsymbol{w}^+ = h_1(\boldsymbol{w}^-, p_{ext})$	given by eq. (3.35)
Impact during HS	$k : 2 \to 2$	$\boldsymbol{w}^+ = h_2(\boldsymbol{w}^-, p_{ext})$	given by eq. (3.32)

During high-force mode (labeled $k = 1$), when the brake is engaged, and high-speed mode (labeled $k = 2$), when the brake is open, the behavior of the system can be described by discrete sets of continuous differential equations $\dot{\boldsymbol{w}} = f_k(\boldsymbol{w}, \boldsymbol{\tau})$. It is also possible that during operation the actuator output hits an object leading to an

impulsive behavior. This behavior is modeled by the appropriate reset map $\boldsymbol{w}^+ = h_k(\boldsymbol{w}^-)$, depending on the active mode. Also, at the moment the brake is opened or closed, the discrete mode and the states also instantaneously change, according to some mapping (up-shift $\boldsymbol{w}^+ = h_u(\boldsymbol{w}^-)$ and down-shift $\boldsymbol{w}^+ = h_d(\boldsymbol{w}^-)$).

3.4.5 Continuous differential equations

High-Force Mode $k = 1$

From eq. (3.18), the continuous differential equation mapping during high-force mode can be written as:

$$\begin{bmatrix} \dot{w}_o \\ \dot{w}_1 \end{bmatrix} = f_1(\boldsymbol{w}, \boldsymbol{\tau}) = \begin{bmatrix} \frac{1}{I_0 + R_2^2 I_2} \left(-(b_o + R_2^2 b_2) w_o + R_2 \tau_2 + \tau_{ext}\right) \\ 0 \end{bmatrix} \tag{3.25}$$

High-Speed Mode $k = 2$

From eq. (3.10), the continuous differential equation mapping during high-speed mode can be written as:

$$\begin{bmatrix} \dot{w}_o \\ \dot{w}_1 \end{bmatrix} = f_2(\boldsymbol{w}, \boldsymbol{\tau}) = \left[-H^{-1}D\right] \boldsymbol{w} + \left[H^{-1}B\right] \boldsymbol{\tau} \tag{3.26}$$

3.4.6 Gear-shift events

Up-Shift $k : 1 \to 2$

During high-force mode, the only DoF is described by the variable w_o. Opening the brake release a constraint in the system and it does not lead to any impulsive behavior. M1 rotor is simply suddenly free to move starting from rest. The mapping is thus given by:

$$\begin{bmatrix} w_o^+ \\ w_1^+ \end{bmatrix} = h_u(\boldsymbol{w}^-) = \begin{bmatrix} w_o^- \\ 0 \end{bmatrix} \tag{3.27}$$

Down-shift $k : 2 \rightarrow 1$

For a down-shift the system goes from 2-DoF to 1-DoF, hence the sudden addition of a constraint (brake locked) can lead to an impulsive behavior. However, as it will be discuss in the next section, the controller will always make sure that M1 is at zero velocity before engaging the brake in order to avoid this impulsive behavior.

In normal operation, M1 velocity will be brought to zero before engaging the brake and the mapping is thus smooth and given by:

$$\begin{bmatrix} w_o^+ \\ w_1^+ \end{bmatrix} = h_d(\boldsymbol{w}^-) = \begin{bmatrix} w_o^- \\ 0 \end{bmatrix} \quad \text{if} \quad w_1^- = 0 \tag{3.28}$$

If the brake would be engaged with non-zero w_1 velocity the output would exhibit an undesirable impulsive deceleration:

$$\begin{bmatrix} w_o^+ \\ w_1^+ \end{bmatrix} = h_d(\boldsymbol{w}^-) = \begin{bmatrix} w_o^- - \frac{w_1^-}{R_1\left(\frac{I_o}{I_2 R_2^2}+1\right)} \\ 0 \end{bmatrix} \tag{3.29}$$

The impulsive mapping of eq.(3.29) is derived using eq.(A.22), where the contact Jacobian is defined by the zero velocity constraint on M1 when the brake closes:

$$0 = J_c \boldsymbol{w} = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \tag{3.30}$$

3.4.7 Output Impacts

During a contact, impulsive forces can create a sudden change of velocity. Hence, if a DSDM actuator is used on a robotic system making contact with objects, its internal velocities can suddenly change. Since there is no impulsive forces on the motor side, the effect of a contacts on a DSDM actuator can be reduce to an impulsive torque applied to the output. The output impulse p_{ext} (integral of the impulsive external torque during the impact) is thus sufficient to characterized the internal velocity

jumps:

$$p_{ext} = \int_{t^-}^{t^+} \tau_{ext} dt \tag{3.31}$$

Section A.3.3 discusses how such impulsive forces can be computed with a dynamic model of a robotic system.

During High-speed Mode

During high-speed mode, applying eq. (A.18), it leads to impulsive map:

$$\begin{bmatrix} w_0^+ \\ w_1^+ \end{bmatrix} = h_2(\boldsymbol{w}^-, p_{ext}) = \begin{bmatrix} w_0^- \\ w_1^- \end{bmatrix} + H^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} p_{ext} \tag{3.32}$$

If the reflected inertia of M2 is much greater than that of M1 ($R_2^2 I_2 >> R_1^2 I_1$), than the impulsive reaction can be simplified to

$$\begin{bmatrix} w_0^+ \\ w_1^+ \end{bmatrix} = \begin{bmatrix} w_0^- \\ w_1^- \end{bmatrix} + \frac{p_{ext}}{I_o + I_1 R_1^2} \begin{bmatrix} 1 \\ R_1 \end{bmatrix} \tag{3.33}$$

Hence, M2 velocity will be unchanged and the velocity discontinuity of the output is transmitted directly to M1 during an impact:

$$\Delta w_o = \frac{p_{ext}}{I_o + I_1 R_1^2} \qquad \Delta w_1 = R_1 \Delta w_o \qquad \Delta w_2 = 0 \tag{3.34}$$

During High-force Mode

During high-force mode, assuming the brake is strong enough not to slip during the impact, the impulsive mapping is given by :

$$\begin{bmatrix} w_0^+ \\ w_1^+ \end{bmatrix} = h_1(\boldsymbol{w}^-, p_{ext}) = \begin{bmatrix} w_0^- + \frac{p_{ext}}{I_o + R_2^2 I_2} \\ 0 \end{bmatrix} \tag{3.35}$$

3.4.8 Nullspace of the system during high-speed mode

Kinematic

During high-speed mode, from the kinematic input-output view point, the DSDM actuator has one redundant degree of freedom. In other words, there is an infinite number of combinations of w_1 and w_2 producing the same output speed w_0, from eq.(3.1):

$$w_o = \begin{bmatrix} \frac{1}{R_1} & \frac{1}{R_2} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{3.36}$$

A vector perpendicular to the above coefficient vector forms the null space of the DSDM actuator system. Any input combination in this direction produces zero output speed:

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 \\ -R_2/R_1 \end{bmatrix}}_{\text{Nullspace Projection}} u \quad \Rightarrow \quad w_0 = 0 \quad \forall u \in \Re \tag{3.37}$$

Dynamic

Interestingly, a similar expression can be obtained for the dynamics of the output in response to electromagnetic torque inputs, from the first line of eq.(3.11):

$$I_T \dot{w}_o + b_T w_o = \begin{bmatrix} R_1 & R_1 \frac{R_1 I_1}{R_2 I_2} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{3.38}$$

Hence, there is a one degree of freedom space of inputs τ_1 and τ_2 that do not affect the output:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \underbrace{\begin{bmatrix} I_1 \\ -\frac{R_2}{R_1} I_2 \end{bmatrix}}_{\text{Nullspace Projection}} u \quad \Rightarrow \quad I_T \dot{w}_o + b_T w_o = 0 \quad \forall u \in \Re \tag{3.39}$$

3.4.9 Equivalence to a two-speed transmission

If M1 and M2 are identical motors and additionally if M1 only is used during high-speed mode and M2 only is used during high-force mode; then the behavior is equivalent to a single motor actuator with a two-speed transmission.

From an input-output point of view, if the commanded torque τ to the DSDM actuator is routed this way:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{cases} \begin{bmatrix} \tau \\ 0 \end{bmatrix} & \text{if } k = 1 \\ \\ \begin{bmatrix} 0 \\ \tau \end{bmatrix} & \text{if } k = 2 \end{cases} \tag{3.40}$$

Then the equation of motions for each operating mode, eq. (3.21) and eq. (3.22), are simplified to:

$$\left[I_o + R_1^2 I_1\right] \dot{w}_o + [b_o]\, w_o = [R_1]\, \tau \quad \text{if } k = 1 \tag{3.41}$$

$$\left[I_o + R_2^2 I_2\right] \dot{w}_o + [b_o]\, w_o = [R_2]\, \tau \quad \text{if } k = 2 \tag{3.42}$$

and they can be reduced the same input-output structure :

$$\left[I_o + R_k^2 I_k\right] \dot{w}_o + [b_o]\, w_o = [R_k]\, \tau \quad k \in 1, 2 \tag{3.43}$$

The same structure can also be found for the impulsive behavior in case of impacts:

$$w_0^+ = w_0^- + \frac{p_{ext}}{I_o + R_k^2 I_k} \quad k \in 1, 2 \tag{3.44}$$

Hence, if motors are identical ($I_1 = I_2$), then the only parameter changing in those equations is the reduction ratio R_k. Thus, from a high-level point of view, with the commanded torque routed as described by eq. (3.40), the behavior of a DSDM is equivalent to a single motor actuator with two gear-ratios.

3.5 Control algorithms

3.5.1 Architecture

Fig. 3-21 shows the proposed hierarchical control architecture for a DSDM actuator. As discussed in section 3.4, it is assumed that high-bandwidth current controllers are implemented for each motors. Moreover, it is also assumed that motor velocities w_1 and w_2 measurement are available to the DSDM controller. The actuator-level DSDM controller has the role of coordinating motors M1 and M2 to make the actuator gear-shift seamlessly between the two operating modes. The idea is to encapsulate DSDM actuators with simple control inputs: a desired motor torque τ_d and a desired gear-ratio k_d, like a semi-automatic transmission in a car. The high-level controller, analogous to the driver using again the car analogy, then only have to specify those two desired values and is released of the low-level management of the gear-shift process. With this architecture, a multi-DoF robotic system using multiple DSDM actuators could have a single centralized high-level controller and independent actuator-level controller for each actuators.

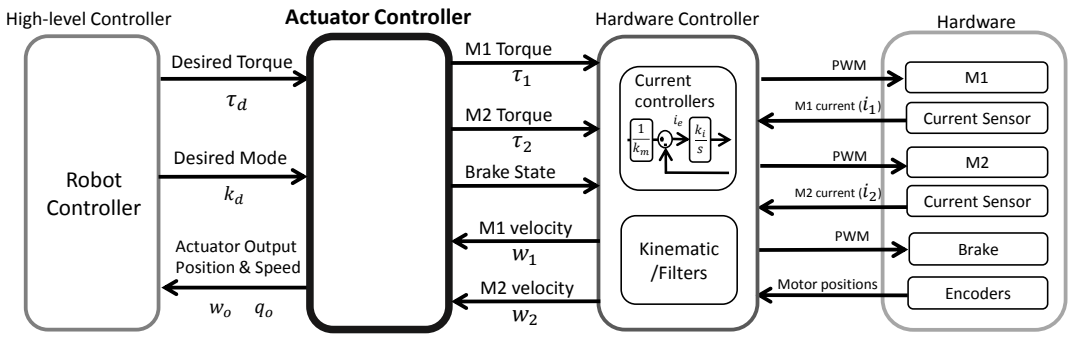


Figure 3-21: Control architecture of a DSDM controller

This section discusses the design of the actuator-level controller, focusing on gear-shift transitions. Chapter 4 will discuss extensively the design of centralized high-level controllers, and one implementation of the whole control system is presented in chapter 5.

3.5.2 State-machine

The proposed control scheme for the DSDM actuator-level controller has three discrete operating modes, described by a state machine illustrated at figure 3-22. There is one set of control laws for each discrete mode k (brake on/off), and also a special case of control laws (synchronization mode) for making sure M1 speed is zero before closing the brake, when a down-shift is desired. Fig. 3-23 shows the resulting closed-loop hybrid behavior.

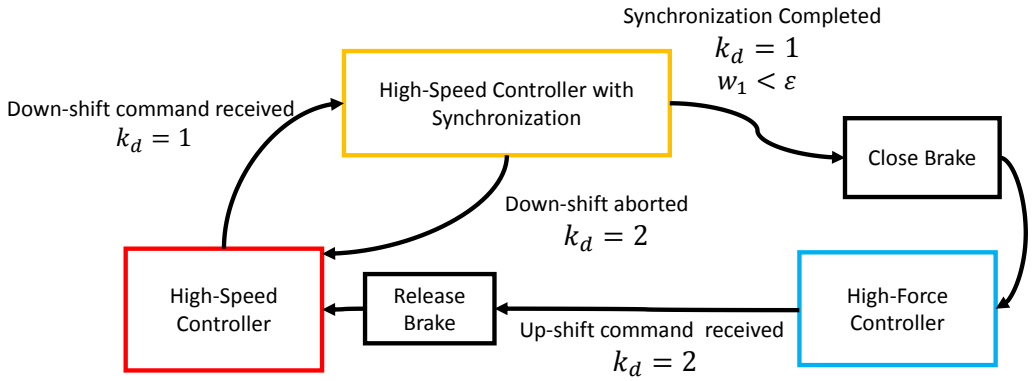


Figure 3-22: State machine of discrete control modes

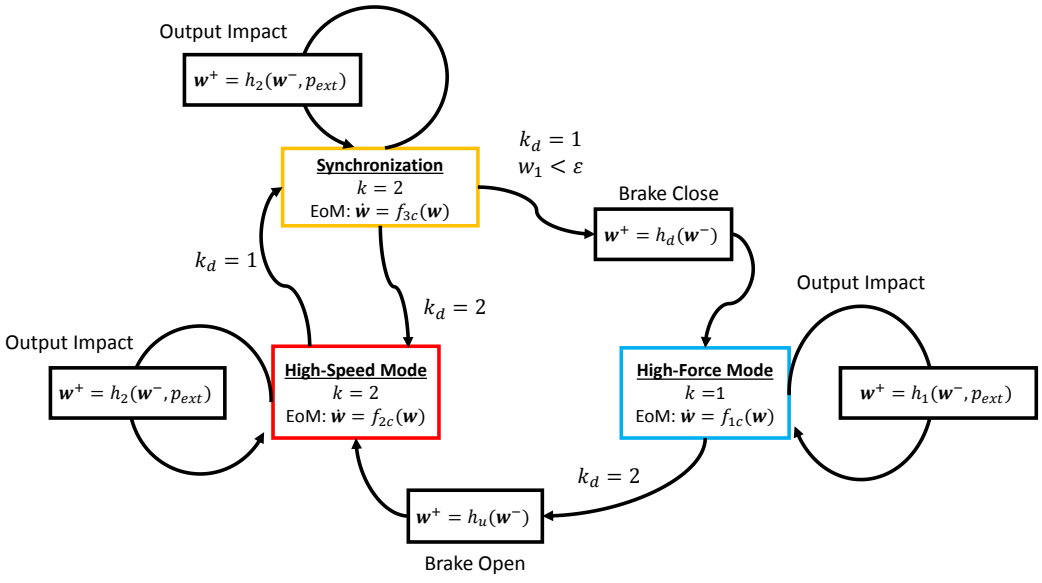


Figure 3-23: Hybrid behavior in closed-loop

3.5.3 High-force mode controller

During high-force mode, the role of the DSDM controller is simple: keeping M1 locked with the brake closed and directly transmits the desired torque τ_d to M2, see Fig. 3-24.

3.5.4 High-speed mode controller

During high-speed mode, the role of the DSDM controller is to: keep the brake open, transmit the desired torque τ_d to M1 and optionally track a secondary objective, see Fig. 3-25. Exploiting the nullspace of eq. (3.39), a secondary objective can be achieved without influencing the output behavior. Note that the nullspace projection vector, see eq.(3.39), depends only on parameters associated with the motors. Therefore, it is possible to project the secondary controller inputs on the output nullspace even if the output dynamic parameters that include the load inertia and damping are unknowns. The nullspace can be used for many possible objectives. In case the first motor is overloaded, for example, the second motor can reduce the load by projecting inputs through the null space, producing no effect upon the output, but changing the proportion of the two input commands. The secondary objective could also be used for maximizing efficiency, avoiding speed saturation of M1, etc. This chapter focus is on using the secondary objective to achieve fast and seamless gear-shifts.

3.5.5 Fast and seamless transitions (gearshifts)

This section addresses transition control between the two discrete operating modes.

Up-shift: $k : 1 \rightarrow 2$

In this case, the transition is simple because the system goes from 1 DoF to 2 DoF. As described by eq. (3.27) opening the brake at anytime does not lead to any undesirable impulsive behavior. As described by the state machine, as soon as an up-shift is commanded, the locking brake can be released, M1 is then instantaneously freed and the controller can immediately switch to the high-speed control mode.

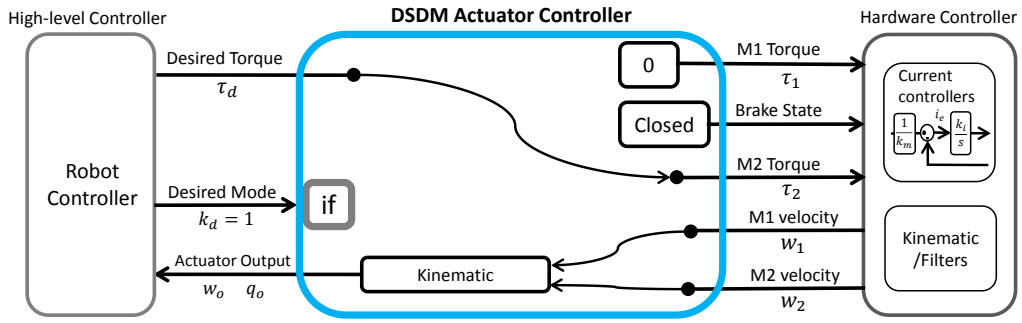


Figure 3-24: High-force mode: Controller

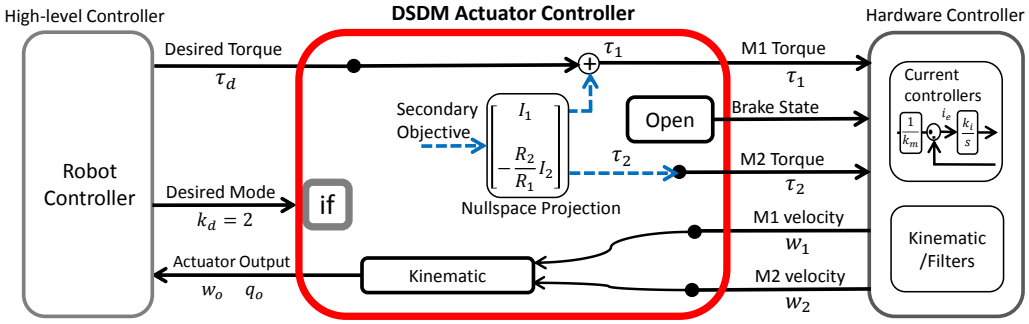


Figure 3-25: High-speed mode: Generic controller

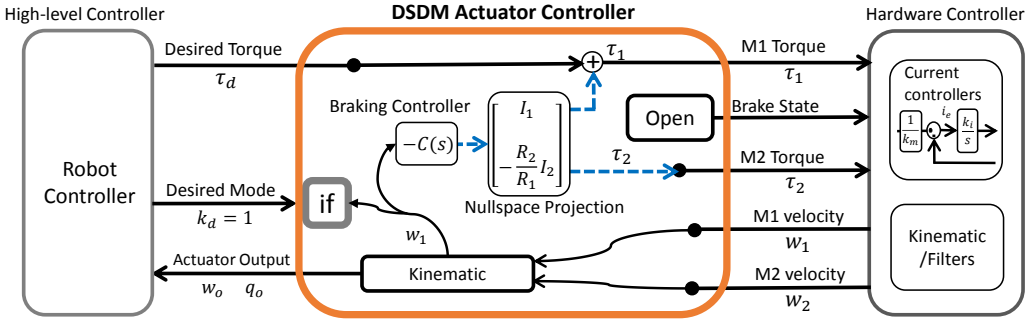


Figure 3-26: High-speed mode: Synchronization controller

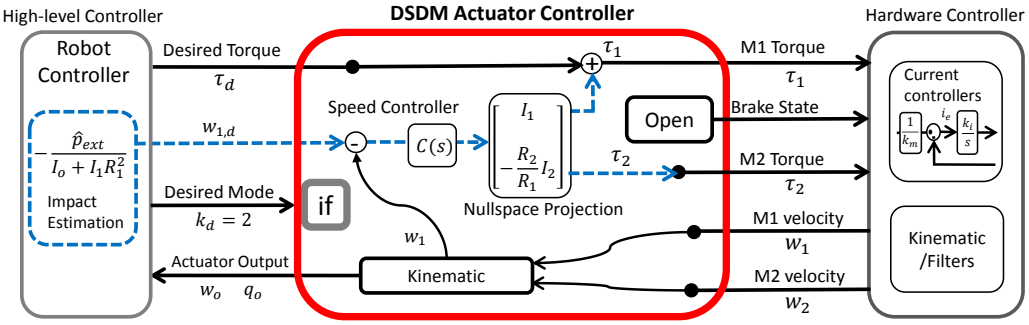


Figure 3-27: High-speed mode: Impact preparation controller

Down-shift: $k : 2 \rightarrow 1$

In this case, the transition is harder because the system goes from 2 DoF to 1 DoF, and some synchronization work is needed. As described by eq. (3.29), closing the brake will lead to an undesirable impulsive behavior if M1 velocity is not zero. M1 speed w_1 must thus be brought to zero so that the locking brake can be engaged smoothly without any impact. Hence, as illustrated at Fig. 3-22, when a down-shift is commanded, and intermediary synchronization control mode is activated, and the brake is only closed after M1 as reach zero velocity (a small value when implemented in practice).

3.5.6 Synchronization controller

First, a solution from a kinematic point-of-view is considered, then more flexible dynamic approach is proposed. Assuming the two motor velocities w_1 and w_2 can be treated as control inputs, then using the nullspace projection vector from eq.(3.37), the kinematic control law can be written as

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 \\ -R_2/R_1 \end{bmatrix}}_{\text{Nullspace Projection}} u_1 + \begin{bmatrix} 0 \\ R_2 \end{bmatrix} u_0 \tag{3.45}$$

leading to

$$w_o = u_0 \quad \text{with} \quad w_1 = u_1 \tag{3.46}$$

Therefore during transitions, using u_1 velocity w_1 can be driven to zero, while fully controlling the output velocity using u_0. The kinematic control law is valid only when high fidelity velocity controls are available and is only compatible with a situation where the high-level controller would specify a speed target instead of a torque target.

Alternatively a dynamic approach is more flexible. As illustrated at Fig. 3-26, while running the general high-speed controller, a braking law for w_1 can be used in

parallel as the secondary objective projected on the output nullspace:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \underbrace{\begin{bmatrix} I_1 \\ -\frac{R_2}{R_1} I_2 \end{bmatrix}}_{\text{Nullspace Projection}} \underbrace{-Cw_1}_{\text{Braking Law}} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tau_d \tag{3.47}$$

leading to independent output dynamics controlled with τ_d:

$$I_T \dot{w}_o + b_T w_o = R_1 \tau_d \tag{3.48}$$

and a w_1 closed-loop dynamic converging exponentially to zero if the synchronization gain C is large:

$$\dot{w}_1 = -Cw_1 + \underbrace{\left[\frac{R_1^2}{I_T} + \frac{R_1^2 I_o}{R_2^2 I_2 I_T}\right] \tau_d - \left[\frac{R_1 b_o}{I_T}\right] w_o}_{\text{Undesirable coupling from main loop}} \tag{3.49}$$

Hence, the output is not influenced by the braking law due to orthogonality, and is only controlled by the desired torque τ_d. On the other hand, w_1 is directly influenced by the braking law but also by the output speed and the desired output torque τ_d. Mathematically, it would be possible to also fully uncouple $\dot{w}_1$ equation, but the control law would not be practical in the scenario of $R_1 << R_2$ when considering torque and speed saturation. The other advantage of this approach over fully uncoupling the equations, is that the nullspace projection only requires motor-side parameters and is valid even when load-side dynamic is unknown.

Increasing the gain C will lead to faster braking of w_1, however motor torque will saturate if the gain is too large. A large synchronization gain C can still be used for faster braking at a cost of deviation from the desired torque τ_d. There is thus a trade-off in practice, passed the torque saturation point, between fast braking of w_1 for fast transition and fidelity to desired motor torque. Moreover, when considering the finite range of motor torques and speeds, there are situations where the synchronization controller won't converge. For instance, if the main loop ask for acceleration of the output with a large τ_d at high-speed while also asking for a down-shift, the situation

is not feasible. In those conflicting infeasible situations there is also a trade-off with prioritizing the main-loop with a small C but failing to synchronize M1, or prioritizing the down-shift command with a large C but failing to track the main loop command. To resolve this, the approach that was implemented is using a complex PI compensator ($C(s) = k_p + \frac{k_i}{s}$). Hence that way, at first the gain C is small and the synchronization controller doesn't affect the main loop. If the main loop behavior is conflicting with the down-shift, and synchronization doesn't converge at first, the gain C will slowly increase and eventually shift the priority to down-shifting. Eventually, when the C becomes very large and both motor torques would eventually saturate:

$$C \to \infty \quad \Rightarrow \quad \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \to \begin{bmatrix} -\tau_{max} \\ \tau_{max} \end{bmatrix} \tag{3.50}$$

which lead to the behavior:

$$I_T \dot{w}_1 = -\left(R_1 b_o\right) w_o - \left(R_1^2 + \frac{R_1^2 I_o}{R_2^2 I_2} + \frac{R_1 I_o}{R_2 I_2}\right) \tau_{max} \tag{3.51}$$

If w_1 is a positive value, then w_o is usually positive too and at worst a small negative value (due to kinematic constraints and motor speed saturations). Hence, the damping term is either negative, and thus helping the process, or negligibly small. Hence w_1 is guaranteed to continuously decrease and to eventually cross zero where the brake can be engaged. This is assuming no external forces overpower the the second term in eq.(3.51). Note that if the high-level controller is planning only feasible gear-shifts, than no down-shift in impossible situations should be commanded. Hence, in normal operation the down-shift controller should never reach this drastic synchronization mode. This mode is however useful when the DSDM controller receive commands directly from a human operator.

3.5.7 Preparation in the nullspace for faster down-shifts

Here a control scheme is proposed to prepare the actuator for a possible incoming down-shift command, using the secondary objective during regular high-speed oper-

ation, in a way that will make the synchronization process faster. The idea is that there is no need to wait to receive a down-shift signal to adjust velocities.

The condition for engaging the brake is :

$$w_1 = 0 \quad \Leftrightarrow \quad w_2 = R_2 w_o \tag{3.52}$$

Hence the ideal situation, to be prepared for a down-shift, is having M2 alone providing the output velocity. This is however only possible for low output velocity, due to motor velocity constraints. When possible to use this approach, the synchronization is done in advance and the down-shift operation is faster, since only limited by the time the brake takes to close. More generally, matching eq.(3.52) as close as possible will improve gear-shift time since synchronization error will be smaller to start with.

This idea can be also extended to a more complex impulsive behavior situations. As illustrated by Fig. 1-1, down-shifting can be particularly useful in contact situations happening in the context of manipulation or locomotion, where large forces must be bear rapidly after contact. For instance a robotic leg approaching the ground with high-speed mode, then making contact with the ground and transitioning to high-force mode to bear the weight of the robot. For this situation, where motor velocity will suddenly change right before the desired down-shift, the preparation condition takes this form:

$$w_1^+ = w_1^- + \Delta w_1 = 0 \quad \Leftrightarrow \quad w_2^+ = R_2 w_o^+ = w_2^- \tag{3.53}$$

Hence, if the impulsive behavior is expected and can be estimated, then the preparation scheme can also be used to improve gear-shift time during a contact. Two control schemes implementing this preparation are presented, first a kinematic approach and then a more flexible dynamic approach. Equation are derived for the case where an impact is expected, but the simpler control laws for non-impulsive situation can be extracted simply by setting the expected impulse to zero.

Rearranging the results of section 3.4.7 to express the two variables of interest (approaching speed w_0^- and M1 velocity post-impact w_1^+), as a function of controllable

inputs (from a kinematic point of view), lead to:

$$\begin{bmatrix} w_0^- \\ w_1^+ \end{bmatrix} = \begin{bmatrix} \frac{1}{R_1} & \frac{1}{R_2} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} w_1^- \\ w_2^- \end{bmatrix} + \begin{bmatrix} 0 \\ R_1 \frac{p_{ext}}{I_o + I_1 R_1^2} \end{bmatrix} \tag{3.54}$$

Hence it is possible to set both motor velocities so that the approaching output speed w_o^- and post-impact M1 velocity w_1^+ can be set arbitrarily and independently using the nullspace, by determining velocity set-points as:

$$\begin{bmatrix} w_1^- \\ w_2^- \end{bmatrix} = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} u_0 + \underbrace{\begin{bmatrix} 1 \\ -\frac{R_2}{R_1} \end{bmatrix}}_{\text{Nullspace projection}} \left(u_1 - R_1 \frac{p_{ext}}{I_o + I_1 R_1^2} \right) \tag{3.55}$$

leading to

$$w_o^- = u_0 \quad \text{and} \quad w_1^+ = u_1 \tag{3.56}$$

Therefore, if an impact is expected and the upcoming impulsive impact reaction $p_{ext} = \int \tau_{ext} dt$ can be computed, using this control law lead to independent control of the approaching speed of the actuator output w_o^-, which would be used for the primary objective (for instance making contact with the object at the right place and time), and resulting velocity of M1 post-impact w_1^+, which would be set to zero to allows for instantaneous down-shift.

Alternatively, as for the braking law used for synchronization during the down-shift process, this problem can be approached from a dynamic point of view. A desired velocity for M1 is defined as a function of the expected impulsive contact reaction:

$$w_{1,d} = -R_1 \frac{p_{ext}}{I_o + I_1 R_1^2} \tag{3.57}$$

Then, while using the regular high-speed controller to control the output, the secondary objective loop can be used to track M1 velocity set-point $w_{1,d}$ in the nullspace. This is analogous to the synchronization controller of eq. (3.47) but with a non-zero

set-point for M1:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \underbrace{\begin{bmatrix} I_1 \\ -\frac{R_2}{R_1} I_2 \end{bmatrix}}_{\text{Nullspace Projection}} \underbrace{C(w_{1,d} - w_1)}_{\text{Speed controller}} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \underbrace{\tau_d}_{\text{Main Loop}} \tag{3.58}$$

As discussed before, this will lead to independent output dynamics controlled with τ_d which would be used by the main output control loop:

$$I_T \dot{w}_o + b_T w_o = R_1 \tau_d \tag{3.59}$$

and a w_1 closed-loop dynamic converging exponentially to $w_{1,d}$ if the synchronization gain C is large:

$$\dot{w}_1 = C\left(w_{1,d} - w_1\right) + \underbrace{\left[\frac{R_1^2}{I_T} + \frac{R_1^2 I_o}{R_2^2 I_2 I_T}\right] \tau_d - \left[\frac{R_1 b_o}{I_T}\right] w_o}_{\text{Undesirable coupling from main loop}} \tag{3.60}$$

If the velocity controller is able to track the velocity set-point and the collision is perfectly predicted, then immediately after the impact, M1 velocity will be zero:

$$w_1^+ = w_1^- + \Delta w_1 \approx w_{1,d} + R_1 \frac{p_{ext}}{I_o + I_1 R_1^2} = 0 \tag{3.61}$$

Hence, it would be possible to engaged the brake and switch to high-force mode almost instantaneously. In practice, because of speed saturation it won't always be possible to perfectly track the desired speed $w_{1,d}$ in the nullspace, or to perfectly predict the contact impulse p_{ext}. However, by running this control scheme and bringing w_1^+ as close as possible to zero, gear-shift will nonetheless be achieved faster because synchronization error will be smaller at the start of the down-shift process. The preparation controller is illustrated at Fig. 3-27. Note that the expected impulse would need to be computed by an high-level controller aware of the general system situation. Section A.3 present techniques to compute impulsive contact forces with a model for multi-DoF robots, that could be used by the high-level controller.

3.6 Experimental results

This section presents experimental results with the wrist DSDM prototype of the DSDM-Arm. Information regarding the mechanical design and controller implementation is presented in Chapter 5. In experiments where the output is controlled in closed-loop for position or velocity tracking, the high-level control loop is a simple PID controller (recall Fig. 3-21). Set-points are commanded from a joystick controller.

3.6.1 DSDM dynamic behavior

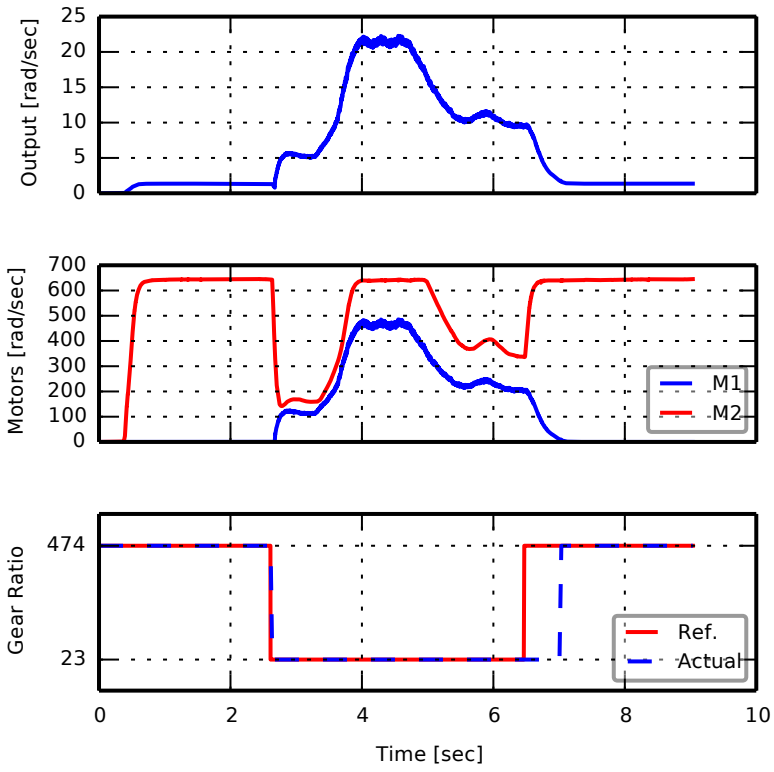


Figure 3-28: DSDM actuator with torque and mode commands set from a joystick

Fig. 3-28 gives an overview of how a DSDM actuator works. Before $t = 2.5$, the brake is engaged and only M2 is used to drive the output. When is brake is engaged, the DSDM is in high-force mode and the effective gear-ratio is 474. At $t = 2.5$, the brake is released and M1 start contributing to the motion. Releasing the brake will be

referred to as an up-shift. Between $t = 2.5$ and $t = 6.5$, the brake is open, both motor contribute to the output speed; the DSDM is in high-speed mode and the effective gear-ratio is 23. At between $t = 6.5$, a down-shift is manually commanded. Between $t = 6.5$ and $t = 7$, the synchronization controller then apply torques to brake M1 to zero speed and the brake is engaged at $t = 7$ when M1 speed is detected to be null. Note that here the down-shift was commanded from an infeasible high-velocity by the user driving the system, and the output speed had to first be slowed down to a speed reachable by high-force mode. This process of braking M1 and then engaging the brake will be referred as a down-shift.

Fig. 3-29 shows the DSDM tracking position set-points. This experiment illustrates that the high-force mode is slow and with a highly damped behavior, while the high-speed mode is fast and with an under-damped second order behavior, because of smaller dissipative forces in the motor.

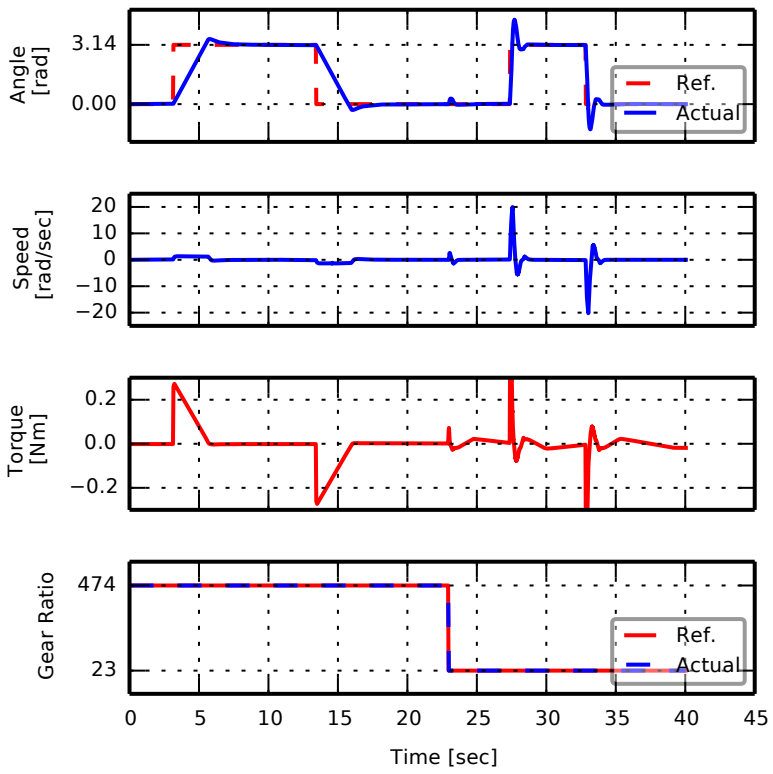


Figure 3-29: DSDM actuator tracking position commanded from a joystick

Fig. 3-30 shows the DSDM tracking speed set-points. Note that the noisy behavior during high-speed mode is only due to an implementation limitation. At low speed there is a quantization problem: there is a too small amounts of encoder ticks per control loop interval. This limited the controller gains for closing the loop in speed when using high-speed mode.

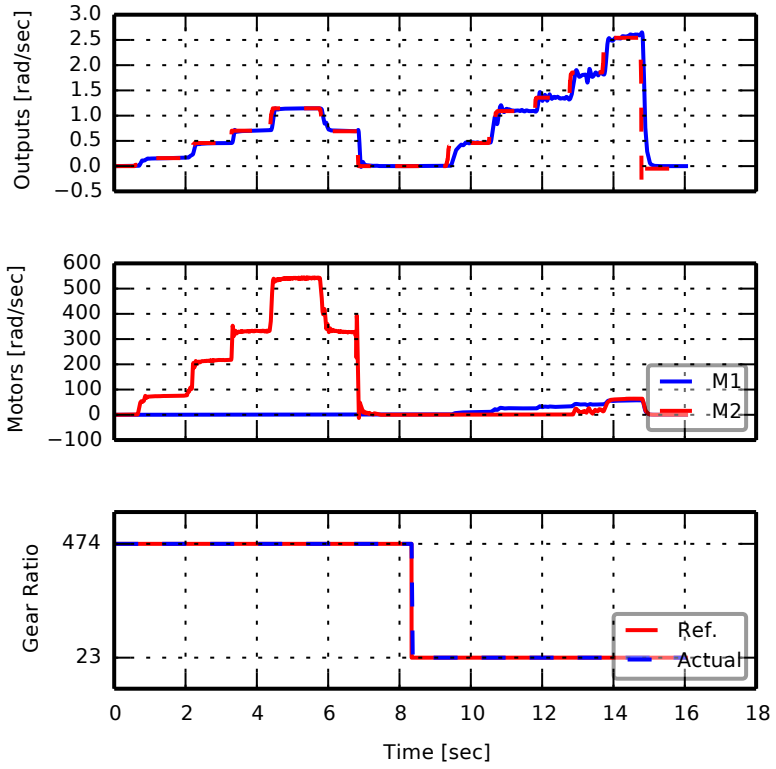


Figure 3-30: DSDM actuator tracking output velocity commanded from a joystick

3.6.2 Nullspace

Fig. 3-31 illustrates the internal DoF of DSDM actuators. Here the DSDM tracks a constant output speed while motor speeds are varied by projecting additional torques in the nullspace. This experiment demonstrates that the internal DoF can be controlled independently of the output using the proposed nullspace projection scheme.

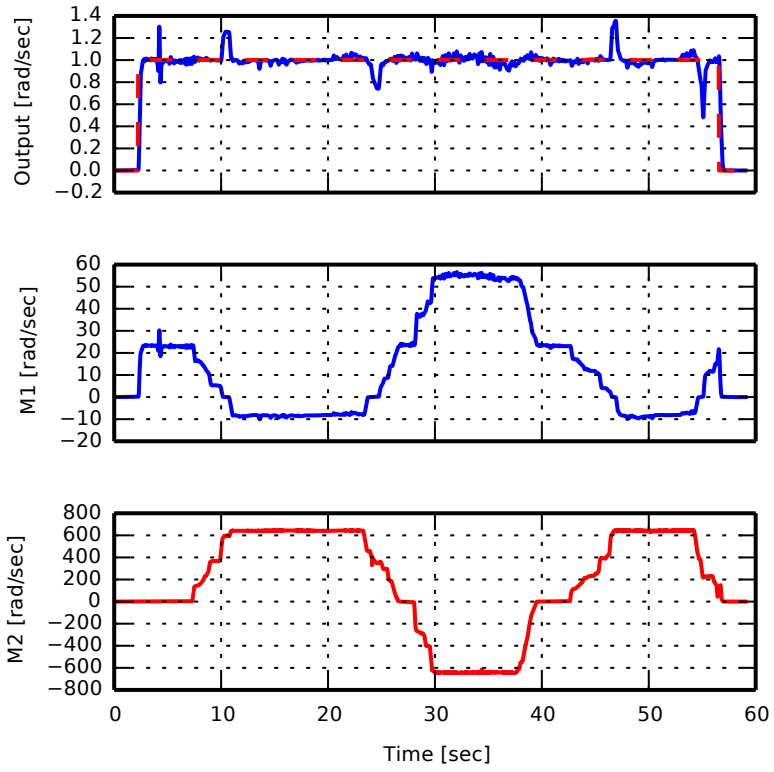


Figure 3-31: Experimental demonstration of independence of the output with respect to the secondary controller input, with the DSDM actuator tracking constant output velocity while sending manual commands in the nullspace

3.6.3 Seamless transitions

Constant Speed

Experiments here show gear-shifts execution while controlling the output to track a fixed speed. With the DSDM-Arm wrist tracking a 1 rad/s velocity, the average synchronization time for a down-shift, which is about 250 msec without preparation, is improved to 30 msec with the preparation scheme. Faster synchronization times of less than 20 msec were also achieved with another experimental set-up [14]. Hence while those experiments demonstrated that very fast gear-shifts in the range of 20 msec are possible, even faster synchronizations could be possible, since the bottleneck here is implementation factors, not fundamental aspects.

Fig. 3-33 illustrates the DSDM actuator maintaining a constant output speed while shifting back-and-forth between high-speed and high-force mode. Fig. 3-34

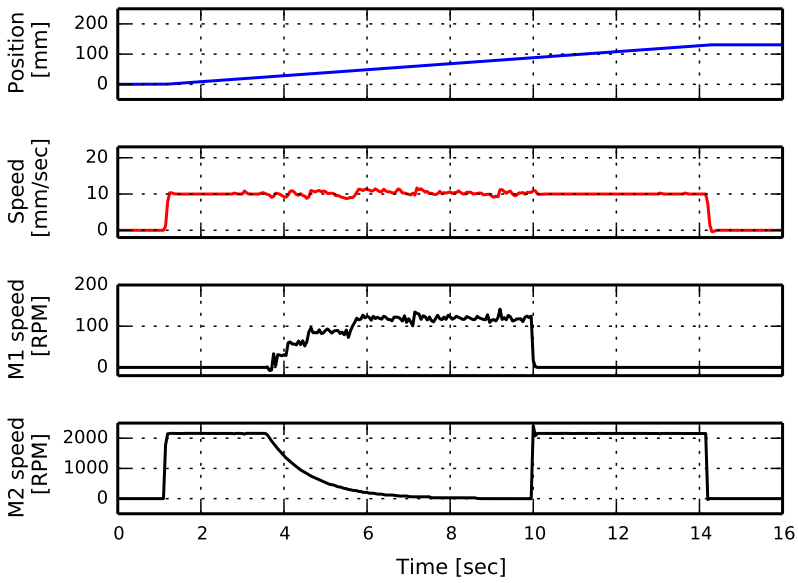


Figure 3-32: Smoother output velocity control during gear-shifts, with a DSDM controller implemented at high sampling rate on a FPGA

illustrates how the idea of preparation, by adjusting M2 velocity in the nullspace, improves the gear-shifting time. Note that the undesirable disturbances in the output velocity at Fig. 3-33 and Fig. 3-34, actually happens during up-shifts (no impacts, the brake is opening) and are caused by the transient response of the high-speed PID controller. This undesirable transient response was limited by implementation issues of the experimental set-up. Fig. 3-32 shows the same experiments when the DSDM controller is implemented on a FPGA at a higher sampling rate, which shows a much smoother output speed throughout the gear-shifting processes.

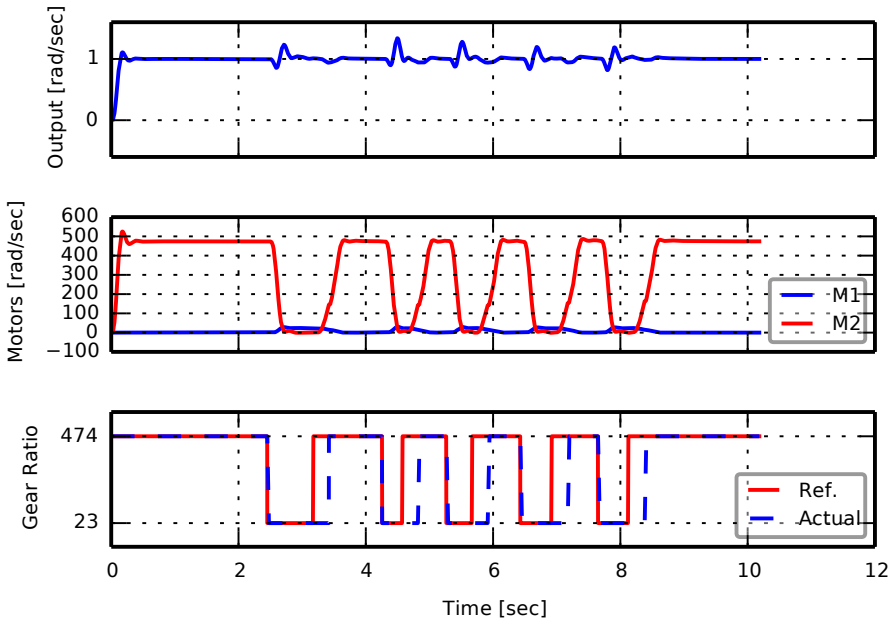


Figure 3-33: DSDM actuator tracking constant output velocity while shifting back and forth between high-speed mode and high-force mode (no preparation)

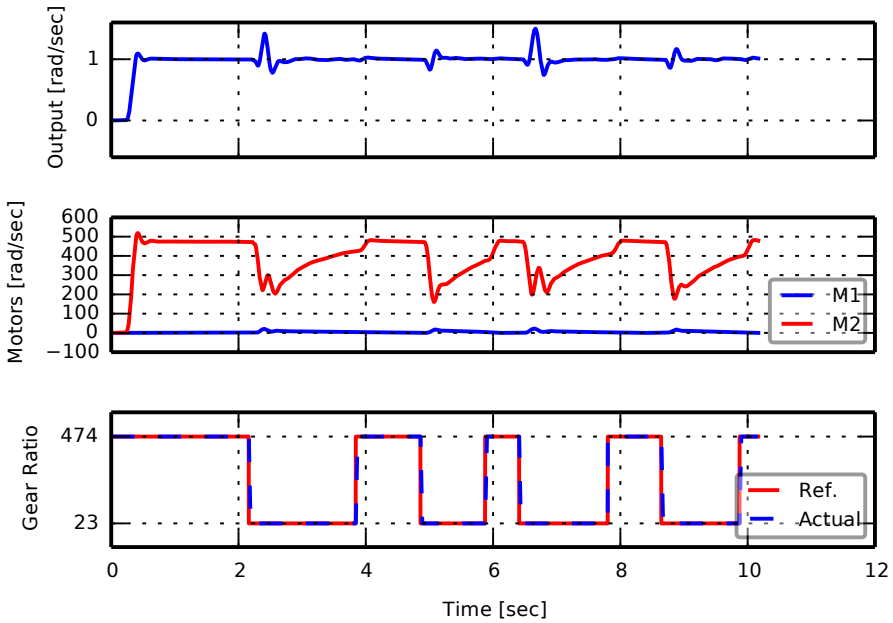


Figure 3-34: DSDM actuator tracking constant output velocity while shifting back and forth between high-speed mode and high-force mode (with preparation)

Contact with a Compliant Load

An experiment of the actuator making contact with a spring-like compliant load and making a smooth down-shift as soon as contact is detected is illustrated at Fig. 3-35. Results show that the controller is successful at down-shifting without any kick-back from the spring-like load (output velocity is always positive). In the experiment, the actuator output arrives with high-velocity and compresses the compliant load using its momentum, then the large gear-ratio is engaged seamlessly and larger forces are applied to compress the load further.

Impact with a Stiff Load

Fig. 3-36 illustrates an experiment where the DSDM is making a quick down-shift after impacting a stiff heavy object. Contact detection is done based on encoder measurements only, which is possible during high-speed mode because of the small reduction ratio leading to good actuator-transparency properties. The contact detection then trigger automatically the down-shift commands. Results show that the DSDM actuator is able to engage high-force mode within 25 msec of the impact, and seamlessly continue its motion pushing the heavy load with large torques.

Video of those down-shifts experiments, for the compliant load and the stiff load, are available at the following link: https://youtu.be/mUVrkDZr0kU.

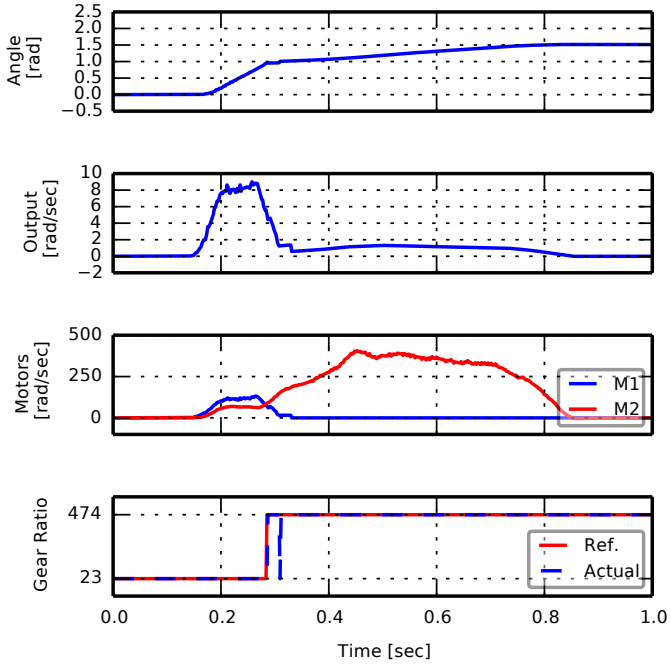


Figure 3-35: Fast downshift during a contact with a compliant load

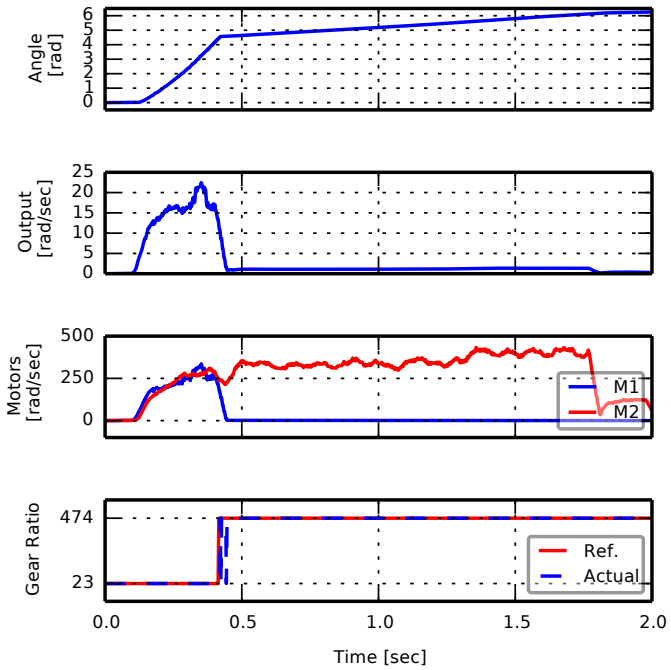


Figure 3-36: Fast downshift during a contact with a heavy stiff load

3.7 Summary

In this Chapter, an actuation technology capable of fast switching between two operating modes, equivalent to a small and a large reduction ratio, is presented. Advantages in term of weight, efficiency and reliability are discussed. Equations of motions for the hybrid behavior of the system are derived. A control scheme leveraging the internal DoF of the system to allow for fast and seamless transitions is proposed and experimentally validated. Prototypes DSDM actuators are found to meet performances requirements in real-world conditions.

3.8 Potential directions of further development

Here are a few possible axis of further development:

- Developing a product-like actuator system:
  - Embedded mechanical design using frameless motors;
  - Dedicated power-electronics and low-level controllers.
- Architecture ideas:
  - Multi-domain transducers, ex.: M1 $\rightarrow$ pneumatic and M2 $\rightarrow$ hydraulic;
  - Clutch system to reconfigure motors so both of them can always actively transmit torques (M1 is deadmass during high-force mode).
- Exploiting more thoroughly the nullspace:
  - Secondary controller maximizing efficiency based on an electrical model;
  - Formulating a multi-objective optimal control problem including speed limits, torque limits, efficiency and other criteria.
- Exploring the new possibilities for robotic tasks:
  - Using DSDM for legged locomotion;
  - Using DSDM for manipulation of large heavy objects.

Chapter 4

Optimal Dynamic Selection of Gear-ratios

"He will win who knows when to fight and when not to fight."

– Sun Tzu

The transmission gear-ratio that couples an actuator to a load has a significant effect on the behavior of the actuator-load system. With a large reduction ratio, the load-side dynamics has no significant effect because it is attenuated by the factor of the square of the gear-ratio. The net load acting on the actuator is mostly its own intrinsic load, including rotor inertia and friction. In contrast, with a small reduction ratio or a direct drive system [2], the behavior is usually dominated by the load-side dynamics which consist of highly non-linear inertial and gravitational forces for robotics manipulators. Sometime it can be advantageous to exploit the load-side dynamics: gravity may push the robot in a desired direction; the robot may coast with small dissipative torques induced at the actuator side; or the robot joints become backdrivable to comply to an external force. In other situations, however, it may be advantageous to isolate the actuators from the load-side dynamics and external disturbances: using a large gear-ratio to bear a large load or moving it slowly against gravity, for example.

This chapter aims to explore the potentials of actuator transmissions that can be switched dynamically, such as the technology presented in chapter 3, to either attenuate or leverage the natural dynamics of the system. Robots using lightweight VGA have the potential for achieving fast motions, high load-bearing, compliance and high-impedance, as required by diverse load conditions encountered by robotic systems. However, to truly exploit those salient features of VGA, control laws to select dynamically gear-ratios based on the current situation and task of the robot must be developed. Here in this chapter, feedback laws for robot control including gear-ratios selection are thus explored. The variable gear-ratios are used not merely for increasing maximum torque and speed, but also to significantly alter the dynamic properties, including load sensitivity, robustness, and backdrivability, advantageously given the situation.

In section 4.0.1, the principle of load leveraging and attenuation is delineated for a simple 1-DoF manipulator. Section 4.0.2 discuss related works and technical challenges and section 4.0.3 details the main original contributions. Section 4.2 introduces a formal mathematical representation and propose a dynamic model for robotic systems using VGA. Two different control approaches are explored, a model-based control laws synthesis in section 4.4, and a computational approach in section 4.6. The advantage of actively changing the gear-ratios are then illustrated with simulations in section 4.7, and with experiments using a robotic arm using custom-built variable gear-ratio actuators in section 4.8.

4.0.1 Illustration of the principle for a 1-DoF manipulator

Fig. 4-1 illustrates a simplified 1-DoF robotic manipulator where an electric motor is coupled to a pendulum through a gearbox with a gear-ratio R. As illustrated by phase portraits in Fig. 4-2, if R is small then the dynamic behavior of the system is dominated by the non-linear pendulum dynamics (Fig. 4-2a), but if R is very large, the behavior is dominated by the intrinsic inertia of the actuator, leading to the double-integrator behavior (Fig. 4-2b).

The vector fields of Fig. 4-2 illustrates the natural evolution of the system with

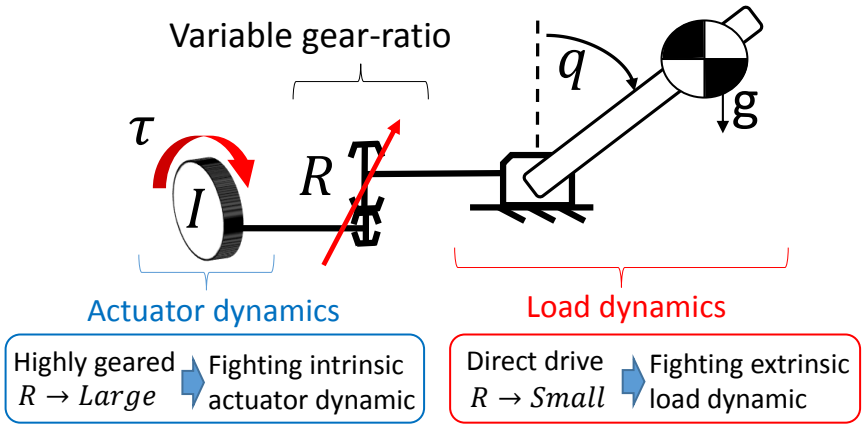
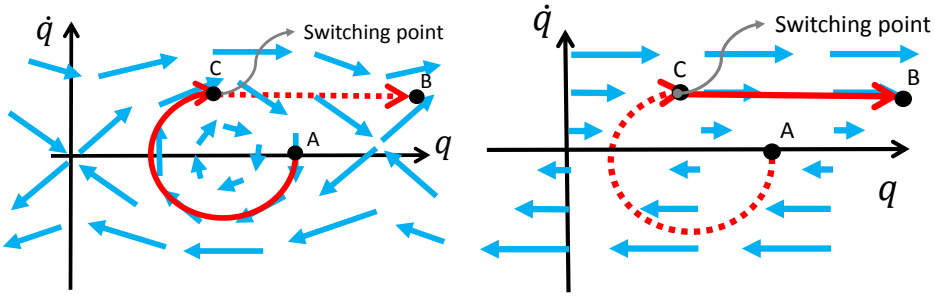


Figure 4-1: Effect of the gear ratio on the dynamics



(a) Reduction ratio R=1 (b) Reduction ratio R=10

Figure 4-2: Phase portraits illustrating the dynamical behavior

no actuator torques. Suppose that we want to move from state A to state B on the phase plane. Starting off state A with the gear-ratio of 1:1 brings the system along the curved trajectory shown in Fig. 4-2a. Switching the gear-ratio to 1:10 at state C will change the trajectory to the one in Fig. 4-2b, and bring the system to the destination state B. Note that no actuator torque is necessary for following this trajectory. A salient feature of actively changing the gear-ratio is that, the natural vector field behavior can be altered, in order to move in the desired direction with small torques.

4.0.2 Challenges and related works

This chapter investigates closed-loop control scheme for robots equipped with variable gear-ratios actuators. While variable transmissions have been studied extensively for

automobile power-trains, they have not yet been fully investigated in robotics, despite significant potential gains. Variable gear-ratio transmissions for electric motors have been proposed for legged locomotion [21], grasping robotic hands [55] , propulsion system [36] [40] and actuation systems [14] [20] [59]. Some of those works address the issue of how to change the gear-ratio, but there is no general approach to the high-level control of automatically selecting the right gear-ratios for non-linear, coupled multi-DoF robotic systems.

From the control perspective, automating the gear-ratios selection in a robotic context is a new and challenging problem. Gear-shifting is a very non-linear process (gear-ratios variables multiply other inputs and states in the equations of motion) and moreover the plant becomes a hybrid dynamical system if the usable gear-ratios are a set of discrete values. Hence most control engineering tools are not suited to tackle this problem. In simple scenarios, the gear-ratio selection can be based on simple principles. For instance, for a system running at a steady speed and load, the best gear-ratio can be selected based on efficiency maps. Alternatively, for rapid acceleration, the gear-ratios may be selected based on the actuator-load inertia matching [13] [9]. A robot, however, experiences diverse types of forces acting simultaneously. These include gravity, friction, and inertial forces as well as Coriolis and centrifugal forces. Hence, it is challenging to find a general control policy for selecting gear-ratios for the multitude of dynamically interacting actuators in the robotics context.

Trajectory planning In robotic, the generation of good reference trajectories is usually formulated as an optimization problem. Most optimal control techniques are based on either variational approaches or some form of gradient descent to find a trajectory that minimizes a cost function [4]. Hence those techniques cannot be used directly to optimize discrete variables. An interesting approach to get around this problem is to use the switching instants as optimization parameters instead [63] [39]. However, to use this approach a sequence of operating modes must be predefined first. Mixed-integer programming can be used to generate optimal open-loop trajectories of dynamical system with both continuous and discrete input variables [52]. For

instance, mixed-integer programming has been used to generate optimal open-loop trajectories for a car with both a continuous torque and a discrete gear-selection input [12]. Computation time was however in the order of hours for a 6 sec trajectory.

Sample-based planning scheme can also be used to find efficient trajectory [34]. These algorithms work generally better then optimization approaches when tackling highly-constrained system and when the goal is only to find a feasible trajectory and not necessary the optimal solution. The other advantage is that discrete control actions, like the selection of a gear-ratio in a discrete set, can be considered without complications since these algorithm works with discretized models.

Open-loop trajectories can be unstable and if the system deviates from the original plan, due to uncertainty, the optimized gear-ratios sequence might be completely unadapted after some time. For a robotic system to really leverage the advantage offered by multiple gear-ratios, it would be advantageous if gear-ratios are selected actively based on the actual conditions of the system. This chapter focuses on finding closed-loop policies for the gear-ratios selection.

Feedback control One possible approach for closing the loop would be simply to re-plan trajectory continuously online. However, the rate at which this would be possible for hybrid robotic systems would not be sufficient. Here we aim at having control policies that can react to a situation in a matter of milliseconds, for instance down-shifting as a robotic leg touch the ground. Most of the analytical results regarding feedback control of hybrid systems are for specific cases, for instance the optimal feedback laws for linear hybrid systems with linear constraints and a quadratic cost function have been shown to have a particular form [7]. One computational technique that generate feedback laws and that can be used for non-linear systems with any kind of constraints is dynamic programming [30]. Two disadvantages of the techniques are however that it is only computationally tractable for low-dimensional systems (so called curse of dimensionality) and also that the resulting feedback laws are in the form of a look-up table. This approach is investigated in section 4.6, which is an extension of work published by the author in [15].

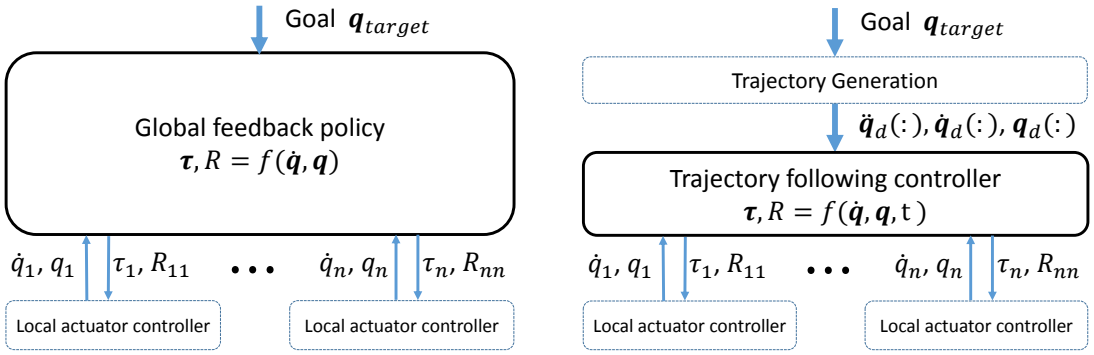
4.0.3 Original contributions

The main original contribution of this chapter, is an alternative model-based approach (section 4.4) with the advantage of scaling to high-dimensional robotics system, and easily applicable to trajectory tracking problems. The main idea was introduced by the author in [16] and this thesis chapter presents improved algorithms and new results. The main ideas to make tractable closed-loop control of this type of multi-DoF non-linear hybrid systems are 1) Using a simple modeling approach for variable transmissions that does not augment the number of state variables. 2) Exploiting the structure in the equations of motion when expressed in the inverse-dynamic form. 3) Using an outer-loop first specifying an instantaneously desired acceleration $\ddot{\boldsymbol{q}}_r$, similarly to feedback linearization, which fix locally the trajectory and makes possible computation of optimal instantaneous gear-ratios.

To the best knowledge of the author, this is the first exploration of closed-loop selection of gear-ratios for multi-DoF robotic systems. Many conceptual insight regarding gear-ratios selection when fighting different type of forces and unknown disturbances are explored. The treatment also encompass the very wide class of n-DoF mechanical system with EoM that can take the form of so-called manipulator equations (eq. (A.1), and is valid for many type of variable transmissions. The main restriction is that each joint of the system must be actuated, although it could also be extended to under-actuated systems when using a partial feedback linearization approach.

4.1 Control architecture

This chapter proposed control schemes to dynamically select the optimal gear-ratios online. The focus is on feedback policies that can react quickly to the states of the robot, and engage the appropriate gear-ratios with minimal delays (in the order of 20 msec). Two approaches are investigated and illustrated at Fig. 4-3.



(a) Dynamic programming approach (b) Model-based approach

Figure 4-3: Proposed control architectures

The architecture of both control approaches is designed so the control signal is a torque and a gear-ratio for each actuator. Note that in the case of discrete gear-ratios options, instead of transmitting the actual gear-ratios R, the control signal can be a label k indexing the discrete R_k options. Hence, it is assumed that low-level VGA controllers handle tracking the motor torque setpoint and the gear-shifting process. This is analogous to automating a car equipped with a semi-automatic transmission, and designing control policies replacing the driver for which the two control inputs are a throttle level a and shift-up/shift-down signal. The low-level VGA controllers would be specific to the type of actuator used in the system. For the particular case where the VGA are DSDM actuators, the low-level controllers would be the control laws described in Chapter 3. The architecture difference is that, for the dynamic programming approach given a goal, a global feedback policy is synthesized directly. However, for the model-based approach the high-level controller is separated into two levels. A feedback policy for trajectory tracking and an open-loop motion planning algorithm that generate a reference trajectory to reach the goal.

Trajectory generation Algorithm synthesizing a dynamic trajectory that meets performance requirements for reaching the target robot configuration starting at the actual robot configuration:

$$\ddot{\boldsymbol{q}}_d(:), \dot{\boldsymbol{q}}_d(:), \boldsymbol{q}_d(:) = f_{planner}(\boldsymbol{q}_{target}, \boldsymbol{q}) \tag{4.1}$$

Computation time is in the order of 1-10 sec depending the robot complexity. Hence, this step is done offline in advance, or alternatively in closed-loop by re-planning continuously but at a very low rate.

Low-level actuator controller Independent actuator controllers executing low-level hardware commands in response to a torque and a gear-ratio set-points. For the particular case of a DSDM actuator the controller computes:

$$\tau_1, \tau_2, b_{state} = f_{DSDM}(\tau_i, R_{ii}) \tag{4.2}$$

Trajectory following controller A function that compute torques and gear-ratios as a function of the robot actual states and the time:

$$\boldsymbol{\tau}, R = f_{ctl}(\dot{\boldsymbol{q}}, \boldsymbol{q}, t) \tag{4.3}$$

The function is synthesized based on a dynamic model of the robot and a desired trajectory. This function is to be executed in closed-loop at a high sampling-rate in the order of 1 kHz.

Global feedback policy A function that compute torques and gear-ratios as a function of the robot actual states. This function is to be executed online in closed-loop at a high sampling-rate in the order of 1 kHz. However synthesis of the policy require a learning phase that require multiple hours of computation.

$$\boldsymbol{\tau}, R = f_{ctl}(\dot{\boldsymbol{q}}, \boldsymbol{q}) \tag{4.4}$$

4.2 Modeling variable gear-ratio actuators

"With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." – John von Neumann

In this section, a simple approach is proposed for modeling robots using variable gear-ratio actuators. Variable transmissions are modeled as variable transformer elements, using the bond-graph terminology. This representation allows for a clear physical understanding of the effect of gear-ratios even for non-linear n-DoF systems. Furthermore, this modeling approach facilitates the implementation of a real-time optimization in the proposed controller. Limitations are discussed at section 4.2.3.

4.2.1 1-DoF system

First, a generic 1-DoF robot with a variable transmission is considered for simplicity. If the actuator's intrinsic resistive forces τ_I (rotor inertia and friction) are approximated to a linear quantity, the equations of motion (EoM) can be written as:

$$H\ddot{q} + D\dot{q} + g(q) = R\left[\tau - I\dot{w} - Bw\right] \tag{4.5}$$

$$\underbrace{\left[H\ddot{q} + D\dot{q} + g(q)\right]}_{\tau_E(\ddot{q},\dot{q},q)} = R\tau - R^2\underbrace{\left[I\ddot{q} + B\dot{q}\right]}_{\tau_I(\ddot{q},\dot{q})} \tag{4.6}$$

$$\tau = \frac{\tau_E(\ddot{q},\dot{q},q)}{R} + R\,\tau_I(\ddot{q},\dot{q}) \tag{4.7}$$

where the effect of the gear ratio can be seen clearly; increasing R attenuates the external dynamic terms τ_E but amplify the intrinsic actuator losses τ_I for a given trajectory. Variable gear-ratios can be modeled as variable transformer elements, using the bond-graph terminology. Fig. 4-4 shows a bond-graph representation.

4.2.2 Generalization to n-DoF manipulators

To generalize the above model to a n-DoF system with n actuators, the load-side dynamics is considered as a generic form of manipulator equations where each port

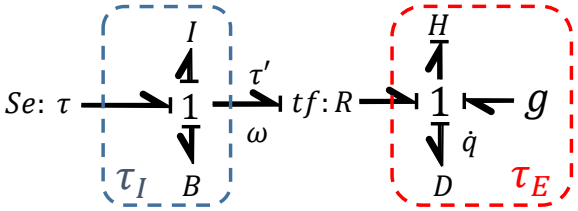


Figure 4-4: Model of a 1-DoF robot with a variable gear-ratio actuator

is connected to an independent actuator through a network of transformers. The network of transformers can be view as a type of coordinate transformation relating effort (force or torque) and flow (velocity or angular velocity) on the load side ($\boldsymbol{f}$,$\dot{\boldsymbol{q}}$) to those on the actuator output side ($\boldsymbol{\tau}'$,$\boldsymbol{w}$):

$$\boldsymbol{f} = R^T \boldsymbol{\tau}' \qquad R\dot{\boldsymbol{q}} = \boldsymbol{w} \tag{4.8}$$

where R is a n by n matrix consisting of all the transformer ratios.

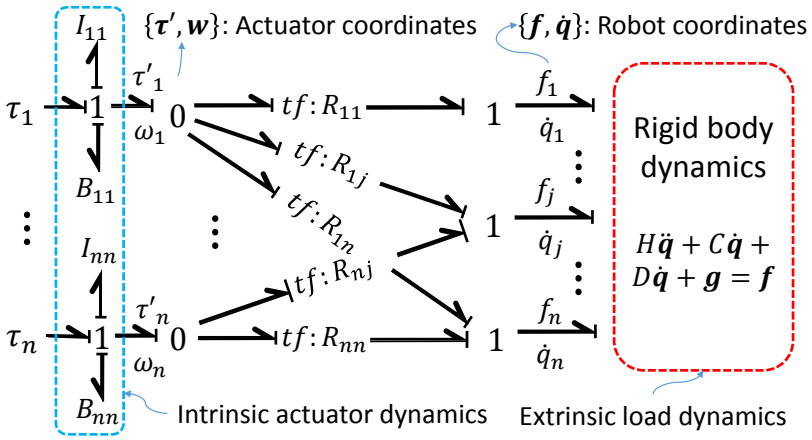


Figure 4-5: Model of a n-Dof robot with variable actuator-joint coupling

The EoM are then given by:

$$\underbrace{H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g}}_{\boldsymbol{\tau}_E(\ddot{\boldsymbol{q}},\dot{\boldsymbol{q}},\boldsymbol{q})} = R^T \underbrace{[\boldsymbol{\tau} - I\dot{\boldsymbol{w}} - B\boldsymbol{w}]}_{\boldsymbol{\tau}'} \tag{4.9}$$

A very wide range of mechanical systems and robots can be represented with this form. Note that, in the case of locomotion or manipulation where the robot interacts with the environment by physically contacting it, the dynamic model $\boldsymbol{\tau}_E$ must reflect

the contact conditions, either by computing contact forces using constraint equations (see section A.3.2) or by formulating $\boldsymbol{\tau}_E$ as a hybrid dynamical system.

In most practical cases, each actuator has its independent variable transmission and, thereby, the R matrix will be diagonal and each diagonal value can be selected independently. Assuming this situation, the EoM can be simplified to a form, similar to the scalar case, illustrating the effect of the gear ratios matrix R:

$$\boldsymbol{\tau} = R^{-1} \underbrace{\boldsymbol{\tau}_E(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}}, \boldsymbol{q})}_{\text{External load dynamics}} + R \underbrace{\boldsymbol{\tau}_I(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}})}_{\text{Intrinsic losses}} \tag{4.10}$$

$$\boldsymbol{\tau}_E \triangleq H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} \tag{4.11}$$

$$\boldsymbol{\tau}_I \triangleq I\ddot{\boldsymbol{q}} + B\dot{\boldsymbol{q}} \tag{4.12}$$

The derivation of this simplified form is available in the Appendix B.1.

4.2.3 Limitation of the simplified model

The main assumption of the proposed model is that gear-ratios are considered as independent control inputs, neglecting all the dynamics and delays associated with changing the gear-ratios. For a system with discrete gear-ratio, transient behaviors from one gear-ratio to another are thus neglected. Physically this implies that the kinetic energy of the system may be discontinuous at a gear-shift since the energy necessary for the transition is not considered. In the case of a car transmission for instance, this model would not keep track of the energy used for accelerating or braking the engine during the synchronization process. This model can be used if the gear-shift process is fast compared to the dynamics of the robots and if the energetic losses due to the gear-shift are negligibly small. Note that, for the DSDM-Arm presented in this thesis, this modeling assumption is supported by the characteristic of DSDM actuators, which can change gear-ratios quickly and seamlessly. For a robot equipped with continuously varying transmissions (CVT), this model would neglect forces associated with the rate of change of the gear-ratios $\dot{R}$, a type of quasi-steady assumption. In addition, this model also assumes that all motor rotors are in an

inertial reference frame, neglecting gyroscopic effects, which may be induced when the axes of motor rotors are rotated.

4.2.4 Uncertainty

Here, two observations are made regarding the effect of the gear-ratios on disturbances. First, considering modeling errors and external forces on the extrinsic side as unknown generalized forces $\boldsymbol{d}$, the EoM given by eq. (4.10) becomes:

$$\boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}}, \boldsymbol{q}) + R\boldsymbol{\tau}_I(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}}) + R^{-1} \underbrace{\boldsymbol{d}}_{\text{Disturbances}} \tag{4.13}$$

where it is assumed that the actuators are accurately modeled. Note that the effect of the disturbances is inversely proportional to the gear-ratios, and thereby attenuated when using large gear-ratios.

Second, large gear-ratios also decrease the sensitivity of the system to uncertainty. The error on accelerations computed with the inverse dynamic model, will be attenuated with large gear-ratios because of the larger actuator inertia reflected to the extrinsic side:

$$\ddot{\boldsymbol{q}}_e = \ddot{\boldsymbol{q}} - \ddot{\boldsymbol{q}}_r = \left[H + R^T I_a R\right]^{-1} \boldsymbol{d} \tag{4.14}$$

Hence, selecting large gear-ratios makes the system less sensitive to uncertainty on the extrinsic side.

4.2.5 Hybridness with discrete gear-ratios

When the variable transmissions have discrete configurations, the gear-ratios matrix can only take a set of discrete value:

$$R_k \in \{R_1, R_2, ..., R_l\} \tag{4.15}$$

where the variable k is a label indexing the different hybrid operating modes of the system, and l is the total number of discrete operating modes. The subscript k will be used to specify a variable specific to a discrete gear-ratios mode. The inverse dynamics equation takes the form:

$$\boldsymbol{\tau}_k = R_k^{-1}\boldsymbol{\tau}_E + R_k\boldsymbol{\tau}_I + R_k^{-1}\boldsymbol{d} \quad \forall\, k \tag{4.16}$$

Where $\boldsymbol{\tau}_k$ represent the torque $\boldsymbol{\tau}$ when using gear-ratios R_k. Note that the defined sum of extrinsic forces $\boldsymbol{\tau}_I$ and extrinsic forces $\boldsymbol{\tau}_E$, are not function of gear-ratios and are thus the same for every discrete mode k.

The equations of motions thus now have a hybrid nature. With the assumption that transitions are seamless however, states are continuous when gear-ratios changes, and the system is called a switched system [37]. The differential equations are discontinuous, but there is no instantaneous jumps in states. The forward dynamics equation takes the following form:

$$\ddot{q} = H_k^{-1}\left[R_k\tau - \boldsymbol{c}_k(\dot{\boldsymbol{q}}, \boldsymbol{q}) + \boldsymbol{d}\right] \quad \forall\, k \tag{4.17}$$

where

$$H_k = H + R_k^T I R_k \tag{4.18}$$

$$\boldsymbol{c}_k = \left[C(\dot{\boldsymbol{q}}, \boldsymbol{q}) + D + R_k^T B R_k\right]\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) \tag{4.19}$$

Note that here the discrete mode k is considered a control input, since it represent the gear-ratios selection.

4.3 Optimal gear-ratios along a trajectory

This section analyzes the optimal gear-ratios at each instant along a known trajectory. By looking backward at the situation, i.e. by evaluating necessary torques and other properties dependent on gear-ratios on a given trajectory, the situation is simplified. For any point on a given trajectory, accelerations $\ddot{\boldsymbol{q}}$, speeds $\dot{\boldsymbol{q}}$ and positions $\boldsymbol{q}$ are known, and then necessary torques $\boldsymbol{\tau}$ are only a function of the gear-ratios, which are the only remaining free parameters.

4.3.1 Selection criteria

The two main advantages of changing gear-ratios are 1) lowering the necessary torque to follow a trajectory, 2) modifying the effective impedance reflected on the environment and 3) avoiding rotor-speed limits.

Torque Optimization for reducing torque can be done by minimizing $\boldsymbol{\tau}^T\boldsymbol{\tau}$ at each point along the trajectory, over all possible gear-ratios. More generally a quadratic function $\boldsymbol{\tau}^T Q \boldsymbol{\tau}$ could be used to weight each actuator differently.

Impedance Optimization for reflected impedance can be done by minimizing the difference between desired task-space impedance and the actual one, which is directly affected by the matrix R. The end-point inertia matrix contains the gear ratios:

$$M = [J(\boldsymbol{q})^T]^{-1}\big[H(\boldsymbol{q}) + \underbrace{R^T I R}_{\text{Actuator contribution}}\big]J(\boldsymbol{q})^{-1} \tag{4.20}$$

The natural viscous damping reflected to the end-point is also influenced by gear-ratios:

$$V = [J(\boldsymbol{q})^T]^{-1}\big[D + \underbrace{R^T B R}_{\text{Actuator contribution}}\big]J(\boldsymbol{q})^{-1} \tag{4.21}$$

Constraints Another point of practical importance is that R should be constrained to values not leading to rotor velocities $\boldsymbol{w} = R\dot{\boldsymbol{q}}$ exceeding their maximum speed. This

is to avoid solutions with infeasible gear shifts, for example using the low gear at a high speed is impossible. Motor torque saturation could also be included by adding domain constraints on motor torques.

4.3.2 Optimization Formulation

The optimal gear-ratios are determined by minimizing the total actuator torques and, optionally, the difference in end-point impedance:

$$R^*(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}}, \boldsymbol{q}) = \underset{R}{\operatorname{argmin}} \left[\boldsymbol{\tau}^T \boldsymbol{\tau} + \alpha_1 \|M_d - M\| + \alpha_2 \|V_d - V\| \right] \tag{4.22}$$

$$\text{s.t} \quad R\dot{\boldsymbol{q}} \leq \boldsymbol{w}_{max} \tag{4.23}$$

where α_i are parameters to set the trade-off between minimizing motor torques and matching the desired impedance. Note that torques $\boldsymbol{\tau}$ can be substituted by the EoM in the inverse dynamic form (eq. (4.10)), and the minimized cost is a function of gear-ratios R, accelerations $\ddot{\boldsymbol{q}}$, speeds $\dot{\boldsymbol{q}}$ and positions $\boldsymbol{q}$.

4.3.3 Minimal Torque Solution

For a 1-DoF system, the optimal gear ratio leading to minimal torque, not considering any constraints, at a given instant on a trajectory is given by

$$R^* = \underset{R}{\operatorname{argmin}} \left[\tau^2 \right] = \sqrt{\left| \frac{\tau_E(\ddot{q}, \dot{q}, q)}{\tau_I(\ddot{q}, \dot{q})} \right|} \tag{4.24}$$

The derivation is available in the Appendix B.2.1.

Similarly for a multi-DoF system, if R is a diagonal matrix, the optimal gear-ratios can be obtained independently for each axis:

$$[R^*]_{ii} = \sqrt{\left| \frac{[\boldsymbol{\tau}_E(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}}, \boldsymbol{q})]_i}{[\boldsymbol{\tau}_I(\ddot{\boldsymbol{q}}, \dot{\boldsymbol{q}})]_i} \right|} \tag{4.25}$$

The derivation is available in the Appendix B.2.2.

Note that large gravitational forces or external disturbances, only present in $\boldsymbol{\tau}_E$, will usually lead to larger optimal gear-ratios, unless they cancel-out other forces in a way that makes $\boldsymbol{\tau}_E$ smaller. If inertial or viscous forces, present both in $\boldsymbol{\tau}_E$ and $\boldsymbol{\tau}_I$, dominate, then the optimal gear-ratios will be a compromise such that extrinsic and intrinsic forces are balanced, a form of impedance matching. The optimal gear ratio given by (4.25) includes both gravity, inertial and viscous effects as well as all other effects, hence it can be applied to any arbitrary dynamic situations.

4.3.4 Reduction to impedance matching

In simplified situation where there is only one type of force acting on the system, the general solution reduce to a case of impedance matching. For instance if only inertial forces are involves:

$$R^* = \sqrt{\left|\frac{H\ddot{q}}{I\ddot{q}}\right|} = \sqrt{\frac{H}{I}} \quad \Rightarrow \quad R^{*2} I = H \tag{4.26}$$

Hence the optimal gear-ratio is the one for which the external load inertia is equal to the reflected actuator inertia. Also, if only linear dissipative forces are involves:

$$R^* = \sqrt{\left|\frac{D\dot{q}}{B\dot{q}}\right|} = \sqrt{\frac{D}{B}} \quad \Rightarrow \quad R^{*2} B = D \tag{4.27}$$

Then the optimal gear-ratio is the one for which the external damping coefficient is equal to the effective actuator damping coefficient reflected to the output. Note that those optimal solutions are only valid locally. In general for robotic systems, H and D are state dependent.

4.3.5 Examples of optimal gear-ratios in simple scenarios

Here eq. (4.24) is applied to the robot in Fig. 4-1, an inverse pendulum with an actuators equipped with a CVT, in simple scenarios.

Acceleration from rest When the robot accelerates from rest with no viscous forces, the optimal gear ratio at the up-right position, where no gravity acts, is given by:

$$R^* = \sqrt{\left|\frac{H\ddot{q}}{I\ddot{q}}\right|} = \sqrt{\frac{H}{I}} \tag{4.28}$$

In this situation, the problem is reduced to impedance matching for two inertial loads. The optimal gear ratio minimizing the torque for a given acceleration is the one for which the load inertia and the motor reflected inertia are the same.

Supporting gravity without moving In the situation where the robot is not moving and fighting against gravity, then the optimal gear ratio is:

$$R^* = \sqrt{\left|\frac{\boldsymbol{g}}{0}\right|} \to \infty \tag{4.29}$$

In this static case, the largest possible gear-ratio is the optimal choice.

Coasting In the situation where a robot maintains a constant speed, assuming the output load is purely inertial and not dissipative ($D = 0$), but that there is friction in the motors:

$$R^* = \sqrt{\left|\frac{D\dot{q}}{B\dot{q}}\right|} = \sqrt{\frac{D}{B}} \to 0 \tag{4.30}$$

In this situation, to avoid dissipative motor forces it is best to have the smallest possible gear-ratio. In the limit, this corresponds to completely disconnecting the load from the actuator.

4.4 Model-based Controllers

"If you know the enemy and know yourself, you need not fear the result of a hundred battles." – Sun Tzu

In this section, control algorithms relying on a dynamic model of a robot and its load are proposed. Methodologies are proposed to synthesize feedback laws, for both the torques and gear-ratios input variables, to follow a trajectory with minimal effort.

4.4.1 R* Computed Torque

The proposed closed-loop controller, shown in Fig. 4-6, is based on the Computed Torque technique (see any robotic textbook such as [1]), but includes an optimization step to compute and select the optimal gear-ratios. As explored in section 3.6, when accelerations, speeds and positions are given, locally optimal gear-ratios can be computed. The idea is thus as follow, first compute a desired instantaneous acceleration $\ddot{\boldsymbol{q}}_r$ leading to guaranteed convergence to the desired trajectory:

$$\ddot{\boldsymbol{q}}_r = \ddot{\boldsymbol{q}}_d + K_D(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}) + K_P(\boldsymbol{q}_d - \boldsymbol{q}) \tag{4.31}$$

Then given the actual position $\boldsymbol{q}$, actual speed $\dot{\boldsymbol{q}}$ and desired instantaneous acceleration $\ddot{\boldsymbol{q}}_r$, extrinsic and intrinsic forces for this dynamic state are computed:

$$\boldsymbol{\tau}_E = H\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} \qquad\qquad \boldsymbol{\tau}_I = I\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + B\dot{\boldsymbol{q}} \tag{4.32}$$

Then the controller computes and executes the locally optimal gear-ratios R^*, as described previously for a known trajectory, and execute the corresponding motor torques:

$$R^* = \underset{R}{\operatorname{argmin}} \left[\boldsymbol{\tau}^T\boldsymbol{\tau}\right] \quad \text{where} \quad \boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E + R\boldsymbol{\tau}_I \tag{4.33}$$

$$\boldsymbol{\tau}^* = (R^*)^{-1}\boldsymbol{\tau}_E + R^*\boldsymbol{\tau}_I \tag{4.34}$$

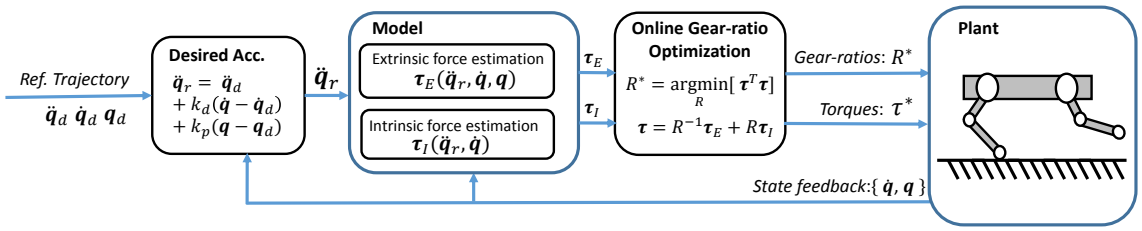


Figure 4-6: R* Computed Torque controller

Note that here a simple minimum torque square objective is illustrated for simplicity, but more complex cost functions can be used. As discussed in sec. 4.3, more complex objectives for optimizing gear-ratios are possible, for instance including motor constraints, desired impedance, etc.

The salient feature of the R* Computed Torque controller is that the optimal gear-ratios is selected based on state-feedback, i.e. even in situations not foreseen in the planner that generated the nominal trajectory. For instance, if a disturbance pushes the robot in a state where the robot faces a large gravitational force requiring a large gear-ratio, the controller will automatically select it. Similarly if facing a contact force, that is included in the model, the R* controller will automatically select the appropriate gear-ratio. Fig. 4-7 offer a graphical interpretation of the R* algorithm in the phase plane.

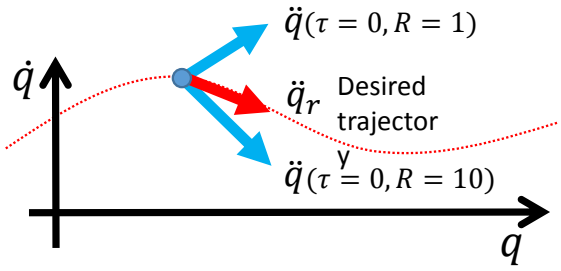


Figure 4-7: The R* algorithm can be interpreted graphically, as selecting the gear-ratios for which the natural acceleration vector is the closet to the desired instantaneous acceleration vector $\ddot{q}_r$ (after scaling the distance with the inertia), in order to minimize the necessary torques to apply on the system.

4.4.2 R* Sliding Mode Control

In general Computed Torque Control is susceptible to modeling uncertainties and disturbances. This section presents an approach based on sliding mode control [56], to improve robustness and guaranteeing performance despite uncertainty. Moreover, the presented control algorithm leverages the gear-ratios options to decrease sensitivity of the robot to disturbance when uncertainty is large. The control scheme is illustrated at Fig. 4-8.

First, the following intermediary variables are computed:

$$\boldsymbol{q}_e = \boldsymbol{q} - \boldsymbol{q}_d \qquad \dot{\boldsymbol{q}}_e = \dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_d \tag{4.35}$$

$$\boldsymbol{s} = \dot{\boldsymbol{q}}_e + \lambda \boldsymbol{q}_e \qquad \ddot{\boldsymbol{q}}_r = \ddot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}_e \tag{4.36}$$

As for the R* Computed Torque, intrinsic and extrinsic forces are then computed for this dynamic state:

$$\boldsymbol{\tau}_E = H\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} \qquad \boldsymbol{\tau}_I = I\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + B\dot{\boldsymbol{q}} \tag{4.37}$$

The instead of simply using the inverse dynamic equation to compute torques, a discontinuous gain is added:

$$\boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E(\ddot{\boldsymbol{q}}_r, \dot{\boldsymbol{q}}, \boldsymbol{q}) + R\boldsymbol{\tau}_I(\ddot{\boldsymbol{q}}_r, \dot{\boldsymbol{q}}) - R^{-1}Gsgn(\boldsymbol{s}) \tag{4.38}$$

Then the controller computes and executes the locally optimal gear-ratios R^*, minimizing the sliding mode torque that depends on selected gear-ratios, and execute the corresponding motor torques:

$$R^* = \underset{R}{\operatorname{argmin}} \left[\boldsymbol{\tau}^T\boldsymbol{\tau}\right] \quad \text{where} \quad \boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E + R\boldsymbol{\tau}_I - R^{-1}Gsgn(\boldsymbol{s}) \tag{4.39}$$

$$\boldsymbol{\tau}^* = (R^*)^{-1}\boldsymbol{\tau}_E + R^*\boldsymbol{\tau}_I - (R^*)^{-1}Gsgn(\boldsymbol{s}) \tag{4.40}$$

To guarantee convergence despite uncertainty, the discontinuous gain are set as a

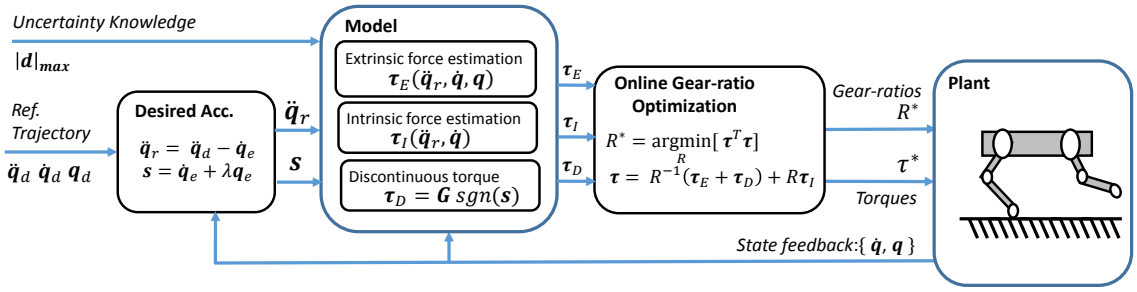


Figure 4-8: R* Sliding Mode controller

function of the bounds on uncertainty:

$$G = \left[H + R^T I R\right] K \tag{4.41}$$

$$K_{ii} > \max_{\boldsymbol{d}} \left| \left(\left[H + R^T I R\right]^{-1} \boldsymbol{d} \right)_i \right| + \eta \tag{4.42}$$

where $\boldsymbol{d}$ is an unknown generalized force vector, representing modeling uncertainty in the extrinsic dynamics and external disturbances, see section 4.2.4. Note that the discontinuous gain is a function of the state-dependent inertia matrix, bounds on disturbances and the selected gear-ratios. The interesting features of the R* Sliding Mode Controller is that with gear-ratios selected to minimize the sliding mode torque, then naturally, larger gear-ratios are selected in response to large uncertainty.

1-DoF system exemplified

To clarify the behavior of R* Sliding Mode, the simplified control laws for a 1-DoF system are analyzed. The feedback law for torque in 1-DoF is reduced to

$$\tau = \frac{\tau_E(\ddot{q}_r, \dot{q}, q)}{R} + R\tau_I(\ddot{q}_r, \dot{q}) - \frac{G sgn(s)}{R} \tag{4.43}$$

and the discontinuous gain reduced to

$$K = \frac{d_{max}}{H + R^T I R} + \eta \tag{4.44}$$

$$G = \left[H + R^T I R\right] K = d_{max} + \left[H + R^T I R\right] \eta \tag{4.45}$$

Rearranging the torque law gives:

$$\tau = \frac{\tau_E(\overbrace{\ddot{q}_r - \eta sgn(s)}^{\ddot{q}_a}, \dot{q}, q) - d_{max} sgn(s)}{R} + R\tau_I(\overbrace{\ddot{q}_r - \eta sgn(s)}^{\ddot{q}_a}, \dot{q}) \tag{4.46}$$

Minimizing the torque computed with this sliding mode control law over possible R values, also have an enlightening analytical solution:

$$R^* = \underset{R}{\text{argmin}} \left[\tau^2\right] = \sqrt{\left| \frac{\tau_E(\ddot{q}_a, \dot{q}, q) + |d|_{max}\, sgn(s)}{\tau_I(\ddot{q}_a, \dot{q})} \right|} \tag{4.47}$$

If no disturbance is expected ($|d|_{max} = 0$) then the solution is, as before, a compromise between extrinsic and intrinsic forces. However, knowledge of uncertainty in the form of disturbances bound modifies the optimal gear-ratios solution. When large disturbances are expected ($|d|_{max}$ is large) the solution is biased toward larger gear-ratios, which is consistent with the sensitivity analysis that concluded that larger gear-ratios attenuate the effect of disturbances. All in all, minimizing torque computed with the sliding mode law, is a way to naturally improve the decision regarding the best gear-ratios, in function of some knowledge of the expected uncertainty.

4.4.3 Adaptation

If the uncertainty is structured as unknown model parameters in the extrinsic dynamics, the term represented by $\boldsymbol{\tau}_E$, then traditional adaptation schemes can be used for estimating the unknown parameters. Then, if adaptation converges to the correct computed torque, then the computed best gear-ratios will also converge to the true optimal gear-ratios:

$$\hat{\boldsymbol{\tau}}_E \to \boldsymbol{\tau}_E \quad \Rightarrow \quad \hat{R}^* \to R^* \tag{4.48}$$

If used in conjunction with the R* Computed Torque controller, decisions regarding the optimal gear-ratios would improve as adaptation converges. Note that correct gear-ratios decision depends on the correct estimation of extrinsic torque, not model

parameters explicitly, which are harder to estimate since requiring condition related to sufficient excitation.

As an example, for a 1-DoF robot with the following EoM:

$$\left[H + R^2 I\right] \ddot{q} + \left[R^2 B\right] \dot{q} + \left[mg\sin(q)\right] = R\tau \tag{4.49}$$

the regression to identify unknown extrinsic parameters H and m, could be built this way:

$$\overbrace{\underbrace{\begin{bmatrix} \frac{\ddot{q}}{R} & \frac{g\sin(q)}{R} \end{bmatrix}}_{\boldsymbol{\phi}} \underbrace{\begin{bmatrix} H \\ m \end{bmatrix}}_{\boldsymbol{\theta}}}^{\tau_E} = \underbrace{\tau - \overbrace{RI\ddot{q} - RB\dot{q}}^{\tau_I}}_{y} \tag{4.50}$$

where $\boldsymbol{\theta}$ is a vector of unknown parameters, $\boldsymbol{\phi}$ is a known regressor vector (the controller is always aware of the selected gear-ratio R) and y is a known scalar output.

Adaption on both the extrinsic and intrinsic dynamic parameters could also be conducted, with a regression constructed this way:

$$\underbrace{\begin{bmatrix} \frac{\ddot{q}}{R} & R\ddot{q} & R\dot{q} & \frac{g\sin(q)}{R} \end{bmatrix}}_{\boldsymbol{\phi}} \underbrace{\begin{bmatrix} H \\ I \\ B \\ m \end{bmatrix}}_{\boldsymbol{\theta}} = \underbrace{\tau}_{y} \tag{4.51}$$

However, care would need to be used since the first two terms in the regression vector would be fully correlated if data using a single gear-ratio is used, which makes the solution ambiguous in term of possible H and I parameters. If data with multiple different gear-ratios is used, then all terms could be independently identified.

4.4.4 Generalization to more complex models

Many modeling assumption, for instance leading to extrinsic vs. intrinsic force separation, were assumed to hold during the control algorithm presentations mainly because it gives many physical insight. However, this restricted class of model is not necessary to implement the proposed algorithms, the minimum needed is to have a model of the inverse dynamic, which could include any non-linearity, in the form:

$$\boldsymbol{\tau} = f(\ddot{\boldsymbol{q}}_r, \dot{\boldsymbol{q}}, \boldsymbol{q}, R) \tag{4.52}$$

to compute necessary torque $\boldsymbol{\tau}$ to achieve a specified acceleration $\ddot{\boldsymbol{q}}_r$, given actual states $(\dot{\boldsymbol{q}}, \boldsymbol{q})$ and selected gear-ratios R.

4.4.5 Closed-loop selection of discrete gear-ratios

So-far the proposed control schemes made no assumption regarding the different gear-ratios options, and analytical solutions assumed that gear-ratio have continuous domains. However, many variable transmission mechanisms, such as the DSDM actuator technology proposed in this thesis, offer only a discrete set of possible values. In that situation, the optimization step in the proposed control algorithms would then be a combinatorial optimization problem. However, if the number of options l is reasonably small, then every possible options can be computed quickly to find the optimal discrete option:

$$k^* = \underset{k}{\operatorname{argmin}} \left[\boldsymbol{\tau}_1^T \boldsymbol{\tau}_1, ..., \boldsymbol{\tau}_k^T \boldsymbol{\tau}_k, ..., \boldsymbol{\tau}_l^T \boldsymbol{\tau}_l \right] \tag{4.53}$$

In practice this brute force approach is usually feasible. For instance, for the robot presented in this thesis, there is 3 actuators each with 2 gear-ratios options, leading to only $l = 2^3 = 8$ possible matrix R.

Point-by-point

In theory if the controlled robotic system would behave exactly like the proposed model, then selecting optimal gear-ratios at each time steps would be the optimal behavior. However, because the control effort associated with gear-shifts is neglected in the model and the basic gear-selection scheme does not penalize mode transitions, using the proposed controller can lead to rapid switching between gear-ratios (chattering) in certain situations. In practice, this is not desirable since for most type of variable transmissions, since changing the gear-ratio: 1) is not really instantaneous 2) the shifting process would require some effort/energy 3) fast switching can lead to mechanical wear and also excite un-modeled vibration modes.

Hysteresis

To avoid undesirable rapid switching behaviors, hysteresis can be added to the gear-selection logic. Instead of simply executing the optimal gear-ratios command at each time step, a logical step is added, which only allows mode change if a minimal amount of time Δt as elapsed since the last mode change. This thus directly guaranteed a minimal period between gear-shifts. However, this technique is clearly sub-optimal in certain situations. For instance, imagine a system with two modes, on a trajectory where $k = 1$ is optimal for all time except for the interval $t \in [1, 1.1]$. If a hysteresis of $\Delta t = 1$ is used in the controller, then $k = 2$ will be selected for the interval $t \in [1, 2]$. Hence, not gear-shifting at all would have been better (sub-optimal during 0.1 sec) compared to shifting with hysteresis (sub-optimal during 0.9 sec), from the whole trajectory point of view.

Minimax optimization for sliding mode

When optimizing the torque computed with the sliding mode, chattering can be very severe as components of $sgn(\boldsymbol{s})$ can oscillate rapidly between values of ± 1. One approach to alleviate this is to reformulate the optimization to treat variables $sgn(\boldsymbol{s})$

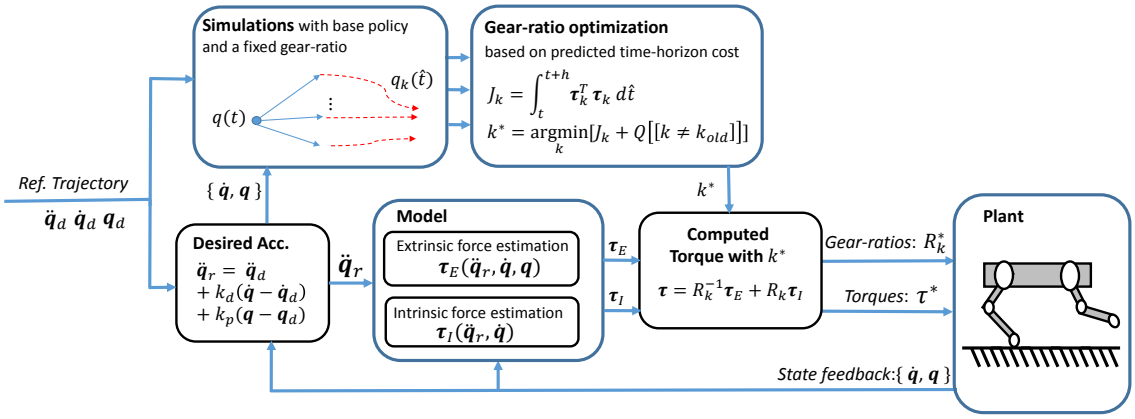


Figure 4-9: Rollout with Computed Torque Control as base policy

as random disturbances and optimize for the worst-case scenario:

$$R^* = \underset{R}{\text{argmin}} \max_{sgn(\boldsymbol{s})} \left[\boldsymbol{\tau}^T \boldsymbol{\tau}\right] \quad \text{where} \quad \boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E + R\boldsymbol{\tau}_I - R^{-1}Gsgn(\boldsymbol{s}) \tag{4.54}$$

The result of this optimization is independent from the position of states with respect to the switching surface $s = 0$, and would remove this source of rapid change of optimal gear-ratios solutions. Also, the sliding mode controller could be implemented with smoothing techniques for the discontinuous torque, many techniques exist [56] [46]. This would also alleviate one source of chattering for the gear-ratios selection.

4.4.6 Rollout gear-ratios selection

A better approach to the problem of avoiding fast gear-shifting, is to associate a one-time cost with transitions and optimize over a time-horizon. This way a trade-off between changing gear-ratios quickly to minimize torques and minimizing the number of gear-shifts can be formalized. The proposed approach to implement this idea in a closed-loop control scheme, is to use predictive simulations over a receding time-horizon, analogous to the model predictive control (MPC) approach. Predicted trajectories are computed by simulating the robotic system under the control of a base policy that consist in 1) keep the gear-ratios fixed 2) motor torques controlled with the computed torque or sliding mode feedback law. The approach is illustrated

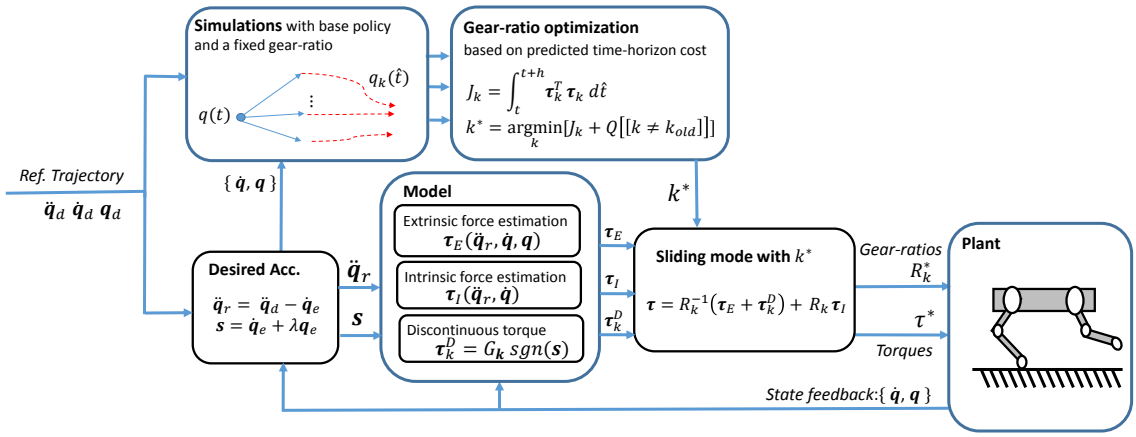


Figure 4-10: Rollout with Sliding Mode Control as base policy

used in conjunction with a Computed Torque at Fig. 4-9 and sliding mode at Fig. 4-10.

An integral cost J_k for each of those l simulated trajectories with fixed gear-ratios is computed:

$$J_k = \int_t^{t+h} \boldsymbol{\tau}_k^T \boldsymbol{\tau}_k d\hat{t} \tag{4.55}$$

where t is actual real time, h is the horizon, $\hat{t}$ is the simulation time and $\boldsymbol{\tau}_k$ are the torques computed with the base policy in the simulations. Once those simulations and integrals are computed, the gear-ratios are selected by conducting the following optimization:

$$k^* = \underset{k}{\text{argmin}} \, [J_k + Q[[k \neq k_{last}]]] \tag{4.56}$$

where Q is the cost penalty for gear-shifting. Hence, optimal gear-ratios, computed at time t, are the optimal ones considering an integral cost over the future horizon h. Note that state errors are not penalized in the integral cost, because the base policy for motor torques already enforces trajectory tracking. The scheme is illustrated at Fig. 4-11.

This scheme is closely related to the Rollout control approach [3], and it will thus be refer to as the Rollout gear-ratios selection. However here the optimization is

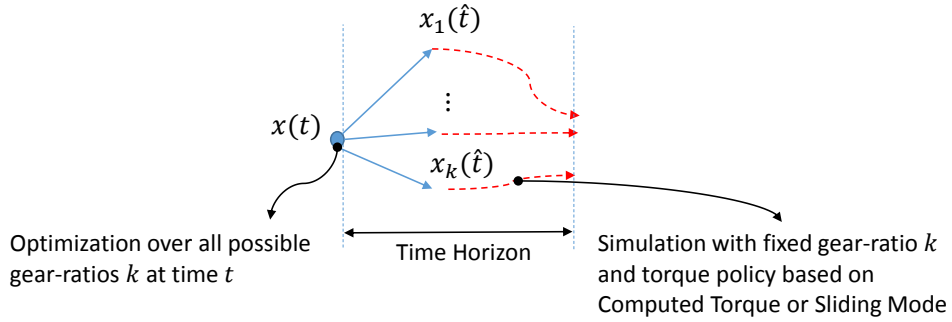


Figure 4-11: Rollout gear selection

conducted only over the gear-ratios options, not all possible control actions at time t like in the original technique. The two reasons for not implementing a full scale optimization including also many possible torque inputs at time t are: 1) computational limitations and 2) by applying the base policy torques at time t, convergence to the desired trajectory can be guaranteed.

Advantageous features of the Rollout gear-selection scheme are 1) filtering-out fast un-desirable gear-shifts 2) commanding gear-shifts in advance, which can compensate for gear-shift delay in the physical system (situation just entering the horizon at time $t+h$ influence the gear-ratios selection at time t).

Extension from Rollout to Model Predictive Control (MPC)

The proposed control scheme requires simulating l trajectory at each time steps, which is usually tractable computationally. However, this is sub-optimal in the sense that gear-ratios are fixed on each trajectory. With more computational power, additional options of more complex sequences of discrete gear-ratios could be included. Each option of a predetermined sequence of gear-ratios could be seen as a form of motion primitive, a proposed approach to simplify control/planning when the number of possible action is too large to fully explore [18]. This would also connect with the idea of family of modes proposed to simplify MPC control of hybrid systems [22]. The proposed Rollout gear-selection scheme, can thus also be seen as optimizing over l motion primitives consisting of using fixed gear-ratios for the next h seconds.

Furthermore, if the optimization would also be conducted over all possible motor

torques, instead of constraining them to always follow a given feedback law, then the control scheme would correspond to full scale MPC. However, MPC is in general very hard to implement at satisfactory fast rate for non-linear multi-DOF robotic systems, especially when the system is hybrid [22]. All in all, depending on the complexity of the controlled robotic system and the available computational power, the proposed control scheme can be adapted from a very easy-to-compute point-by-point optimization with hysteresis, to more optimal but computationally-heavy predictive schemes with a variety of level of complexity.

4.4.7 Stability

Here stability properties of the proposed control laws are discussed. The main interesting conclusion is that complex gear-selection schemes, such as Rollout, can be implemented in conjunction with the proposed base feedback laws for torques (computed torque or sliding mode), without affecting the stability results.

R* Computed Torque Convergence to the desired trajectory, with the R* Computed Torque controller, is guaranteed assuming the model used by the controller is exact. Interestingly this results is valid for any arbitrary sequence of gear-ratios as long as the feedback law computing the continuous torques $\boldsymbol{\tau}$ is aware of the discrete operating mode k, see details in sec. B.3.1. Hence, any gear-ratios selection scheme can be used, without compromising the convergence to the desired trajectory, including hysteresis, Rollout optimization, etc.

R* Sliding Mode Guaranteed convergence to the desired trajectory, with the R* Sliding Mode controller, can be extended to situation where the uncertainty can be bounded, assuming gains G are chosen accordingly, see details in sec. B.3.2. Stability is also guaranteed for any sequence of discrete gear-ratios k and thus would not be affected by the closed-loop gear-ratios selection scheme.

4.4.8 Chattering and high-frequency switching

This section discusses results regarding the chattering behavior when using the proposed control schemes.

Decision boundaries

Chattering (quickly selecting back-and-forth) between two discrete mode i and j can occur when the system stay in a dynamic state for which optimized cost is equal for two or more gear-ratios options. For instance with the quadratic torque criterion when:

$$\boldsymbol{\tau}_i^T \boldsymbol{\tau}_i = \boldsymbol{\tau}_j^T \boldsymbol{\tau}_j \tag{4.57}$$

If the policy for motor torque $\boldsymbol{\tau}_k(\boldsymbol{x})$ is substituted in the cost equality equation, the result is a decision boundary in the state-space, in the form:

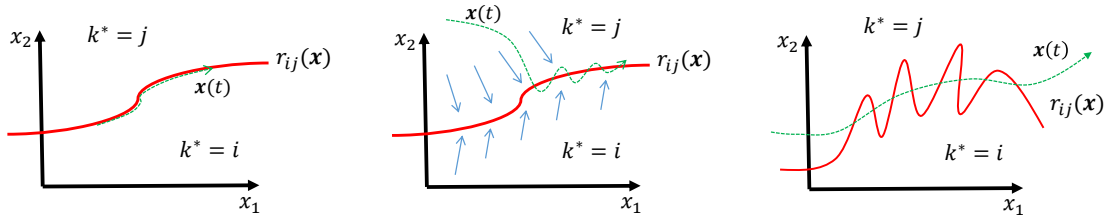
$$r_{ij}(\boldsymbol{x}) = 0 \tag{4.58}$$

If $r_{ij} > 0$ then the decision is $k = i$, while if $r_{ij} < 0$ the decision is $k = j$. When using the R* Computed Torque controller for a 1-DoF system, the decision boundary between $k = i$ and $k = j$ takes the form

$$0 = r_{ij}(\boldsymbol{x}) = \left[\frac{R_i^2 - R_j^2}{\frac{1}{R_i^2} - \frac{1}{R_j^2}} \right] - \frac{\tau_E}{\tau_I} \tag{4.59}$$

Boundary decision could also be created by the inequality constraints in the optimization. The most relevant example is motor speed saturation which can create decision boundaries of the type:

$$0 = r_{ij}(\boldsymbol{x}) = w_{max} \pm \frac{\dot{q}}{R_i} \tag{4.60}$$



(a) Staying on the decision boundary

(b) Decision boundary attracting all trajectory

(c) High-frequency crossing of the decision boundary

Figure 4-12: Situations leading to fast gear-ratios switching. r_{ij} is a decision boundary between $k^* = i$ and $k^* = j$

Situations leading to fast gear-ratios switching

Fig. 4-12 illustrates different possible class of situations leading to fast switching.

The theoretical closed-loop behavior, when exactly on the boundary decision is a degenerative case of ambiguous infinitely fast jumps between gear-ratios. In practical implementations, because decision are delayed, staying in the vicinity of the decision boundary $r_{ij} \approx 0$ could lead to chattering, with a period depending on implementation aspects such as computing delay, sampling time and hysteresis in the gear-selection logic.

Fig. 4-12a and 4-12b illustrate two possible class of behavior for which the system stay in the vicinity of the decision boundary and would lead to chattering behavior. A situation of type (a) represent a case where the system trajectory coincide with the boundary decision and a situation of type (b) represent a more severe case where all trajectory would be attracted to the decision boundary, i.e. a type of undesired sliding mode. Those two types of situation can be ruled-out as impossible, if conditions guaranteeing convergence to the desired trajectory are met. By contradiction, state cannot both converge on the desired trajectory and stay on the decision boundary, unless for very special cases where the desired trajectory is in the sub-domain defined by the decision boundary.

Example of gear-shift chattering with a car going up-hill In practice, a situation of type (b) can be a typical failure mode of switched system. This section explore with a simple example of how this situation can arise and how the proposed

control schemes make sure this is avoided. Imagine a car with an automatic transmission going up-hill, with a gear-selection scheme only based on a velocity decision boundary. The car starts from rest with its low-gear $k = 1$ and accelerate. After reaching reaching a threshold velocity, the automatic transmission then engage the high-gear $k = 2$. With the high-gear however, the effect of gravity is much larger and the car slow down until it cross again the threshold velocity. The low-gear $k = 1$ is then engaged again and the car starts accelerating. The car then cross again the threshold and the cycle would continue forever. What is happening here is that the car converge toward its velocity goal when $k = 1$ but diverge when $k = 2$. With the proposed R* Computed Torque controller, this could happen if gravity forces were underestimated in the model. However, if gravity forces are either correctly modeled, or alternatively included in the uncertainty bounds with the sliding mode controller, then convergence would be guaranteed for both modes, and enough torque to continue accelerating toward the target velocity would have been applied.

So far, situations of chattering in the vicinity of decision boundary of type (a) and (b), have been ruled-out impossible if the controller is designed to guarantee convergence for all discrete mode k. However, situations of type (c), illustrated at Fig. 4-12c and called high-frequency switching, is still possible and un-desirable. The decision boundary can very complex for highly non-linear systems, hence even a simple trajectory can cross decision boundaries at undesirable high-frequencies.

Minimum cycle time with Rollout

The Rollout approach is efficient to avoid situations of type (c), high-frequency switching. By optimizing over a time horizon, the Rollout gear-selection scheme is acting as a low-pass filter. Moreover, with the Rollout approach, a minimum value for the cycle period of gear-shifting back-and-forth between two gear-ratios can be guaranteed.

The cycle time Δt of gear-shifting from arbitrary mode i, to another arbitrary mode j, and back to the initial mode i, is lower bounded. If the system follows the

desired trajectory exactly, the minimum value is:

$$\Delta t \geq \frac{Q}{C^{max}} \tag{4.61}$$

where Q is the cost penalty for changing the gear-ratio, C^{max} is the maximum instantaneous cost. With the usual quadratic torque criterion:

$$C^{max} = \max \left[\boldsymbol{\tau}^T \boldsymbol{\tau} \right] \tag{4.62}$$

In an arbitrary situation the minimum value is:

$$\Delta t \geq \frac{Q}{C^{max} + \dot{C}^{max} h} \tag{4.63}$$

where the variable h is the time horizon and $\dot{C}$ represents the sensitivity of computed costs due to changes of trajectories in the predictive simulations. When the system has reach the desired trajectory, the real trajectory and all simulated trajectories follow the desired one, and this sensitivity term vanish. Details of those derivations are available in appendix B.4.

4.4.9 Parameters selection guidelines

R* Computed Torque The only parameters with the computed torque controller are matrices K_D and K_P. Those can be understood as multi-DoF proportional and derivative gains. Stability is guaranteed for any positive definite matrices.

For simplicity, matrices can be diagonal and parametrized by:

$$K_{ii}^D = \frac{2}{\tau} \quad K_{ii}^P = \left(\frac{1}{\tau} \right)^2 \tag{4.64}$$

where τ is a time constant parameter driving the desired exponential error convergence rate.

R* Sliding Mode For the sliding mode controller, there is two parameters to tune the convergence rate, λ which is inversely proportional to error convergence time constant after reaching the sliding surface, and η which is inversely proportional to the guaranteed time for reaching the sliding surface. The discontinuous gain should then be set based on disturbance bounds as follow:

$$K_{ii} = \max_{\boldsymbol{d}} \left[H_k^{-1} \boldsymbol{d} \right]_i + \eta \tag{4.65}$$

in order to guarantee convergence.

Rollout gear-selection For the Rollout gear-selection scheme, there is two free parameters, the time horizon h and the gear-shift penalty Q. If the VGA used by the robot have a gear-shift switching delay of Δt_{shift}, then the Q value can be set, using eq. (4.61), to guaranteed that the robot-level controller never ask its actuators for gear-shifts faster the physical limit:

$$Q = \Delta t_{shift} \max \left[\boldsymbol{\tau}^T \boldsymbol{\tau} \right] \tag{4.66}$$

For the time horizon h it is not necessary a case of larger is better, since the cost computed with fixed gear-ratios over a long simulated period does not represent well the real future cost, since gear-shift are possible in the real future. Very short time horizon would largely inhibit gear-shifting since gain of changing the gear-ratios on a short period would be small compared to the gear-shift penalty.

A rule of thumbs, as a starting point, is to set the time horizon to the expected average amount of time spent between two gear-shifts. This is motivated by the fact that in that case, most predicted trajectories with fixed gear-ratios will be representative of the real future trajectory. Simulations and experiments can be used to experimentally adjust this parameter. Alternatively, the cost function could also be discounted: putting more weight on immediate cost and less weight on far ahead uncertain cost.

4.5 Trajectory planning

Although not the focus of this thesis, this section briefly discusses a motion planning algorithms, that is an important piece of the global control solution for the model-based control architecture. The proposed model-based controllers are theoretically globally stable; given a fixed goal $\boldsymbol{q}_d$, the closed system should converge on it starting from any initial conditions. However, this not considering:

- Motor torque saturation (more generally input constraints)
- Possible obstacle on the path (more generally state constraints)

Moreover, when simply given a fixed goal $\boldsymbol{q}_d$ to the R* Computed Torque controller, while the gear-ratios selection is locally optimized, the overall trajectory is given by

$$\boldsymbol{q}(t) = (\boldsymbol{q}_0 - \boldsymbol{q}_d)e^{-\frac{t}{\tau}} + \boldsymbol{q}_d \tag{4.67}$$

if the controller is parametrized as described by eq. (4.64). Hence, the followed trajectory would be simply a decreasing exponential function of time for each joint angles, which can be clearly highly inefficient and sub-optimal. Hence the role of the motion planning algorithm is to find a feasible reference trajectory away from state and inputs constraints, and ideally an optimal trajectory in term of an integral cost.

As discussed in section 4.0.2, the two class of algorithm that are suited to solve a motion planning problem for a hybrid robotic systems are:

- Mixed-integer programming
- Sampling-based search algorithms

In general, mixed-integer programming is better suited to find optimal solution to simple situations, while sampling-based approach are more efficient at finding feasible sub-optimal solutions in complex situations (many constraints), for instance finding a path in a maze. The approach that was implemented here, focuses on finding feasible low-torque trajectories quickly, and is based on rapidly-exploring random trees.

4.5.1 RRT algorithm for Robots with Discrete Gear-ratios

Rapidly-exploring random trees (RRT) have been highly popular over the past decade as a sampling-based approach to quickly identify feasible trajectories in complex motion planning problems [33] [34]. Its main feature is its property of biasing the random search toward un-explored regions of the state space. This technique also naturally works with discretized action-space, which makes a good fit for switched dynamics system where the discrete mode is a control input.

Action set To implement the algorithm for robot using variable gear-ratios actuators the action space is parametrized into a finite set of possible actions at each time steps. Each actuator torque is split into p possible level:

$$\tau_i \in \boldsymbol{T} : \{-\tau_{max}, ..., 0, ..., +\tau_{max}\} \tag{4.68}$$

where p is an odd number bigger or equal to 3. Possible gear-ratios modes, are already a discrete set, but the possible selectable gear-ratios set $\boldsymbol{K}$ is state-dependent:

$$k \in \boldsymbol{K}(\dot{\boldsymbol{q}}) : \left\{k \in \{0, .., l\} \,|\, \boldsymbol{w} = R_k \dot{\boldsymbol{q}} \in \boldsymbol{W}\right\} \tag{4.69}$$

in order to satisfy rotor-velocity constraints. Hence the global possible action set $\boldsymbol{A}$ at each time step is the combination of all those possible p torque levels for the m actuators and the available state-dependent gear-ratios options:

$$a \in \boldsymbol{A}(\dot{\boldsymbol{q}}) : \left\{ (\boldsymbol{\tau}, k) \mid \tau_1 \in \boldsymbol{T}, ..., \tau_m \in \boldsymbol{T}, k \in \boldsymbol{K}(\dot{\boldsymbol{q}})\right\} \tag{4.70}$$

The maximum number of possible discrete actions at each step is thus $l\, p^m$.

State-space The search is then conducted in the full dynamic state space of the robotic system (as opposed to simply searching in the configuration space):

$$\boldsymbol{x} = (\dot{\boldsymbol{q}}, \boldsymbol{q}) \tag{4.71}$$

hence the dimension of the state-space is $2n$. The state constraints are defined by:

$$\boldsymbol{x} \in \boldsymbol{X} : \left\{ (\dot{\boldsymbol{q}}, \boldsymbol{q}) \mid \boldsymbol{q} \in \boldsymbol{C}_{free} \ , \ \dot{\boldsymbol{q}} \in \boldsymbol{V} \right\} \tag{4.72}$$

The constraints on joint positions are defined by the free configuration space domain $\boldsymbol{C}_{free}$ [38], which exclude configuration leading to collision with obstacles. The constrains on joint velocities are defined as the set for which there exist at least one gear-ratios configuration k leading to allowable rotor velocities:

$$\boldsymbol{V} : \left\{ \dot{\boldsymbol{q}} \mid \exists k \in \{0, .., l\} \Rightarrow R_k \dot{\boldsymbol{q}} \in \boldsymbol{W} \right\} \tag{4.73}$$

Note that $\boldsymbol{W}$ is the allowable rotor velocity set defined as:

$$\boldsymbol{W} : \left\{ \boldsymbol{w} \mid -w_{max} < w_i < w_{max} \quad \forall \, i \right\} \tag{4.74}$$

System evolution The discrete state evolution of the system can be modeled with proposed dynamical model presented in section 4.2, projecting in the future with a small time step Δt:

$$\dot{\boldsymbol{q}}_{t+1} = H_k^{-1} \left[R_k \tau - \boldsymbol{c}_k(\dot{\boldsymbol{q}}_t, \boldsymbol{q}_t) \right] \Delta t + \dot{\boldsymbol{q}}_t \tag{4.75}$$

$$\boldsymbol{q}_{t+1} = \dot{\boldsymbol{q}}_t \Delta t + \boldsymbol{q}_t \tag{4.76}$$

Implementation With the discrete action set, the state-space constraints and the discrete time system evolution, the RRT algorithm can be applied to search for feasible solutions. The other steps are a textbook application of the RRT algorithm [34] and are omitted here. After execution the algorithm return both a trajectory of states and control inputs. However, the proposed model-based controllers, R* Computed Torque and R* Sliding Mode, only required the trajectory to use as a reference and control inputs are fully computed online in closed-loop. Details on the software implementation used for the experimental robotic system is discussed in Chapter 5.

4.6 Dynamic programming approach

"The true logic of this world is in the calculus of probabilities."

– *James Clerk Maxwell*

This section explores an alternative approach to synthesize feedback laws. The idea is to discretize the continuous control problem into a graph search problem, with transition probabilities, where the discrete input actions can be considered naturally. Dynamic programming approaches can then be used to find global optimal feedback policies. This is exemplified here for two simple systems.

4.6.1 Problem formulation

The control problem of obtaining the global desired behavior is formulated as minimizing a scalar cost J that is a function of the state trajectory $\boldsymbol{x}(:)$ and the inputs trajectory $\boldsymbol{u}(:)$, while constraining both states and inputs to be in their respective domains:

$$\min_{\boldsymbol{u}(:)} J(\boldsymbol{x}(:), \boldsymbol{u}(:)) \tag{4.77}$$

$$s.t. \quad \dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) \quad \boldsymbol{u} \in \boldsymbol{U}(\boldsymbol{x}) \quad \boldsymbol{x} \in \boldsymbol{X} \tag{4.78}$$

4.6.2 Constraints

The input set $\boldsymbol{U}(\boldsymbol{x})$ is defined by allowable motor torques respecting saturations and gear-ratios that are feasible given actual joint velocities:

$$\boldsymbol{U}(\boldsymbol{x}) : \begin{cases} \tau_i \in [-\tau_{max}, \tau_{max}] \; \forall \; i \\ k \in \{0, ..., l\} \mid \boldsymbol{w} = R_k \dot{\boldsymbol{q}} \in \boldsymbol{W} \end{cases} \tag{4.79}$$

where $\boldsymbol{W}$ is the set of allowable rotor velocities. The state set $\boldsymbol{X}$ include a range of acceptable states and could exclude some regions if desired.

4.6.3 Cost function

Here cost function is defined as additive instantaneous costs g over an infinite horizon. This form leads to simpler time-independent control policies [3].

$$J = \lim_{t \to \infty} \int_0^t g(\boldsymbol{x}(t), \boldsymbol{u}(t)) dt \tag{4.80}$$

Two different additive cost functions are investigated; a quadratic cost function where error is weighted against control effort:

$$g(\boldsymbol{x}, \boldsymbol{u}) = \sum w_{ii}^x \, x_i^2 + \sum w_{ii}^u \, \tau_i^2 \tag{4.81}$$

where w are weighting factors (note that there is no penalty for either gear ratio options); and a function corresponding to the minimal time problem:

$$g(\boldsymbol{x}, \boldsymbol{u}) = \begin{cases} 1 & \text{if target not yet reached} \\ 0 & \text{if target is reached} \end{cases} \tag{4.82}$$

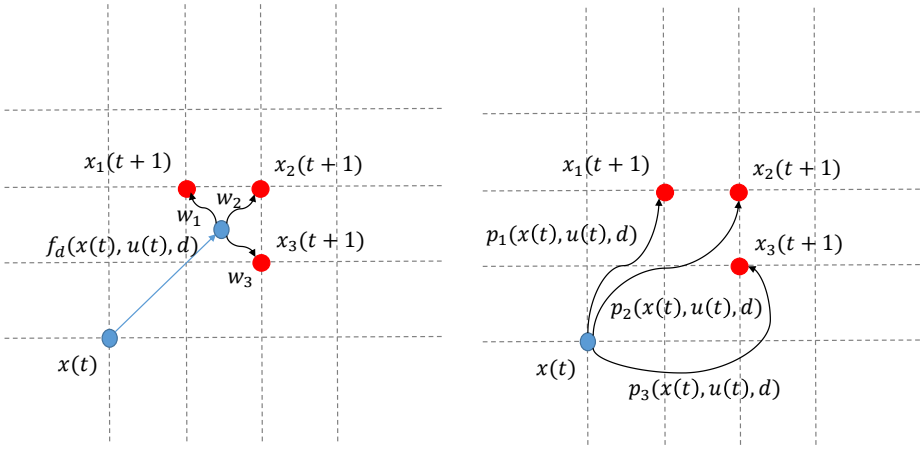
Note that in this section, the goal is fixed and set to $\boldsymbol{x}_d = \mathbf{0}$.

4.6.4 Value Iteration

A dynamic programming algorithm, also known as value iteration, is then used to solve for almost exact optimal infinite-horizon policies.

Discretization and conversion to a Stochastic Shortest Path problem

An evenly spaced grid of nodes is used to represent the state space. The discrete state evolution of the system is computed by projecting in the future with a small



(a) Interpolation (b) Agregation probabilities

Figure 4-13: Approximation of the cost-to-go: two interpretations

time step Δt, using the continuous time EoM of the robotic system:

$$\boldsymbol{x}(t+1) = f_d(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)) \tag{4.83}$$

$$\begin{bmatrix} \dot{\boldsymbol{q}}(t+1) \\ \boldsymbol{q}(t+1) \end{bmatrix} = \begin{bmatrix} H_k^{-1}\left[R_k\boldsymbol{\tau}(t) - \boldsymbol{c}_k(\dot{\boldsymbol{q}}(t), \boldsymbol{q}(t)) + \boldsymbol{d}(t)\right]\Delta t + \dot{\boldsymbol{q}}(t) \\ \dot{\boldsymbol{q}}(t)\Delta t + \boldsymbol{q}(t) \end{bmatrix} \tag{4.84}$$

However, the equations of motion of the system can lead the system to a future state that will not be exactly on a grid point corresponding to a discrete node, see Fig. 4-13a. To solve this problem, the cost-to-go of the next point is computed by interpolating the cost-to-go of the neighboring grid points, based on geometric interpolation weight:

$$J(\boldsymbol{x}(t+1)) \approx \sum w_j J(\boldsymbol{x}_j(t+1)) \qquad \sum w_j = 1 \tag{4.85}$$

The interpolation can also be interpreted as computing an expected value with probabilities w_j of arriving on a neighboring node j:

$$\sum w_j J(\boldsymbol{x}_j(t+1)) = E\left[J(\boldsymbol{x}(t+1))\right] \tag{4.86}$$

$$\text{if} \quad p\Big(\boldsymbol{x}(t+1) = \boldsymbol{x}_j \;\Big|\; \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)\Big) = w_j \tag{4.87}$$

Thus the value iteration updates have the form:

$$J(\boldsymbol{x}_i) \Leftarrow \min_{\boldsymbol{u}} \Big[g(\boldsymbol{x}_i, \boldsymbol{u}) + \sum w_j J(\boldsymbol{x}_j(t+1)) \Big] \tag{4.88}$$

and the optimal control policy $\pi : \boldsymbol{x} \mapsto \boldsymbol{u}$ is computed for all nodes as:

$$\pi(\boldsymbol{x}_i) = \operatorname*{argmin}_{\boldsymbol{u}} \Big[g(\boldsymbol{x}_i, \boldsymbol{u}) + \sum w_j J(\boldsymbol{x}_j(t+1)) \Big] \tag{4.89}$$

or equivalently with the probabilistic view:

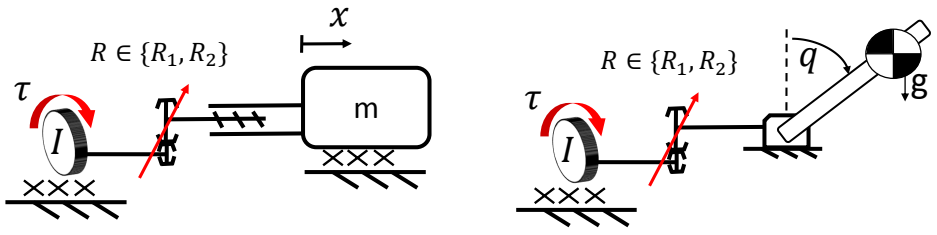
$$J(\boldsymbol{x}_i) \Leftarrow \min_{\boldsymbol{u}} E\Big[g(\boldsymbol{x}_i, \boldsymbol{u}) + J(\boldsymbol{x}_j(t+1)) \Big] \tag{4.90}$$

$$\pi(\boldsymbol{x}_i) = \operatorname*{argmin}_{\boldsymbol{u}} E\Big[g(\boldsymbol{x}_i, \boldsymbol{u}) + J(\boldsymbol{x}_j(t+1)) \Big] \tag{4.91}$$

As illustrated at Figure. 4-13b, this scheme can also be interpreted as aggregating the continuous space state into discrete nodes with the weight factors corresponding to aggregation probabilities. Hence, the problem that is solved corresponds to a stochastic shortest path problem. Knowledge regarding stochastic distribution of disturbances could be included easily with this formulation. However here, disturbances are fixed to their expected value of zero, the certainty equivalence assumption [3]. Even if the uncertainty of the system is not modeled directly, the interpolation scheme has an effect equivalent to including a disturbance leading to an uncertainty on the state evolution of roughly the size of the grid.

4.6.5 Example systems

Numerical results of feedback policy to reach a fixed-target are presented for two simple robotics system using VGA. The first system, see Fig. 4-14a, is a single-axis linear actuator with two gear-ratios, the output dynamic is considered to be a linear mass-damper and rotor-velocity saturations are included. The second system, see Fig. 4-14b, is a non-linear inverted pendulum with no rotor-velocity saturation.



(a) Linear mass-damper (b) Inverted pendulum

Figure 4-14: Two studied robotic systems

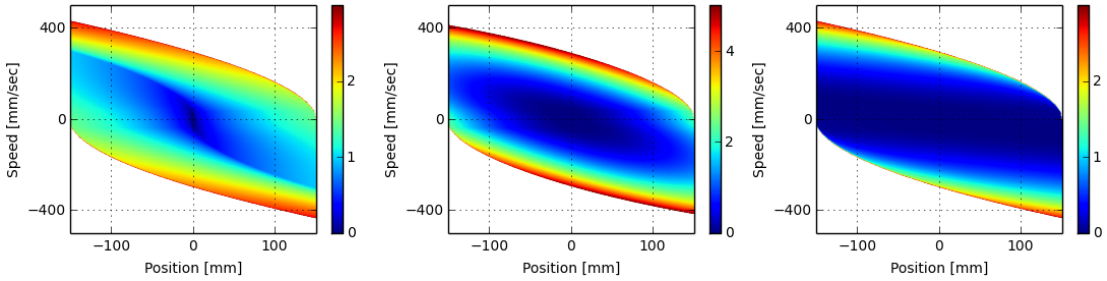
4.6.6 Implementation

The discretization parameters used are as follow: the state space is discretized into a 101 x 101 grid (101 equally spaced positions and 101 equally spaced speed values) leading to 10201 possible nodes, the actuator torque is discretized into 21 possible torque values, leading to a total of 42 possible control actions including the gear ratio selection. The state transition is computed using a time discretization of $\Delta t = 0.05$ sec. The interpolation weights are computed using a bivariate spline approximation. The value iterations are stopped manually when the maximum difference between J_k and J_{k+1} are many orders of magnitude smaller than the variations of J across the state space. The computation takes on average 200-500 iterations and 2-5 minutes.

4.6.7 Numerical results

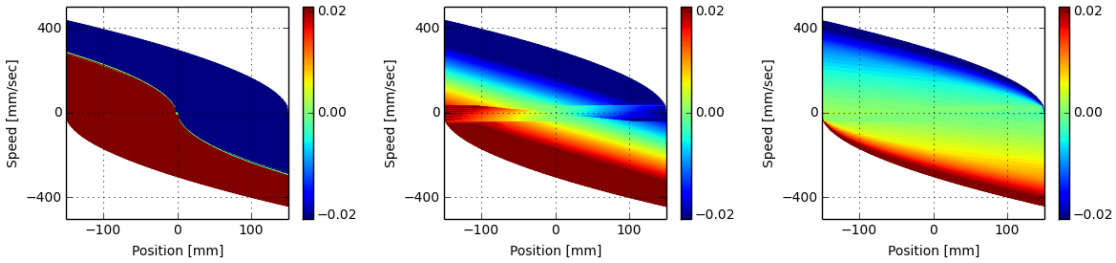
System 1 - Linear mass-damper robot

Fig. 4-15 to 4-17 illustrate the numerical results for system 1. Three different cost functions are explored, a quadratic cost function, a minimum time cost and minimum energy case which is simply a special case of the quadratic cost where weight are sets to drastically penalize control inputs over state error. The absence of color indicates states with no solution (a constraint will be violated for any possible control actions). Fig. 4-18 shows the closed loop behavior of the system in the phase plane when the optimal policy is applied.



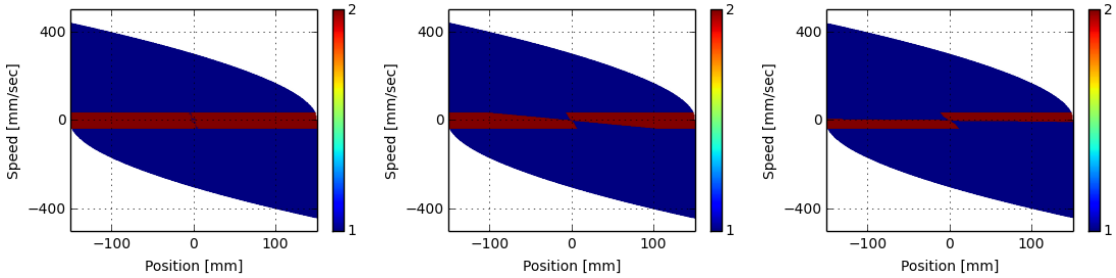
(a) Minimum time (b) Quadratic cost (c) Minimum energy

Figure 4-15: Optimal cost-to-go J^*



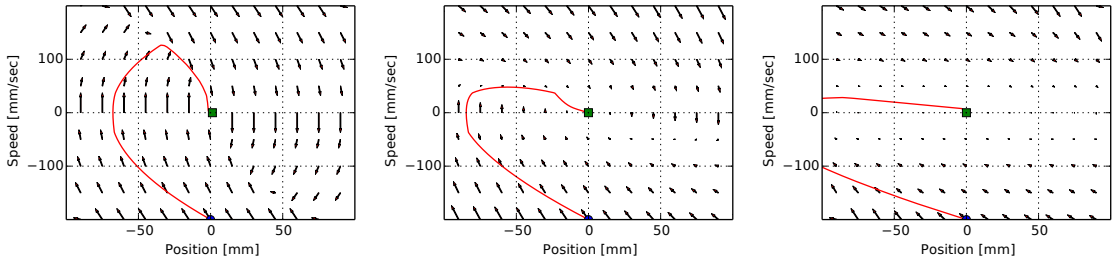
(a) Minimum time (b) Quadratic cost (c) Minimum energy

Figure 4-16: Optimal policy for the continuous torque command τ [Nm]



(a) Minimum time (b) Quadratic cost (c) Minimum energy

Figure 4-17: Optimal policy for the gear-ratio mode selection k



(a) Minimum time (b) Quadratic cost (c) Minimum energy

Figure 4-18: Closed loop behavior with the optimal policy illustrated in the phase plane

Minimum time For the minimum time problem, the optimal policy is a bang-bang law for the torque and always using highly-geared mode when possible. Note that the bang-bang switching curve accounts for the fact the large gear ratio will be used during the final part of the trajectory.

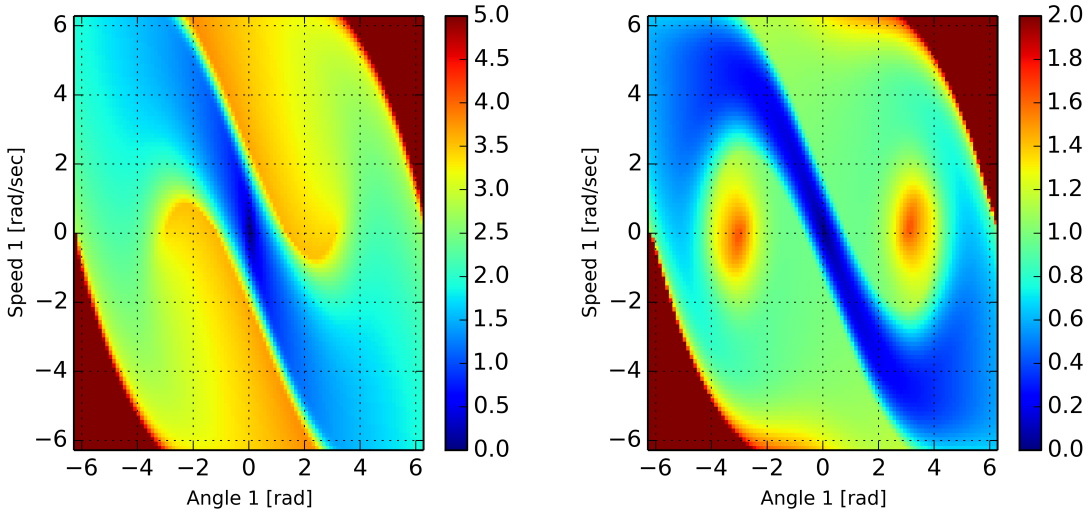
Quadratic cost For the quadratic cost, the gear-ratio selection optimal policy is almost as simple as the minimum time problem except for small features in quadrant II and IV. The more interesting result comes from the continuous torque control law, the gains when using the large reduction ratio are larger than those when using the small reduction ratio. This results in the controller taking action mainly at low speed when its actions have the biggest impacts on the system, and lead to a highly non-linear closed-loop behavior.

Minimum energy For the minimum energy controller, interestingly the mode selection policy is not trivial even for this simple linear model. This shows that it does not take much complexity to have non-trivial optimal policies for hybrid systems. Here, the large reduction ratio is used almost only for braking, and in quadrant II and IV the small reduction ratio is used even at low speed to coast with low viscous resistance. Also globally the gains are much lower than the other controllers except for zones where it is necessary to use energy to stay in the state-domain.

System 2 - Inverted pendulum robot

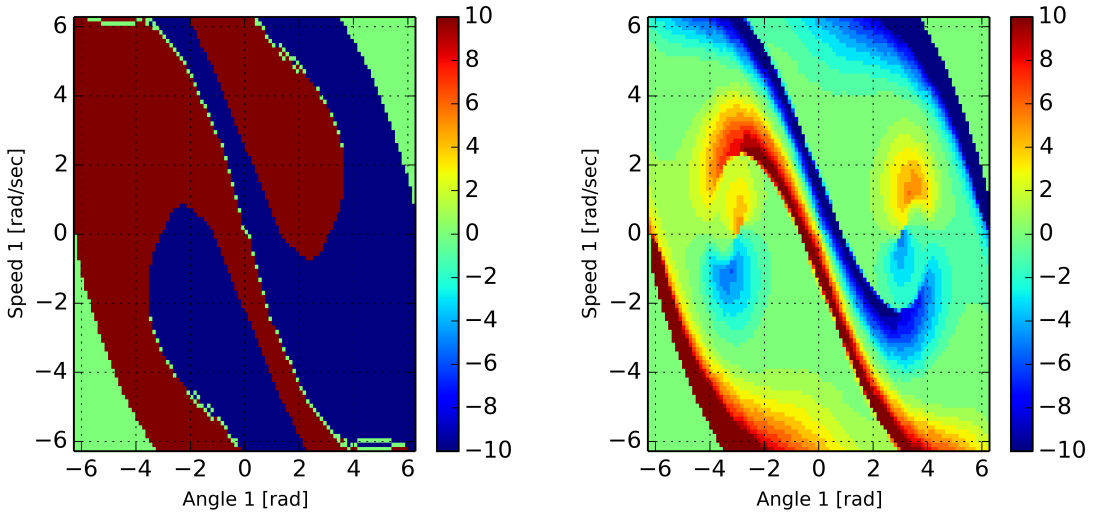
Figure 4-19 illustrates the computed optimal cost-to-go for the inverted pendulum system for both a minimal time goal and a quadratic cost minimization. Figure 4-20 shows the optimal torque policy and Figure 4-21 shows the optimal gear selection policy. The resulting closed loop behavior is illustrated in the phase plane at Figure 4-22. Figure 4-22 also shows closed-loop state trajectories and control inputs for a simulation starting at q = -2 rad.

For both cost functions the optimal policy for the discrete gear-ratio mode k is quite complex. This illustrates that optimal solutions for this type non-linear hybrid



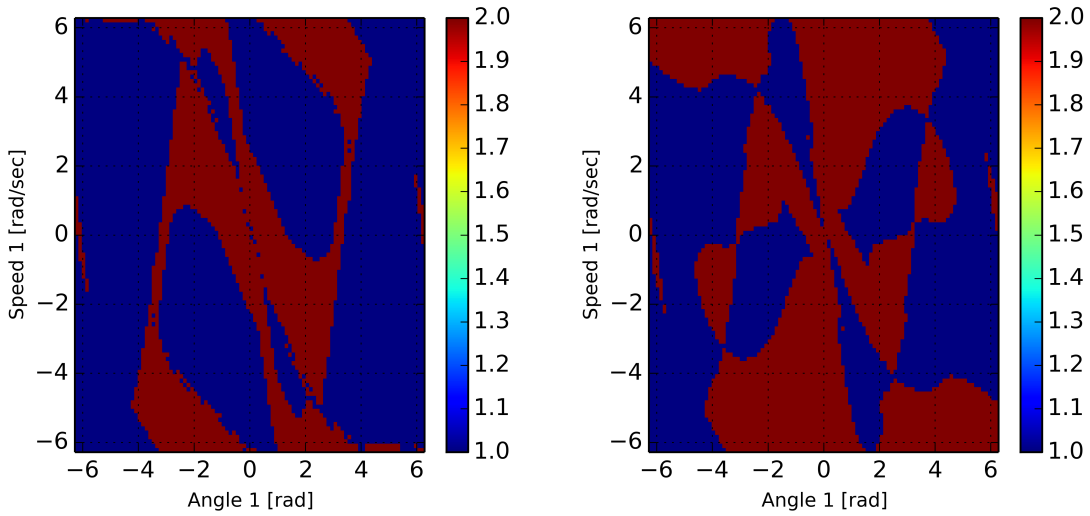
(a) Minimum time (b) Quadratic cost

Figure 4-19: Optimal cost-to-go J^*



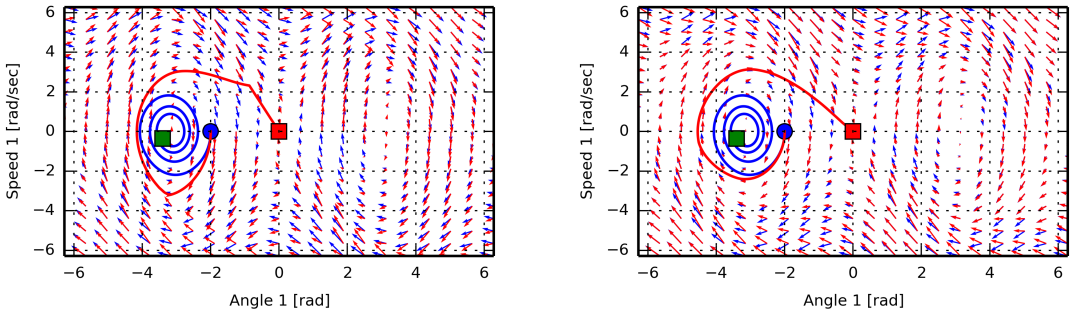
(a) Minimum time (b) Quadratic cost

Figure 4-20: Optimal policy for the continuous torque command τ [Nm]



(a) Minimum time (b) Quadratic cost

Figure 4-21: Optimal policy for the discrete gear selection $k \in \{1, 2\}$



(a) Minimum time (b) Quadratic cost

Figure 4-22: Closed loop behavior with the optimal policy illustrated in the phase plane. Two trajectories starting at $q = -2$ are illustrated, blue is open-loop, red is closed loop

system representative of robotic systems are not trivial, even for a single axis. One observation regarding the gear-ratio selection policy, is that the larger second gear-ratio is used often in situation where the robot needs to be drastically deviated from its natural motion, and the small second gear-ratio is used when the controller let the robot move in the natural direction. This connects back with the idea of attenuating the load-dynamics, with a large gear-ratio when it is advantageous, or leveraging the load-dynamics, with a small gear-ratio when the natural motion is advantageous.

4.6.8 Advanced dynamic programming techniques

Value iteration is a powerful tool, but computations are intractable for high-dimensional systems. However, there exist many approximate techniques that can be used; approximate dynamic programming is also known as reinforcement learning. The approach proposed in this section, of formulating the control of hybrid robot as a stochastic shortest path problem, could thus be used in conjunction with many approach derived in the field of artificial intelligence such as Q-learning. However, the problem of approximating the cost-to-go, the state-space or the policies in a lower dimensional space, to make computation tractable, is not trivial for non-linear robotic system.

4.7 Simulation Results

4.7.1 Model-based approach

Here, the advantages of dynamically changing the gear-ratios, using the R* computed torque controller, are illustrated using simulations of two robots: first a 1-DoF pendulum, then a 3-DoF arm. Both robots are considered having VGA with two possible gear-ratios: 1:1 or 1:10. Reference low-torque trajectories to reach target positions are computed with a RRT algorithm. The first simulated experiment uses a single-axis inverted pendulum equipped with a VGA, see Fig. 4-14b, with the task of reaching the up-right position starting at the bottom. Fig. 4-23 shows the robot tracking the reference low-torque trajectory, where at first the robot accumulates energy, using the 1:1 gear-ratio, and then finishes the motion using the 1:10 gear-ratio. When the gravitational forces are pushing advantageously toward the trajectory the controller select the 1:1 gear-ratio, but when it is advantageous to fight the intrinsic actuator dynamics instead, the 1:10 gear-ratio is selected. This can be seen by comparing the trajectory to natural phase plane vectors as illustrated by Fig. 4-24.

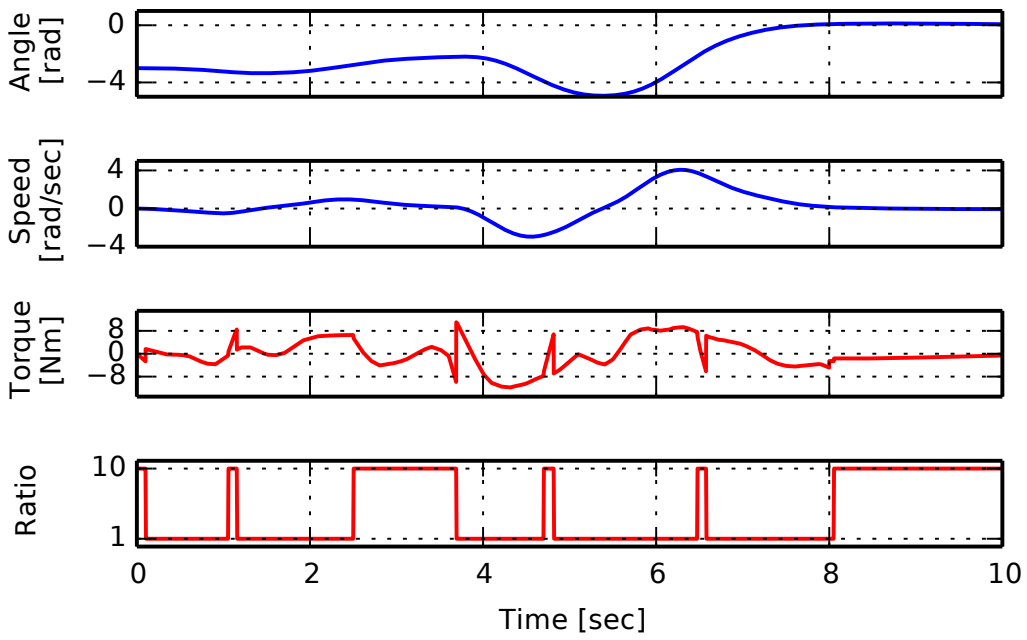
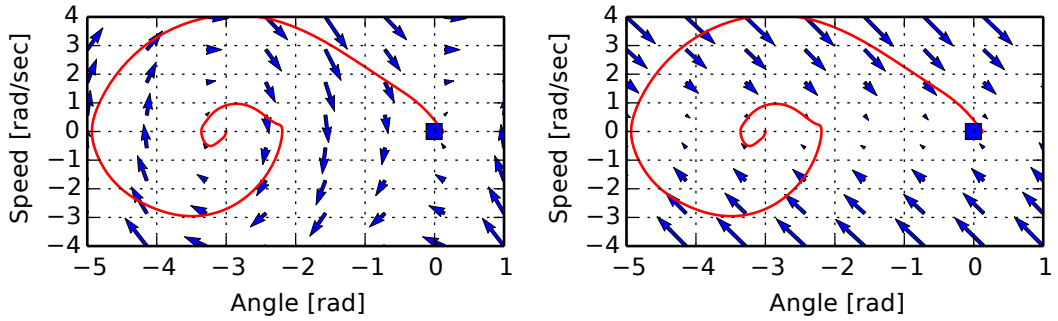


Figure 4-23: 1-DoF robot simulation: states and inputs trajectory



(a) Reduction ratio R=1 (b) Reduction ratio R=10

Figure 4-24: Trajectory superposed with natural dynamics vectors

In the second experiment, a 3-DoF manipulator is tasked with going from configuration A to configuration B with the 3D trajectory shown at Fig. 4-25. For this robot the controller is actively selecting the best gear-ratios matrix R_k out of the possible $2^3 = 8$ options. Fig. 4-26 shows the control inputs activity. During the initial falling-down phase, at around $t = 1$, the robot is using 1:1 gear-ratios for all actuators, leveraging gravitational torques. In contrast, during the final lifting phase, at around $t = 6$, the robot is using 1:10 gear-ratios for all actuators.

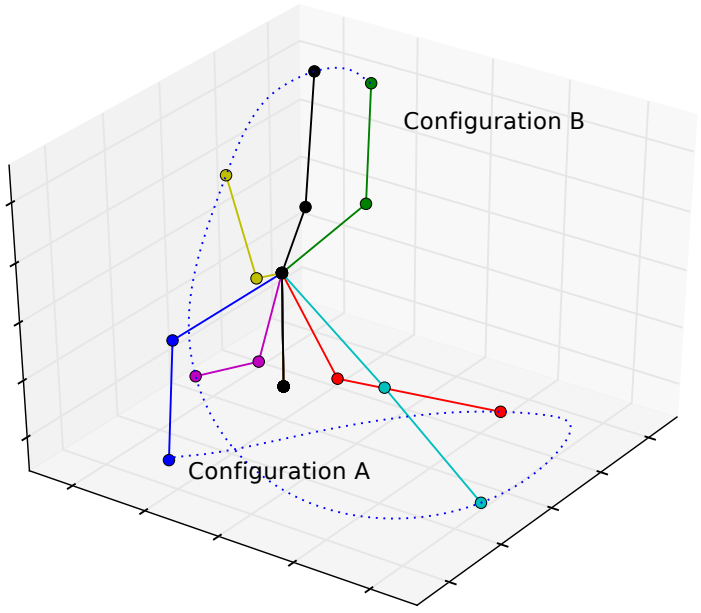


Figure 4-25: 3-DoF robot simulation: 3D trajectory

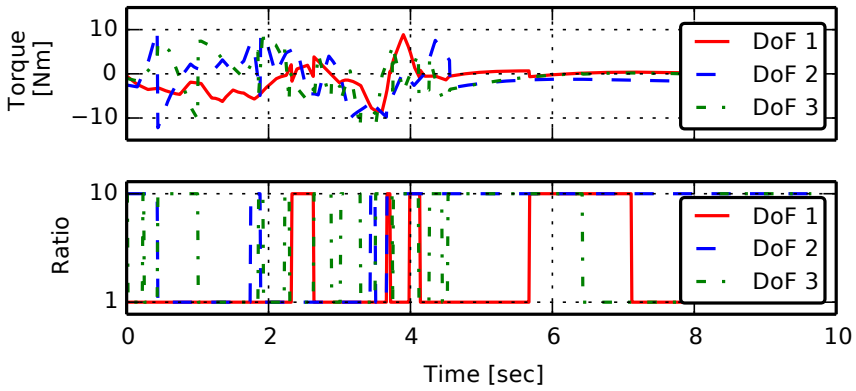


Figure 4-26: 3-DoF robot simulation: control inputs trajectories

4.7.2 Comparison to fixed-gear performance

To evaluate the performance gain of actively changing the gear-ratio, simulations with fixed gear-ratios are conducted where a regular computed torque controller tracks the same trajectories. Results are summarized in Table 4.1, in terms of maximum absolute torque, which relates to the required size and weight of motors, and integral of torque squared, which relates to power consumption. Active gear-ratios selection is found to greatly improve both metrics, especially for the 3-DoF robot trajectory where the arm must both achieve high-speeds and also sustain a constant gravitational load at the final configuration. Note that in those simulations high-velocity with 1:10 reductions is inhibited by friction in the motors, no maximum rotor velocity is enforced. For the 3-DoF trajectory, active gear-shifting is found to reduce the maximum torque required by a factor two and the integral of the torque square by a factor 10, compared to any of the fixed-gear options. Those results show the potential of using variable gear-ratio transmissions for huge improvements in terms of actuator size and power consumption. Moreover, here in the simulations, the load was always the same manipulator in different dynamic situations. If the load dynamics is radically changing because of different contact situations with the environment, the performance gain of changing the gear-ratios could be even greater.

Table 4.1: Required torque comparison

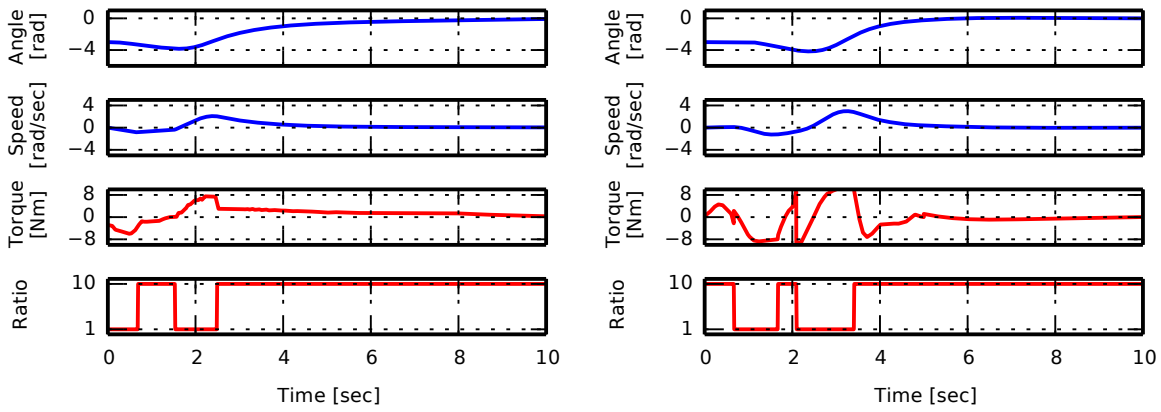
	Fixed gear 1:1	Fixed gear 1:10	Active gearshifting 1:1 or 1:10
Max Absolute Torque [Nm]			
1-link robot	15	88	12
3-link robot	24	42	12
Torque squared integral $\int (\boldsymbol{\tau}^T \boldsymbol{\tau}) dt$			
1-link robot	377	8133	226
3-link robot	2774	3617	295

4.7.3 Comparison to Value Iteration

For low-dimensional systems, numerical results obtained with the value iteration algorithm, when discretization is very fine, can almost be seen as a ground truth for the optimal trajectory and optimal global feedback policy. It is thus interesting to compared the model-based approach (RRT planning + R* Computed Torque) to a solution obtained with value iteration, to evaluate how far from the global optimum the model-based control scheme is in some situations. Fig. 4-27 shows side-by-side results for a pendulum swing-up task. The trajectory behavior of both solutions is roughly similar, both solutions do a pumping motion to accumulate kinetic energy. The R* Computed Torque controller however apply torques more aggressively, to track the reference trajectory generated by the RRT algorithm which is a rough feasible but un-optimized solution. In term of integral of torque-squared for the whole trajectory, the value iteration solution leads to 72 $(Nm)^2 \times sec$, compare to 180 $(Nm)^2 \times sec$ for the RRT with R* computed torque solution.

It is interesting that the model based approach can find a solution with a trajectory and a gear-shifting pattern almost identical to the optimal solution found using value iteration. However, the main drawback is the rough reference trajectory, which as a lot of discontinuities (un-bounded jerk) because the RRT algorithm use a discrete version of the world. A possible approach to alleviate this would be and intermediary step: smoothing out the reference trajectory with a local optimization before it is sent to the R* Computed Controller. By using the RRT trajectory solution as initial conditions and also the gear shift sequence solution, a local optimization would now

be much easier to conduct. Quadratic programming algorithms could then be used efficiently to fine tune the reference trajectory solution.



(a) Value Iteration

(b) RRT with R* Computed Torque

Figure 4-27: Model based approach compared to Value Iteration

4.7.4 Fast gear-shifting inhibition

In the simulations presented so-far, fast gear-shifts were not inhibited by any of the proposed fast-switching inhibiting schemes. Fig. 4-28 shows simulation results, illustrating the gear-ratio command, for the same situation but with three different controllers.

This figure shows that the hysteresis scheme is not necessary an improvement over the point-by-point gear-ratio selection, since for the gear-shift at $t = 1.8$, it would have been better to avoid it completely. The Rollout approach is shown more successful at filtering-out high-frequency switching. Furthermore, another interesting advantageous side effect is observed: with the Rollout approach, the first gear-shift is commanded slightly in advance. This can be advantageous when controlling a real system where the gear-shift process is executed with a delay: the simulation can model this delay and the controller can react in advance accordingly.

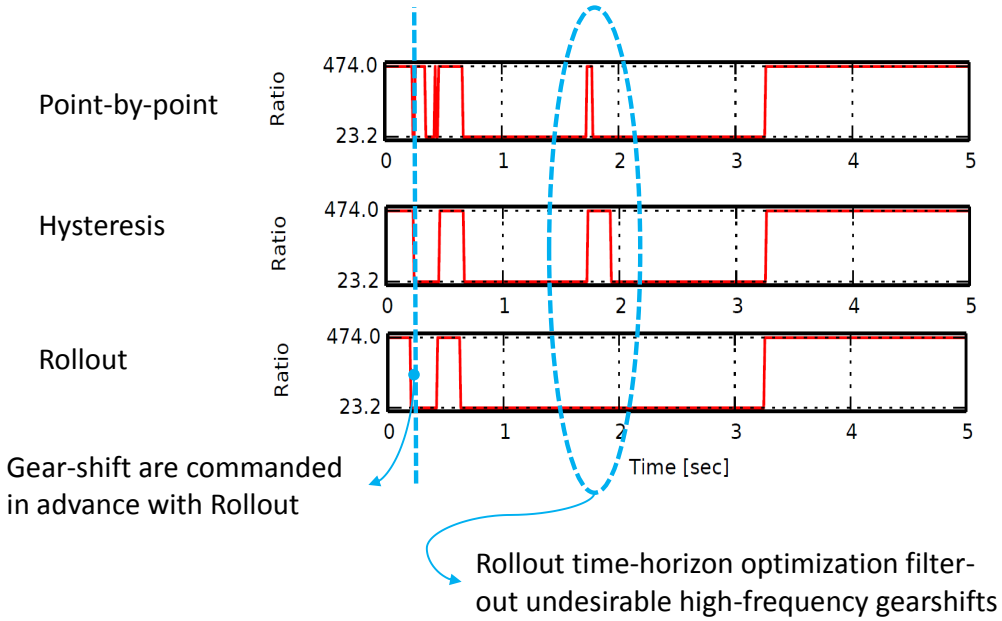


Figure 4-28: Fast gear-shifting inhibition

4.8 Experiments Results

This section presents experimental results, focusing on the high-level control aspect, obtained with the experimental DSDM-arm presented in Chapter 5.

4.8.1 R* Computed Torque controller and RRT trajectory

First a trajectory following experiment using only the wrist joint of the robot is presented. A 1.5 Kg load is mounted on the end-effector, and the task is to bring it from the bottom position ($q = -\pi$) to the up-right position ($q = 0$) using as little torques as possible. This corresponds to the inverted pendulum swing-up problem. An RRT trajectory planning algorithm is used to search for a feasible low torque trajectory reaching the goal, see Fig. 4-29. Then the R* Computed Torque Controller is used to track the reference trajectory. The experimental results are shown in Fig. 4-30. Results show that the robot is using its 1:23 gear-ratio to accumulate kinetic energy by swinging the arm link back and forth. Also the R* controller selects the 1:474 gear-ratio automatically to attenuate the load dynamics, when the actuator has to force the robot to stay with the trajectory. Interestingly, the reference trajectory

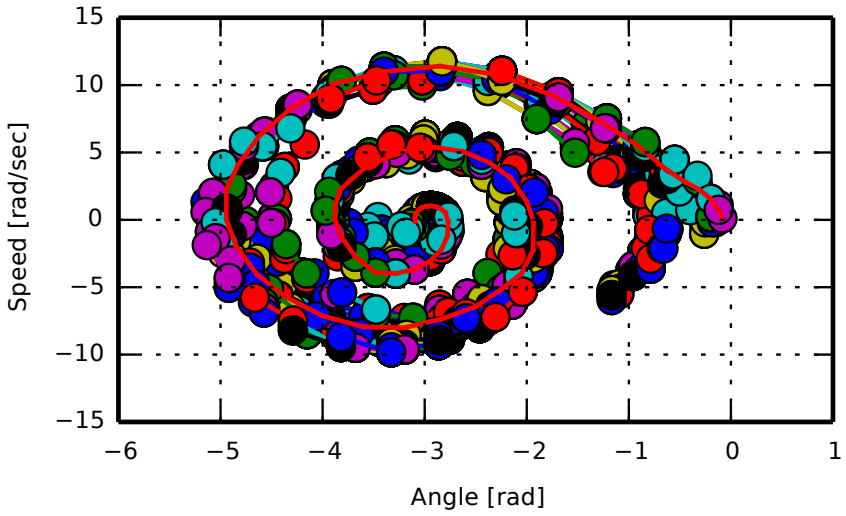


Figure 4-29: RRT algorithm searching for a low torque solution

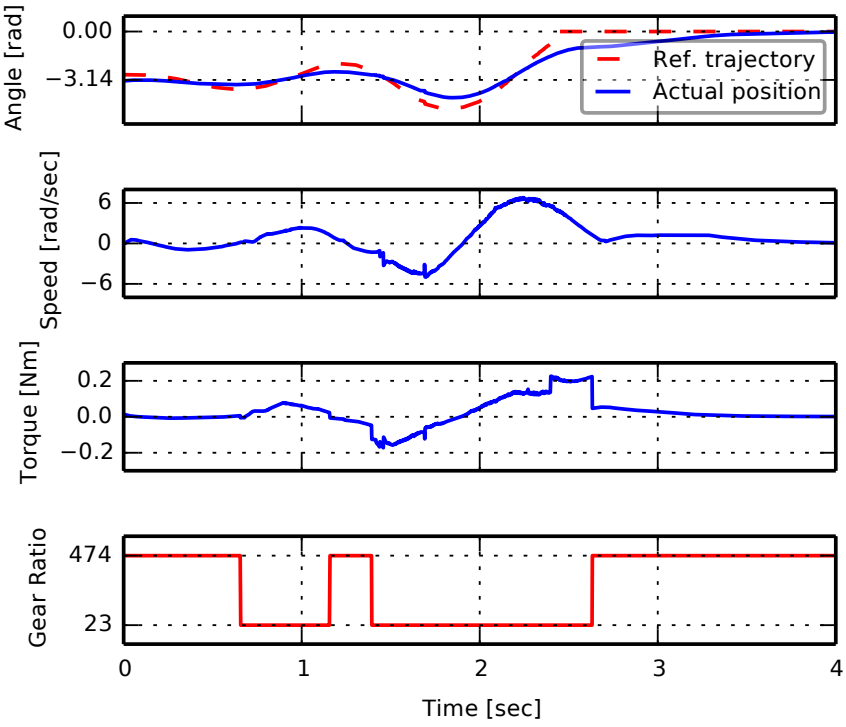


Figure 4-30: Experimental trajectory and control inputs

was planned so the robot would accumulate enough kinetic energy to swing straight up with the last swing. However, in the experiment, the dissipative forces are greater than anticipated by the planner, and the last swing is too small (the robot almost stop at $q = -0.9$ at $t = 2.6$ in Fig. 4-30). Then, the R* controller automatically engage the large 1:474 gear-ratio, to continue converging on the desired trajectory with much smaller torques than those required if keeping using the 1:23 gear-ratio in this situation (no momentum and a large gravitational force to overpower). This illustrates that including the gear-ratio selection in the feedback loop also increases the robustness of the system. Without the 1:474 gear-ratio option, tracking would have failed as the computed torque with 1:23 in this situation was greater than the maximum allowable motor torque.

4.8.2 R* Sliding Mode controller

Fig. 4-31 shows four additional experiments with the wrist demonstrating how disturbance rejection can be improved by using the R* Sliding Mode controller. Here the controller is only given a simple fixed point-target in all cases. First, when a low uncertainty bound is given to the controller, the robot can reach its target when unloaded (a) but failed when an unknown (to the controller) 0.4 Kg load is added to the end-effector (b). However, when a larger uncertainty bound is given, the robot can reach its target in both cases, unloaded (c) and loaded (d).

Note that the discontinuous torque required to guarantee convergence despite disturbances is bigger for the case when the disturbance bound is increased, and is greatly reduced when using the large gear-ratio at low speeds (since the required discontinuous gain is inversely proportional to the gear-ratio). In a practical implementation, smoothing techniques should be implemented to avoid exciting the unmodeled high-frequency modes with the torque chattering.

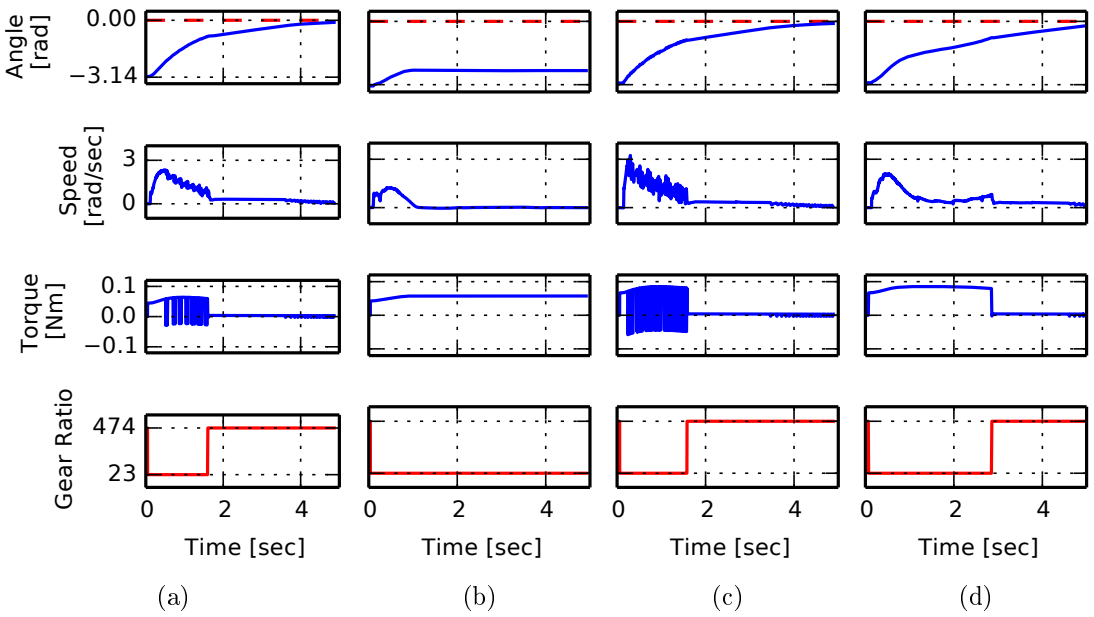


Figure 4-31: Experiments with the R* Sliding Mode controller

4.8.3 2-DoF experiments

Fig. 4-32 and Fig. 4-33 show an experiment using 2-DoF, the wrist joint and the elbow joint of the DSDM-Arm. The goal is a fixed joint configuration, and the R* Sliding Mode controller, including the Rollout gear-selection scheme, is used. Results shows success in tracking the goal for both DoF. Also, it is possible to observe that the downshift at $t = 2.6$ allows for a drastic reduction of the necessary discontinuous torque to guarantee convergence, illustrating the advantage of isolating the motor from the external load with a large reduction ratio in some situations.

Video of experiments and simulations presented in this chapter are available at the following links: https://youtu.be/-jo6dzvtfY4 and https://youtu.be/rx6dt8TYXus.

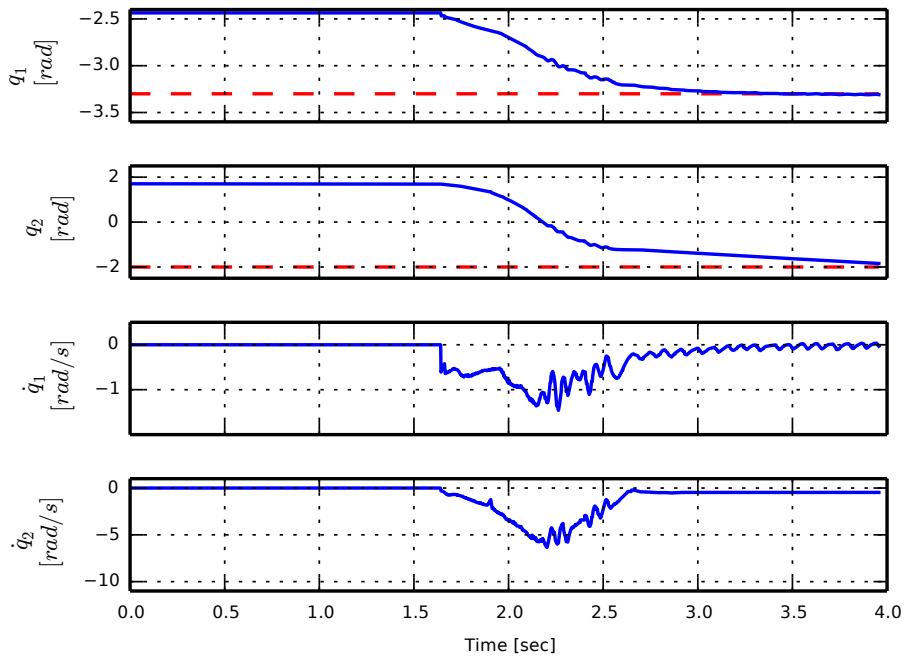


Figure 4-32: R* Sliding Mode controller tracking a target with 2 DoF: trajectory

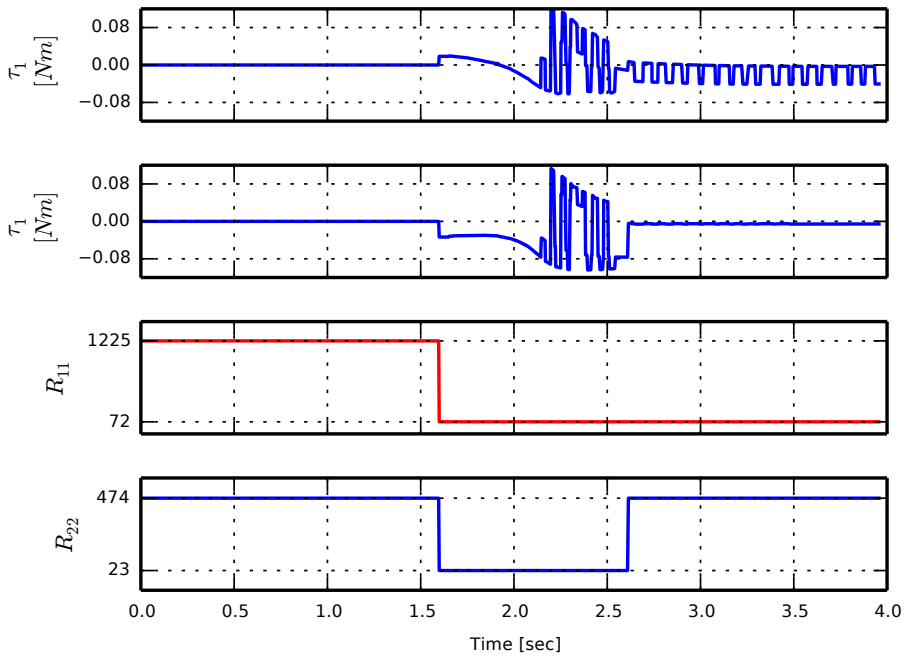


Figure 4-33: R* Sliding Mode controller tracking a target with 2 DoF: control inputs

4.9 Summary

In this Chapter, feedback laws for controlling both the torques and gear-ratios of robotic systems equipped with variable transmissions are proposed. A simple dynamic model is proposed, and analytical solutions are derived for the optimal gear-ratios on a known trajectory. The approach is extended to trajectory tracking control schemes (R* Computed Torque and R* Sliding Mode), which can guarantee convergence on a reference trajectory and execute locally optimal gear-ratios based on state feedback. An approach (Rollout gear-selection) is also proposed to inhibit fast gear-shifting by optimizing over a receding time-horizon. An alternative computational technique (Value iteration) is also explored to generate global control policies for both torque and gear-ratios for simple systems. Simulation and experimental results demonstrate the effectiveness of the proposed controllers.

4.10 Potential directions of further development

Here are a few possible axis of further development:

- Improving the proposed control approaches:
  - Using reinforcement learning to learn good gear-selection policies;
  - Explore adaptive controllers;
  - Decentralizing the gear-ratios selection decisions.
- Explore more specific applications:
  - Legged locomotion or manipulation including contacts;
  - Interaction tasks requiring a wide range of impedance.
- Open research questions not addressed:
  - Controlling under-actuated robots using VGA;
  - Efficient trajectory optimization for robots using VGA.

Chapter 5

The DSDM Lightweight Arm

Mechanical Design, Control and Software Architecture

"What I cannot create, I do not understand."

– Richard Phillips Feynman

This chapter present a novel 3-DoF robotic arm prototype using DSDM actuators, see Fig. 5-1 and Fig. 5-2. The mechanical design of the DSDM actuators and the robotic arm is discussed, as well as the control and software implementation.

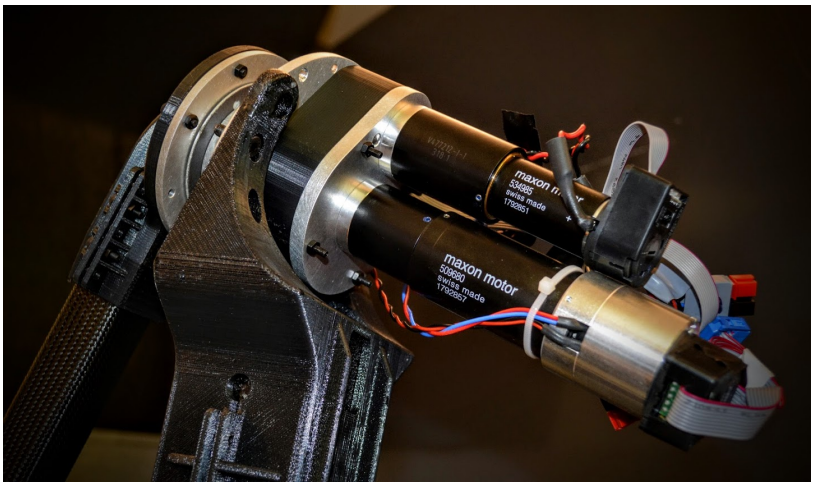


Figure 5-1: Second joint of the DSDM-Arm

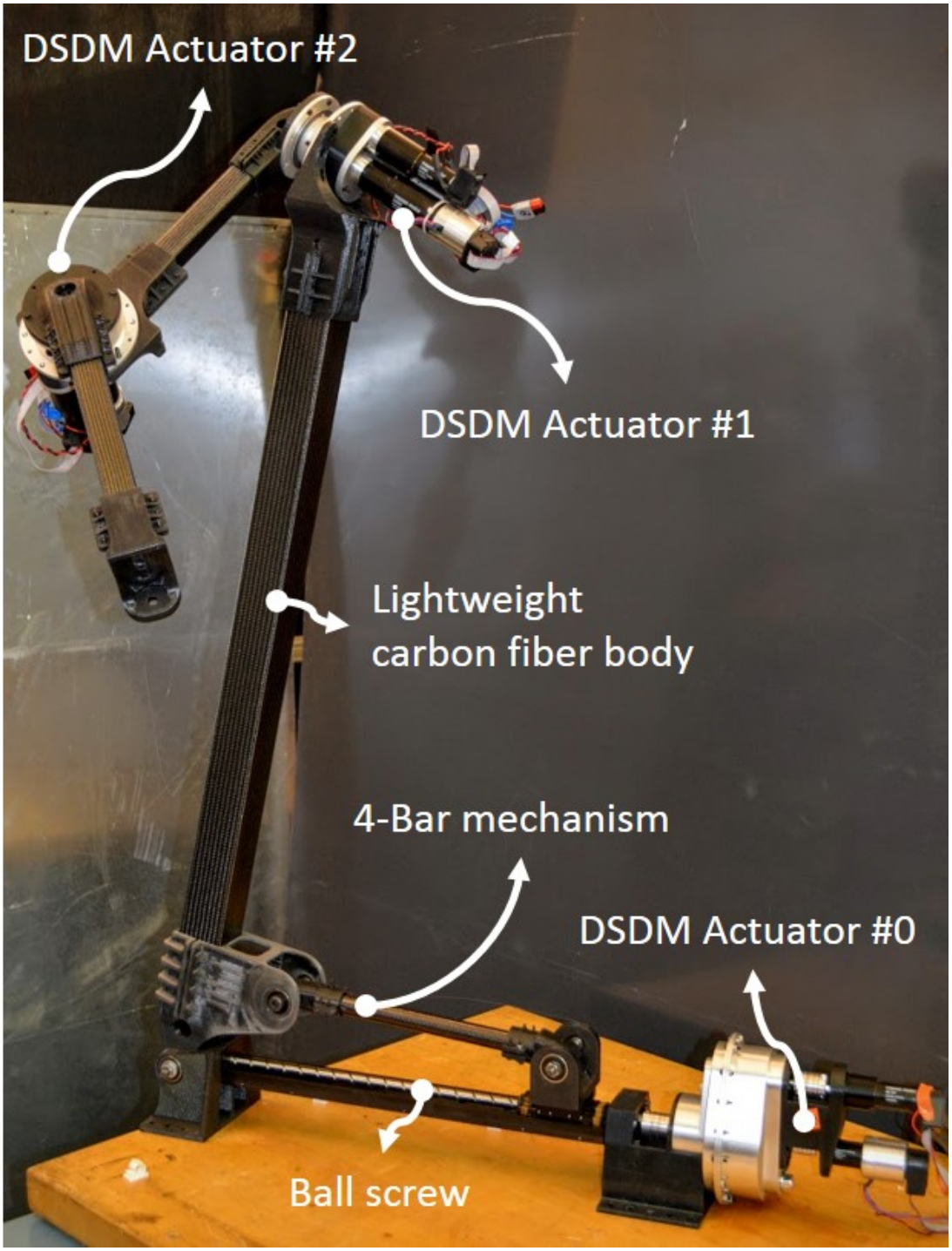


Figure 5-2: DSDM-Arm: 3-DoF custom arm using 3 DSDM actuators

5.1 Mechanical Design

This section describes the mechanical design of the arm. The goal was to develop a research platform to validate experimentally the ideas proposed in this thesis, but also to demonstrate the advantages of robotic systems with variable gear-ratio actuators. This arm is thus design to be light-weight compared to robotic arm of similar size, maximum speeds and forces.

5.1.1 DSDM actuator design

Three actuator prototypes were developed for the shoulder, elbow and wrist DoF of the arm, with different mechanical advantages. The shoulder actuator is designed to drive a ballscrew, for a large efficient reduction, and the others actuators are embedded into revolute joints.

3-port differential gear-box

One of the main design challenge arising from the DSDM architecture, is the mechanical implementation of the differential junction between the two motors and the output. In a car powertrain the differential (required to allow a single motor to transmit torque to two wheels rotating at different velocities) is typically implemented with bevel gears. The approach taken here is different, a planetary set of gear is used, where the ring-gear (typically fixed) is mounted on bearing and connected to a parallel shaft. The connection to the parallel shaft is done with a stage of spur gears, with external gear teeth on the ring-gear assembly and another spur gear on the parallel shaft, see Fig. 5-3a. This configuration allows for all the shafts in the transmission and the motors to be parallel, which simplify the design. In all prototypes, the correspondence between planetary ports and inputs/outputs is the same, and described by Table 5.1. This correspondence is picked to match the kinematic relationships arising from sizing constraints in the gearing. From eq. (3.1), with a typical ratio in the planetary of $N = 3$ (planet-gear size over internal ring-gear size), and a reduction ratio of $r_2 = 4$ for the parallel shaft connection (smaller r_2 would

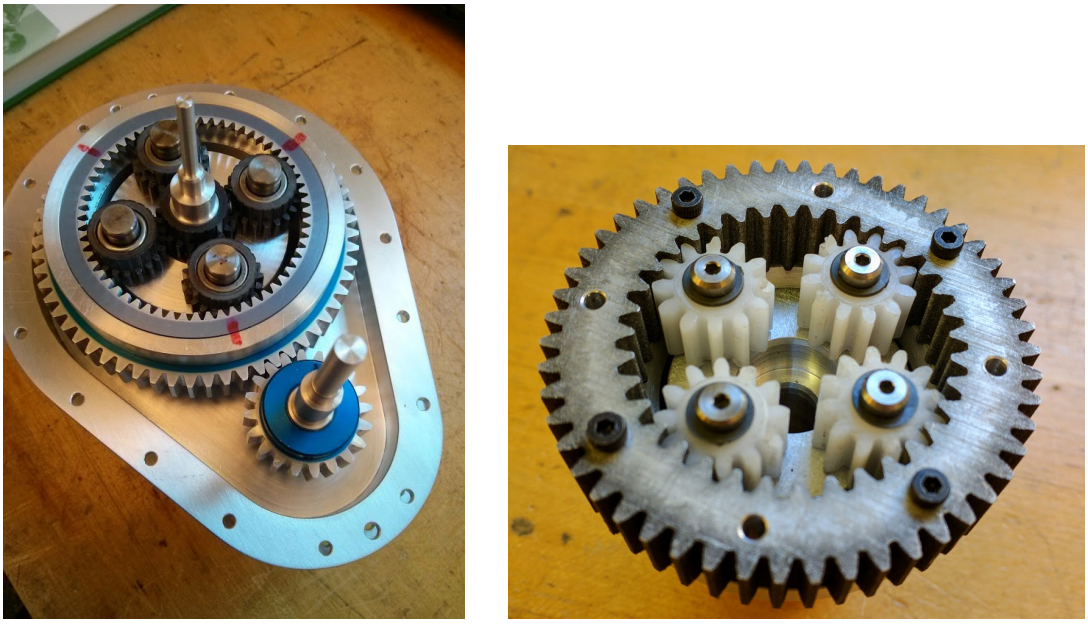
require a large distance between parallel shafts leading to larger transmission volume and larger r_2 is difficult to achieve in a single spur-gear stage), it leads to:

$$N \approx 3 \quad r_2 \approx 4 \quad \Rightarrow \quad R_1 = N + 1 \approx 4 \quad R_2 = r_2 \frac{N+1}{N} \approx 5 \tag{5.1}$$

Hence, the largest reduction through the differential gearing is assigned to M2 and the smallest reduction to M1. This choice is also motivated by the fact that the path going through the ring-gear and the additional stage, would lead to more friction and inertia which is less a concern for high-force mode than for high-speed mode.

Rotating assembly	Role
Planet carrier assembly	Actuator output
Parallel shaft (connected to the ring-gear)	Motor M2 input (high-force)
Sun gear shaft	Motor M1 input (high-speed)

Table 5.1: Planetary gearing inputs and outputs



(a) Linear Actuator (b) Revolute Joint

Figure 5-3: Differential gear-box implemented with a planetary

Fig. 5-3a shows the designed differential gearing for the linear actuator prototype and Fig. 5-3b shows the designed differential gearing for the revolute actuators. The first design (a) was using only off-the-shelf gears, which led to a big assembly.

For the second design (b) a lot of effort was put into downsizing the assembly. For instance, custom ring gears with both internal and external gear teeth were designed, to minimize the diameter of the ring-gear assembly.

Brake

The second challenging mechanical component in a DSDM, is the brake. As discussed in Chapter, 3 control schemes can be used to bring M1 velocity to zero before engaging the brake, hence the brake only has to be a locking mechanism and does not have to be able to dissipate power. However, during high-force mode, large holding torques must be sustained. From eq. (3.2), the holding torque requirement can be computed in term of desired maximum output force during high-force mode, or by the maximum M2 motor torque:

$$\tau_{brake} = \frac{\max\left[\tau_{output}\right]}{R_1} = \frac{R_2}{R_1}\max\left[\tau_2\right] \tag{5.2}$$

Hence, the design of the brake is coupled with the desired ratio between R_1 and R_2. If the high-force mode gear-ratio R_2 is 10 times greater than the high-speed mode gear-ratio R_1, than the brake must be able to hold torques 10 times greater than M2 maximum output torque.

Because it allows for simpler and modular designs, it was decided to use off-the-shelf *Maxon* motor brakes that can be mounted directly on a motor assembly. All actuator designs use the *Maxon* motor brake AB-28, which have a holding torque capability of 0.4 Nm and weight 51 g.

Revolute joint actuators

Fig. 5-4 shows the prototype for a revolute DSDM actuator. The elbow and wrist actuator have the same design with the exception of using different gear-head for the motors, leading to overall different gear-ratios R_1 and R_2. The design for the revolute actuator consists of a custom housing holding both the planetary differential and support bearings for the output. Discrete *Maxon* motors with gear-heads of the

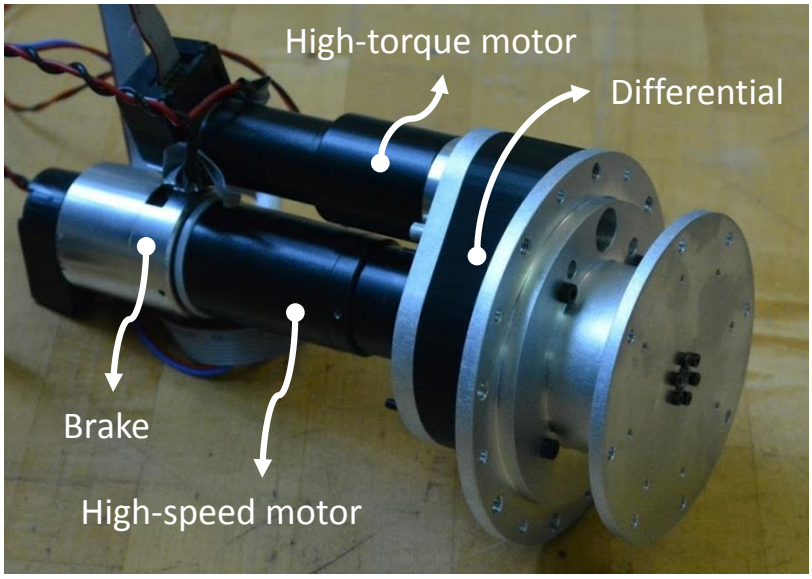


Figure 5-4: Revolute joint prototype with DSDM actuation

series GP32 can be attached to the back of the gear-box. It is thus possible to attach a wide-range of motor, from 20 watts to 200 watts, and with a wide range of additional gear-head reduction. Fig. 5-5 shows the internal architecture of the system with a section view of the CAD model, and Fig. 5-6 shows all the internal parts of the actuator assembly.

A *Maxon* motor of the series RE-25, with maximum continuous power of 20 watts and torque 0.03 Nm, is used for M2, and motor of the series RE-35, with maximum continuous power of 90 watts and torque 0.1 Nm, is used for M1. The revolute actuator prototypes use additional gear-head reduction for both motor, to increase both value of total reduction R_1 and R_2, to reach useful range of torque and speeds, as illustrated at Table 5.2.

Table 5.2: Specifications of revolute actuator prototypes

Role	R_1	R_2	M1 power	M2 power	Max. Torque	Max. Velocity
	$\frac{w_1}{w_o}$	$\frac{w_2}{w_o}$	Watts	Watts	Nm	RPM
Wrist	23	474	100	20	14	220
Elbow	72	1225	100	20	37	70

Note that the ratio R_2/R_1 is always keep at a factor of about 20, to match the capability of the brake according to eq. (5.2). Also, the specifications of maximum

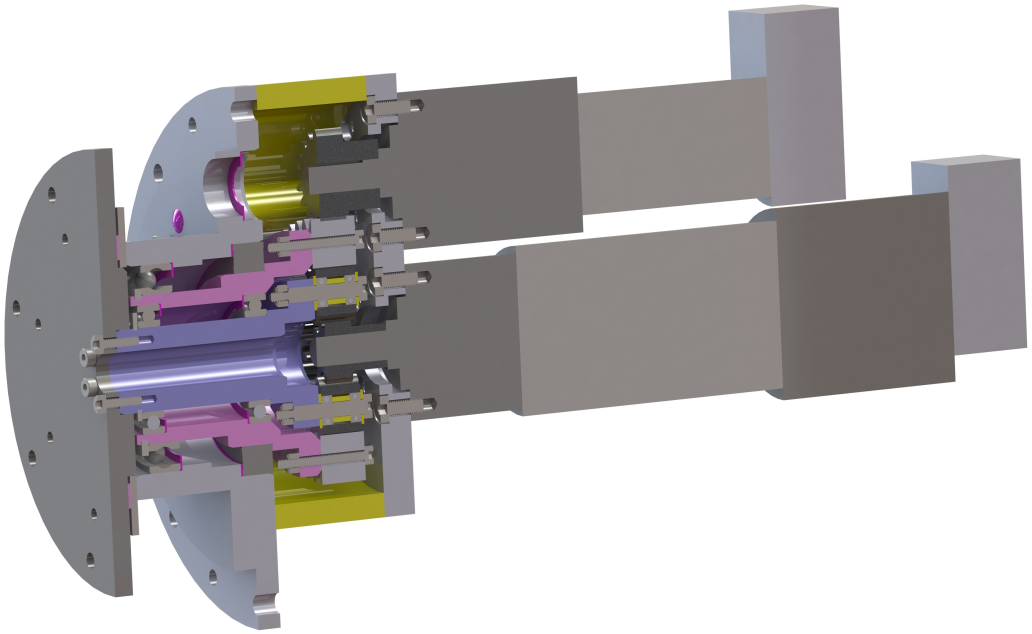


Figure 5-5: Section view of the CAD model of the revolute actuator prototype

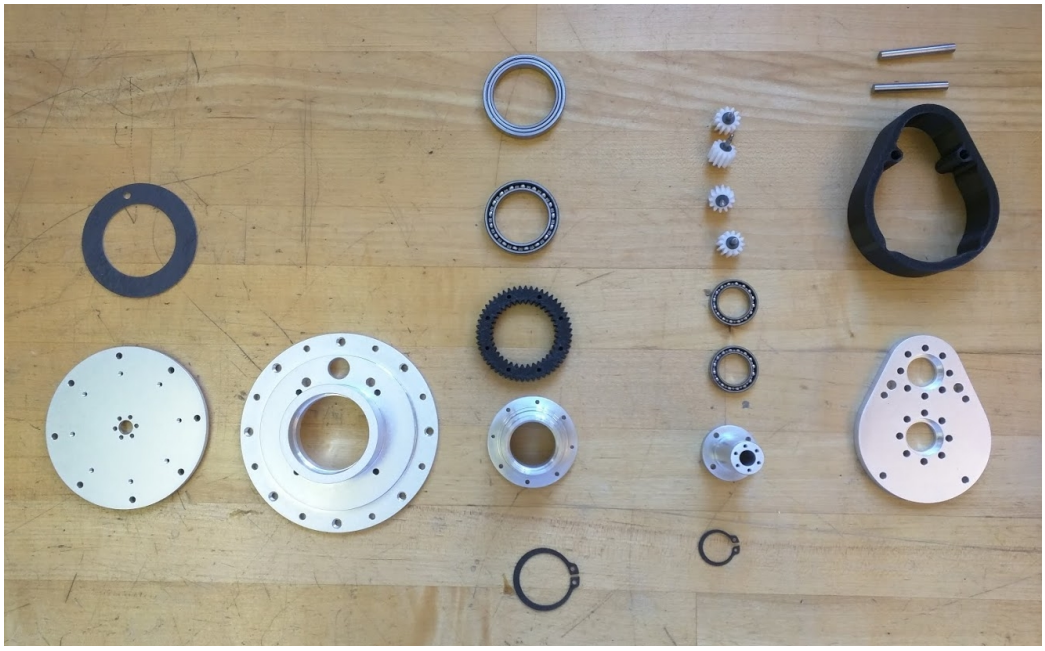


Figure 5-6: Internal components of the revolute actuator prototype

torque, are given in term of very conservative continuous value advertised by *Maxon*. Better performance could be obtained in term of peak torque during short time periods. The whole assembly of embedded DSDM actuator and support bearings for the revolute joint weight about 1.5 Kg. About half of this weight is due to *Maxon* motor assemblies (motors, gear-heads and brakes) and the other half is the custom built transmission and joint support. This value should not be taken as a state-of-art comparison reference to other actuation technologies since 1) no optimization for weight has been conducted, 2) industrial DC *Maxon* motor are don't have the best available power density and 3) the design was focus on ease of implementation and modularity.

Linear Actuator

To achieve the large reduction needed for the shoulder actuator of the robot, while keeping the mechanism back-drivable during high-speed mode, a large-lead ballscrew linear stage is used. Fig. 5-7 shows the linear actuator assembly, and Fig. 5-8 shows the internal components.

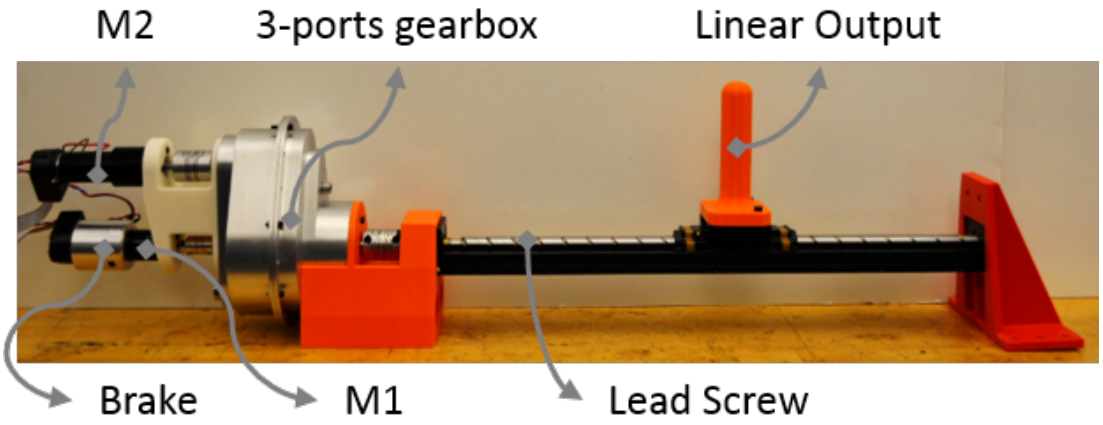


Figure 5-7: Linear actuator assembly in a preliminary test configuration

The linear actuator assembly was initially designed as an experimental test bench for the DSDM technology [14]. Unlike the revolute actuators, the transmission is sealed and lubricated with oil and flexible coupling are used to connect all the com-

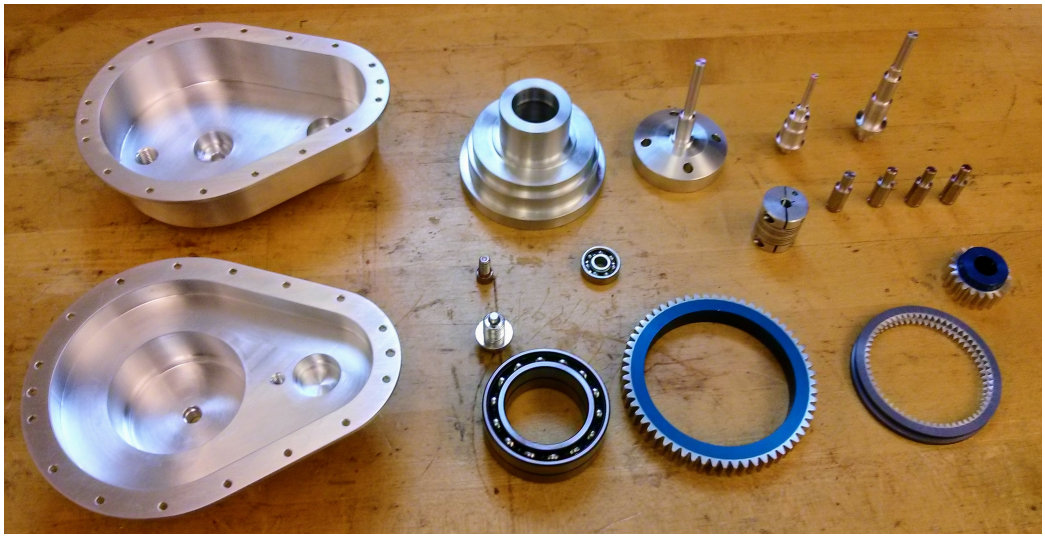


Figure 5-8: Internal components of the DSDM linear actuator

ponents. The linear actuator is thus more heavy duty, bigger and heavier than the revolute actuators. However, the linear actuator assembly is fixed to the ground and not a moving part of the DSDM-Arm, hence drawback of its weight and volume are limited.

Two *Maxon* RE-25 motors capable of continuous operation of 20 Watts and 0.03 Nm are used for both M1 and M2. Only M2 is equipped with a gear-head for the linear actuator; the reduction provided by the ball-screw and the differential are sufficient for the high-speed mode. Overall gear-ratios in the design and resulting specifications are given at Table. 5.3.

Table 5.3: Specifications of the linear actuator prototype

R_1	R_2	Lead	M1 power	M2 power	Max. Force	Max. Velocity
$\frac{w_1}{w_o}$	$\frac{w_2}{w_o}$	mm/rev	Watts	Watts	N	m/s
4	72	20	20	20	600	0.7

It is interesting to note that any commercially available linear actuator matching similar specifications of force and speed are much bigger and heavier than the presented linear actuator prototype, even with this un-optimized design. With a single gear-ratio, commercial actuator would need to use a DC motor of roughly $0.7\,\frac{m}{s} \times 600N = 0.4kW$, almost 2/3 of a horsepower. A *Maxon* motor (in the same

DC category) weight over 2 kg to meet those specifications, compared to two 20 W motor weighting each about 130g. Of course, the linear actuator with a single big motor would be much more powerful than the presented linear actuator, but such power might be unnecessary. There is a need for actuators that can be both fast and strong, not necessary powerful, and very lightweight.

5.1.2 Arm design

The DSDM-arm is built using very lightweight square tubing of carbon fiber. 3D printed plastic (ABS) parts are designed to make the junction between revolute joint assemblies and the tubes. The revolute joint assemblies are joined to printed parts with a bolt pattern, and printed parts are simply clamped on the square tubes. This allows for very quick reconfiguration of the arm, tubes of different lengths can be used and revolute joints can be mounted at different angles on the tubes. The carbon fiber tube between the shoulder and the elbow has a cross section of 2"x2" while the tubes linking elbow-wirst and wrist-end-effector have a 1"x1" cross section.

Shoulder 4-bar mechanism

A 4-bar mechanism is designed to transmit the linear actuator motion to the revolute shoulder joint, see Fig. 5-9. This combination of ballscrew with a 4-bar mechanism allows for a very large mechanical advantage to be achieved, and with very good transmission efficiency.

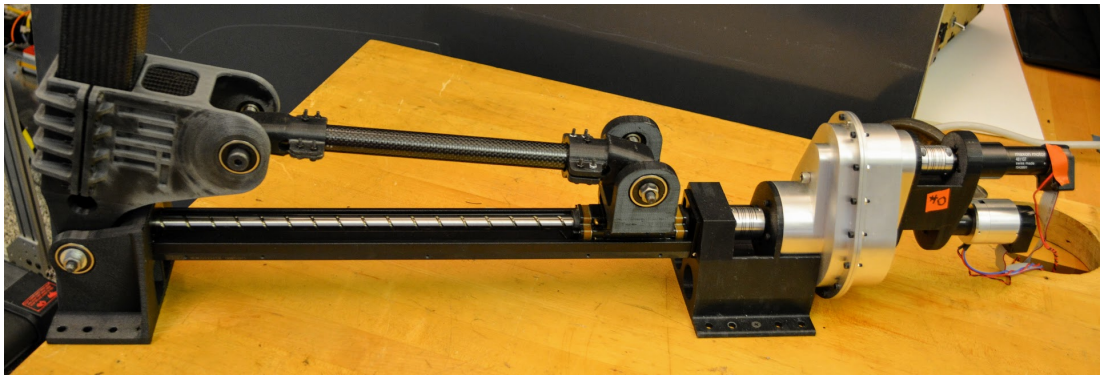


Figure 5-9: Shoulder 4-bar mechanism

The geometry of the 4-bar mechanism has been designed to achieve the desired range of angles for the shoulder, and keeping the kinematic relationship, between linear displacement and shoulder angle, as linear as possible. Fig. 5-10 illustrates the kinematic relationship of the designed mechanism. Note that in the software controlling the arm, this kinematic relationship is computed explicitly for the inverse-kinematic but the forward kinematic is approximated with a numerical interpolation (when computing the shoulder position based on encoder measurements of the linear actuator). The shoulder forward kinematic has a unique solution for the range of physically possible linear actuator displacement.

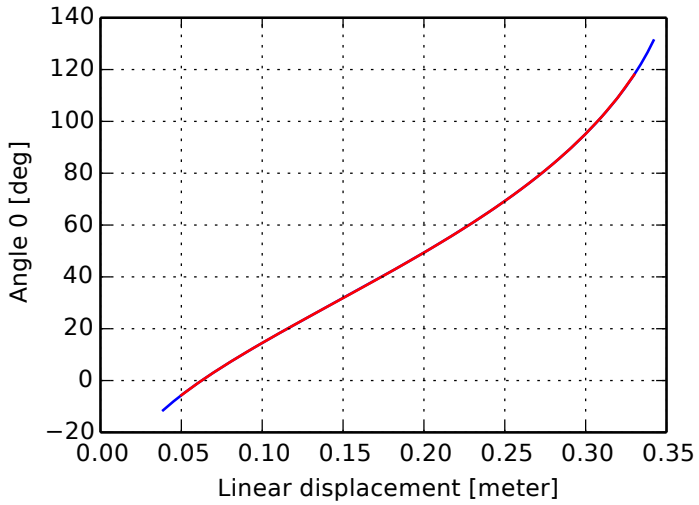


Figure 5-10: Shoulder 4-bar mechanism kinematic

The mechanical advantage between linear motion and shoulder rotation is thus on average about 360 deg/m, or a ratio of 50:1 from ballscrew rotation to shoulder rotation. This correspond to average total reduction ratios of $R_1 = 200$ and $R_2 = 3600$. Surprisingly, the shoulder joint mechanism is backdrivable, even during high-force mode. This illustrates the efficiency of ballscrew-based reduction mechanisms. During high-speed mode, a human can easily move the first joint of the robot with almost no resistance. During high-force mode, the reflected inertia is very large, but a human can still make the shoulder joint move very slowly by pushing hard (with open circuit for the motor). However, if there is a small damping force at M2 motor

shaft (by closing the motor electric circuit for instance), than back-driving the system is almost impossible.

Arm Specifications

Table 5.4 shows the arm specifications at the joints in term of force, speed and reflected motor inertia. Table 5.5, shows the DSDM arm joint specifications transposed to end-effector space where they are more meaningful. For doing so, a configuration where all link are aligned and tube lengths of 0.5 m, 0.25 m and 0.25 m is assumed, see Fig. 5-11. Then the maximum force, maximum speed and impedance at the end-point, due to each actuator taken independently, are computed. All in all, with the designed reduction ratios, during high-speed mode the end-point inertia due to reflected motors inertia is negligible and speed can reach multiple meters per second. With the high-force mode, the end-point force capability reach 50 N (limited by the wrist) even in this most disadvantageous fully extended configuration, with the end-effector at arms' length.

Table 5.4: DSDM-Arm joint specifications

	Range deg	Reduction		Max. Velocity RPM		Max. Torque Nm		Inertia kg m^2	
		HF	HS	HF	HS	HF	HS	HF	HS
Wrist	∞	474:1	23:1	10	220	14	2	0.22	0.004
Elbow	∞	1225:1	72:1	4	70	37	7	1.5	0.04
Shoulder	120	3600:1	200:1	1	25	108	6	13	0.04

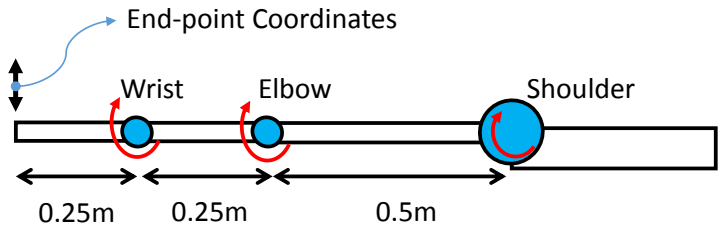


Figure 5-11: Arm configuration used to compute end-point specifications

Note that the DSDM-Arm is strong enough to sustain its own weight in all configurations when using high-force (HF) modes. However, this is not the case when

Table 5.5: DSDM-Arm end-point specifications with the configuration illustrated at Fig. 5-11

	Max. Velocity m/s		Max. Force N		Reflected Mass kg	
	HF	HS	HF	HS	HF	HS
Wrist	0.25	5	56	8	3.5	0.06
Elbow	0.2	3.5	74	14	6	0.16
Shoulder	0.1	2.5	108	6	13	0.04

using the actuators in high-speed (HS) mode.

5.1.3 Limitations and recommendations for improvements

The main limitations of the DSDM-Arm are due to 1) backlash in the actuator gearing and 2) compliance in the structure. For the ease of implementation, the custom built actuator transmissions use only standard spur gears. The flaw of this design is that the output has a backlash of a few degrees. While it was not a critical issue for demonstrating the proposed control scheme of this thesis, it would be problematic for using the DSDM-Arm platform in task involving precise positioning of the end-effector. The second issue is the arm compliance due to 3D printed plastic parts used to connect the joints to the carbon fiber tubes.

Regarding the backlash in the gearing, this could be avoided by spending more engineering effort to make thorough precision gear-box design. For the compliance in the structure, using metal parts could solve the issue but would lead to a heavier robotic system. Instead, it would be interesting to use 3D printed parts reinforced with carbon fiber, such as the *Markforged* technology.

5.2 Control and Software Architecture

This section discusses implementation of the control algorithms for the DSDM-Arm.

5.2.1 Global architecture

Fig. 5-12 illustrates the hierarchical control architecture for the DSDM-Arm. At the very high-level an operator gives commands to the system using a wireless *xbox-360* controller, and at the very low level, pwm signal are sent to motor power-electronic circuits. The physical platforms include a desktop computer running Ubuntu 14.04 and ROS [50], and multiple micro-controllers.

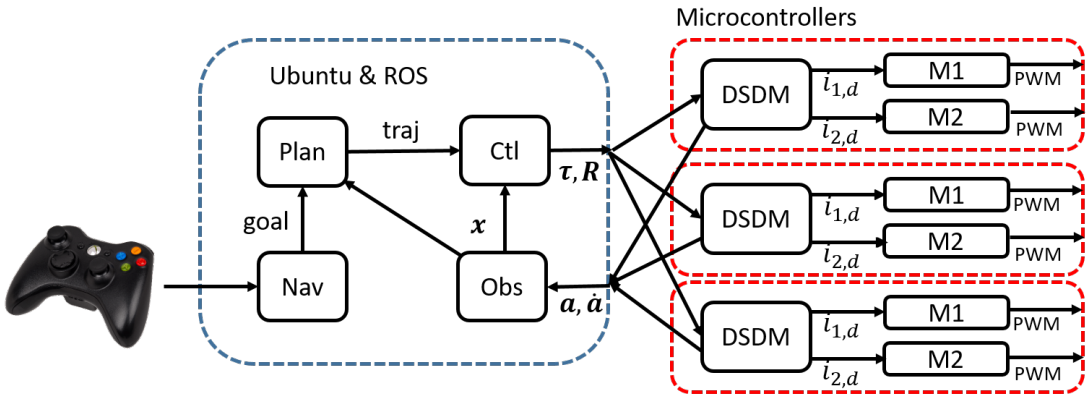


Figure 5-12: Control software architecture

The high-level motion planning algorithm, the centralized robot controller is running on the desktop. Each motor as its own micro-controller handling the low-level current control loop. The plan for the DSDM controllers, was to decentralize them on one individual micro-controllers for each actuators. However, at the time of writing these lines, all three DSDM controllers are programs running on the desktop, which communicates directly with the six motor boards. Communication between the desktop and motor micro-controllers is done over USB connections.

5.2.2 ROS architecture

Fig. 5-13 and 5-14 illustrates the software architecture in ROS, those figures were generated directly from ROS using the *rqt_ graph* command. Each ellipse, nodes using the ROS nomenclature, represent an independent program, and arrows represent communication pipelines in between those programs, topics using the ROS nomenclature. All programs were written in *python* with the exception of the FlexSEA drivers, handling the communication with motor micro-controllers, which are written in *C++*.

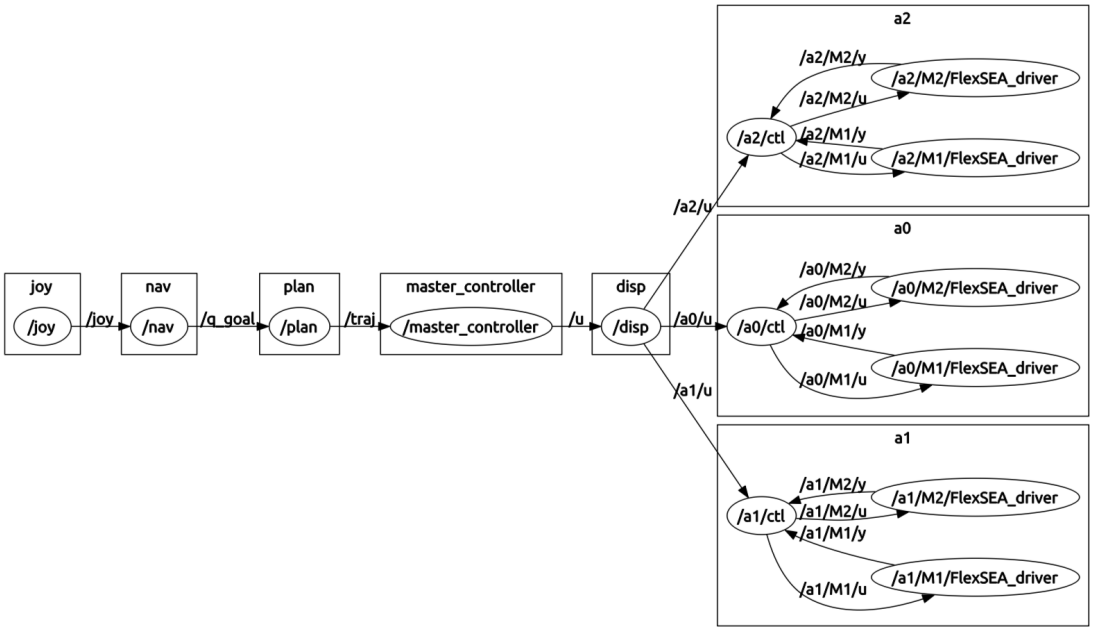


Figure 5-13: ROS architecture for the full robot (feedback connections are omitted)

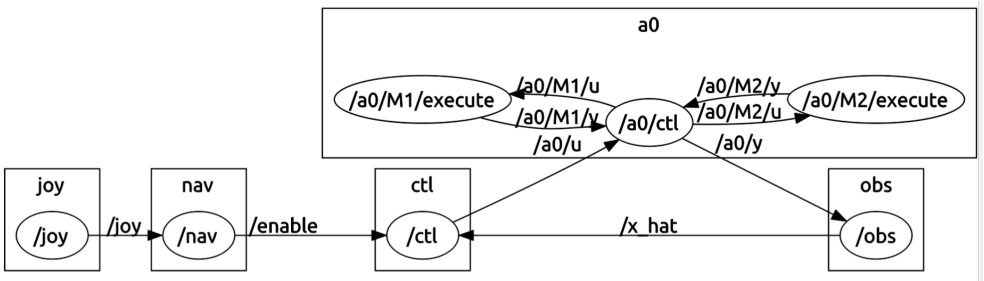


Figure 5-14: ROS architecture for controlling a single DSDM actuator directly

5.2.3 Navigation

The *joy* node read button states of the wireless controller, and broadcast them when button state changes. This program is available as an open-source ROS package. The *nav* node then maps buttons to operating modes and desired goals. Multiple operating modes are available: sending a configuration goal to the planner, sending a reference directly to the robot controller, or manually controlling actuator torques and gear-ratios.

5.2.4 Trajectory planning

The *plan* node implement the RRT algorithm, as described in sec. 4.5, to generate feasible trajectories. When receiving a new configuration goal, the node reads the current states, search for a feasible trajectory, and then broadcast the reference trajectory solution. Execution time of the search is stochastic, but is on the order of about 1 sec for 1-DoF systems and 10 sec for 2-DoF systems, with the custom implementation.

5.2.5 State feedback

The FlexSEA driver nodes are programed to continuously request and receive encoder measurements from motor boards at a rate of 500 Hz. The DSDM controller nodes then read those values, process them (kinematic relationship based on gear-ratios and filtered differentiation to compute velocities), and broadcast position and velocity of actuator output coordinates. The *obs* node then read all actuator measurements and transposes them to joint coordinates. This step is trivial except for the 4-bar mechanism of the shoulder joint for which an interpolation function synthesized offline is used.

5.2.6 Robot controller

The master robot controller, loads a reference trajectory, and compute actuator torques and gear-ratios in closed-loop based on state feedback at the rate of 500

Hz. Multiple control policies from chapter 4 are implemented:

- R* Computed Torque controller
- R* Sliding Mode controller
- Rollout gear-selection

For controlling only 1-DoF with two gear-ratios options, the Rollout gear-selection can run with a time horizon of 1-2 sec without slowing down the 500 Hz rate of the controller. However, when running the controller for 2-DoF with four gear-ratios options, the predictive simulations had to be simplified by using very rough integration steps for the scheme to run at 500 Hz. This should not be taken as a computational limit of the approach since here non-optimized inefficient *python* code was used.

5.2.7 DSDM actuator controllers

DSDM controllers are currently implemented as ROS nodes. They receive a torque and a gear-ratio command from the master robot controller and send current set-points and brake command to the motor drivers, also at 500 Hz. DSDM controller nodes implement the control schemes described in Chapter 3.

5.2.8 Motor drivers

The motor are controlled by open-source FlexSEA Execute motor board [10]. Those boards handle low-level high-bandwidth current loops and encoder signal processing. They also include a circuit to control a brake. All those functions are made available from the FlexSEA driver nodes. Here the driver are set to request, receive and broadcast all sensor information from the Execute boards constantly at 500 Hz, which set the tempo for the feedback loops.

5.2.9 Limitations and recommendations for improvements

The initial control implementation for DSDM actuator prototypes used a *NI compact rio* where feedback loops were running at very high-sampling rate on a FPGA and

a real-time micro-controller [14]. This type of system was however poorly suited to scale to multi-DoF systems, and implementing complex planning algorithms. Hence, for the DSDM-Arm the architecture was design to: easily scale to multiple DoF, make possible the use of high-level programming languages and facilitate connections with motion planning algorithms. However this came at the cost of losing some control on the low-level control loops implementations. The performance of actuator-level DSDM controllers has decreased with the new implementation. To improve the system, DSDM controllers should be implemented directly on micro-controllers, handling all actuator-level functions, at a higher sampling rate. One implementation issue that could also be improved, is that velocity measurements, at low speeds in high-speed mode, are very noisy because of resolution problems. An encoder with improved resolution, or direct an angular velocity sensor should be used to improve the performance.

Chapter 6

Conclusion

This thesis explored the idea of robotic systems using actuators with variable transmissions, i.e. where the reduction ratio can be dynamically changed online. Although variable transmissions are used extensively in vehicle powertrains, this concept is highly under-explored and under-exploited in the field of robotics, despite huge potential gains as demonstrated in this thesis. Variable gear-ratios actuators can be used not merely for increasing maximum torque and speed, but also to significantly advantageously alter the dynamic properties of robots including load sensitivity, robustness and backdrivability. This thesis main contributions are 1) DSDM actuators: a solution to make gear-shifting transitions adapted to a wide range of robotic tasks, and 2) Control approaches to synthesized optimal closed-loop gear-ratios selection policies, for a very generic class of robotic systems using variable transmissions.

Chapter 2 briefly discussed manufacturing applications, where actuators must meet challenging requirements, which could hugely benefit from the proposed technologies developed in this thesis. Chapter 3 presented the DSDM actuation technology that can change its effective reduction ratio, between a small reduction and a very large reduction, quickly and seamlessly even in highly dynamic situations. Chapter 4 explored the idea of closed-loop selection of gear-ratios for multi-DoF robotic systems, and proposed control schemes that leverage all the advantages offered by variable transmissions. Analytical optimal solutions, for a class of robotic systems, and guarantees in terms of stability and chattering behavior are also derived.

Chapter 5 presented the DSDM-Arm, a novel lightweight robotic system using three DSDM actuators, which was used for experimental validation of all the proposed control schemes and also to demonstrate the advantages of robotic systems equipped with variable gear-ratio actuators. Multiple experiments with the DSDM actuators demonstrated the salient features and the ability of the DSDM technology to change gear ratio quickly and seamlessly even in very dynamic situations, including impacts. Simulations and experiments with the DSDM-Arm were presented and demonstrated that actively changing gear ratios using the proposed control algorithms can lead to an order-of-magnitude reduction of necessary motor torque and power.

The author would like encourage all researchers in the field of robotics, to question the single-gear electric-motor actuation paradigm, and envisioning variable transmissions for applications that require speed and force in a small package. The field is appealing from both an engineering and a scientific perspective. On one hand, it is a very practical solution to relevant power transmission problems: cars, bicycles, electric drills, etc., use multiple speed transmissions. On the other hand, it makes the problem of controlling non-linear robotic systems: even more non-linear and hybrid. Exciting research questions are raised that are both challenging and worth solving, and the author hopes you joint him in visiting this realm where there be dragons.

Appendix A

Robot Dynamics Framework

In this appendix, the nomenclature and the mathematical equations, used to represent the behavior of robotic systems in thesis, are presented. Table A.1 define all the variables.

A.1 Equations of motions

The general form of the equations of motion of robotic systems (interconnected rigid body driven by actuators) is:

$$H(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = B(\boldsymbol{q})\boldsymbol{\tau} \tag{A.1}$$

where $\boldsymbol{q}$ is the generalized coordinates vector, H is the inertia matrix, C is the Coriolis/centrifugal force matrix, D is a damping matrix, $\boldsymbol{g}$ is the gravitational forces vector and B is a matrix mapping motor torques $\boldsymbol{\tau}$ into generalized forces. On occasion, dependence notation is dropped and $\boldsymbol{c}$ is used to represent all state dependent forces, leading to the short form:

$$H\ddot{\boldsymbol{q}} + \boldsymbol{c} = B\boldsymbol{\tau} \tag{A.2}$$

Table A.1: Nomenclature

		Scalars	
n	:	number of DoF	
m	:	number of actuators	
c	:	number of contact constraints	
o	:	number of end-effector coordinates	
l	:	number of discrete operating modes	
i	:	index for DoF	
k	:	index for the operating mode	
		Vectors	
$\boldsymbol{\tau}$	:	Electromagnetic motor torques	m
$\boldsymbol{q}$	:	Joint coordinates position vector	n
$\boldsymbol{x}$	:	Dynamic state vector $[\dot{\boldsymbol{q}}; \boldsymbol{q}]$	$2n$
$\boldsymbol{w}$	:	Motor coordinates velocity vector	m
$\boldsymbol{g}$	:	Gravitational forces vector	n
$\boldsymbol{c}$	:	Sum of state-dependent generalized forces	n
$\boldsymbol{d}$	:	Unknown disturbance forces	n
$\boldsymbol{\tau}_I$	:	Sum of intrinsic actuator forces	m
$\boldsymbol{\tau}_E$	:	Sum of extrinsic forces	n
$\boldsymbol{a}$	:	Actuator coordinates position vector	m
$\boldsymbol{\phi}$	:	Constraint vector	c
$\boldsymbol{f}_c$	:	Contact forces vector	c
$\boldsymbol{f}_e$	:	End-effector forces vector	o
$\boldsymbol{p}$	:	End-effector position vector	o
		Matrices	
H	:	Inertia matrix	n x n
D	:	Damping matrix	n x n
C	:	Coriolis/Centrifugal forces matrix	n x n
B	:	Motor torques / generalized forces matrix	n x m
R	:	Gear-ratio matrix	m x m
I	:	Intrinsic actuator inertia matrix (diagonal)	m x m
B	:	Intrinsic actuator damping matrix (diagonal)	m x m
J_a	:	actuator coordinates / joint coordinates Jacobian matrix	m x n
J_e	:	task-space coordinates / joint coordinates Jacobian matrix	o x n
J_c	:	Contact constraints Jacobian matrix	c x n

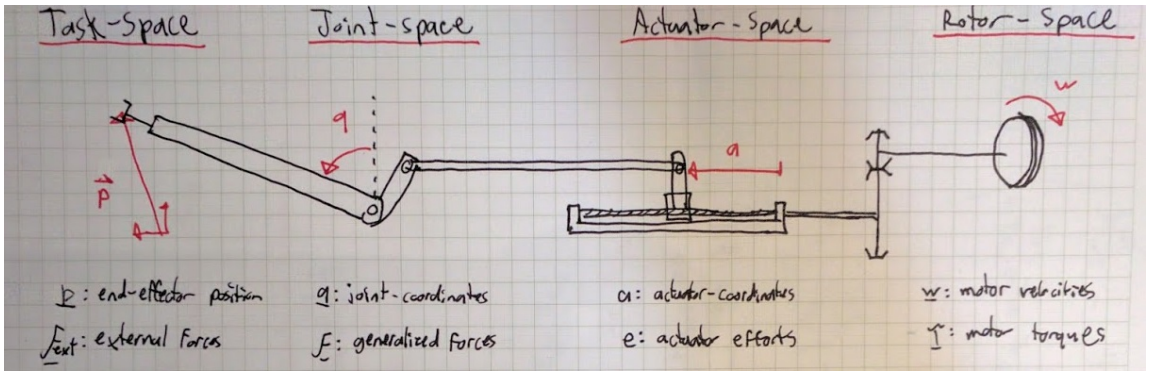


Figure A-1: Coordinate systems

A.2 Coordinate systems

Fig. A-1 shows all the used coordinates systems. The following equation represents the most general case:

$$H_k\ddot{\boldsymbol{q}} + C_k\dot{\boldsymbol{q}} + D_k\dot{\boldsymbol{q}} + \boldsymbol{g} = \underbrace{J_a^T(\boldsymbol{q})R^T}_{B(\boldsymbol{q})}\boldsymbol{\tau} + J_e^T(\boldsymbol{q})\boldsymbol{f}_e + J_c^T(\boldsymbol{q})\boldsymbol{f}_c \tag{A.3}$$

Where matrices with subscript k include the motor-rotor dynamics contributions when the gear-ratio configuration R_k is used:

$$H_k = H + J_a^T R_k^T I_a R_k J_a \tag{A.4}$$

$$C_k = C + J_a^T R_k^T I_a R_k \dot{J}_a \quad \text{if } R \text{ and } J_a \text{ are diagonal} \tag{A.5}$$

$$D_k = D + J_a^T R_k^T B_a R_k J_a \tag{A.6}$$

Coordinates transforms are defined by:

$$\dot{\boldsymbol{p}} = J_e(\boldsymbol{q})\dot{\boldsymbol{q}} \quad \text{from joint-space to end-effector} \tag{A.7}$$

$$\dot{\boldsymbol{a}} = J_a(\boldsymbol{q})\dot{\boldsymbol{q}} \quad \text{from joint-space to actuator-space} \tag{A.8}$$

$$\boldsymbol{w} = R\dot{\boldsymbol{a}} \quad \text{from actuator-space to rotor-space} \tag{A.9}$$

Note that the distinction between actuator output coordinates and joint coordinates, useful for the DSDM-Arm shoulder mechanism, is omitted for brevity in Chapter 4.

A.3 Contact

This section presents equations for representing contact situations.

A.3.1 Kinematic constraints

If a robotic manipulator enter contact with a fixed object, then some DoF are constrained. In the case of a bilateral constraint, the constraint can be expressed as:

$$\boldsymbol{\phi}(\boldsymbol{q}) = 0 \tag{A.10}$$

The time-derivative of the constraint must also be equal to zero, which gives some constraints in terms of velocity and acceleration:

$$\frac{d\boldsymbol{\phi}(\boldsymbol{q})}{dt} = J_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0 \tag{A.11}$$

$$\frac{d^2\boldsymbol{\phi}(\boldsymbol{q})}{dt^2} = J_c(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0 \tag{A.12}$$

when J_c is the constraint Jacobian:

$$J_c(\boldsymbol{q}) = \frac{d\boldsymbol{\phi}(\boldsymbol{q})}{d\boldsymbol{q}} \tag{A.13}$$

A.3.2 Constraint forces

The constraint Jacobian can be used to map constraint forces $\boldsymbol{f}_c$ to generalized forces in the EoM:

$$H\ddot{\boldsymbol{q}} + \boldsymbol{c} = B\boldsymbol{\tau} + J_c(\boldsymbol{q})^T\boldsymbol{f}_c \tag{A.14}$$

Solving for $\ddot{\boldsymbol{q}}$ in eq. (A.14) and substituting in eq. (A.12), it is possible to get and expression for the constraint forces $\boldsymbol{f}_c$ as a function of states and applied torques:

$$\boldsymbol{f}_c = \left(J_c H^{-1} J_c^T\right)^{-1}\left(J_c H^{-1}[\boldsymbol{c} - B\boldsymbol{\tau}] - \dot{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}\right) \tag{A.15}$$

Alternatively, it possible to solve for acceleration $\ddot{\boldsymbol{q}}$ and constraints forces $\boldsymbol{f}_c$ simultaneously by solving the following system of equations:

$$\begin{bmatrix} H & -J_c^T \\ J_c & 0 \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ \boldsymbol{f}_c \end{bmatrix} = \begin{bmatrix} B\boldsymbol{\tau} - \boldsymbol{c} \\ -\dot{J}_c\dot{\boldsymbol{q}} \end{bmatrix} \tag{A.16}$$

A.3.3 Impact impulsive behavior

When the robot first enters contact with a fixed object, impulsive contact forces will act on the system. Integrating eq. (A.14) over the short impact interval gives:

$$\int (H\ddot{\boldsymbol{q}} + \boldsymbol{c})dt = \int (B\boldsymbol{\tau} + J_c(\boldsymbol{q})^T \boldsymbol{f}_c)dt \tag{A.17}$$

$$H\dot{\boldsymbol{q}}^+ - H\dot{\boldsymbol{q}}^- = J_c(\boldsymbol{q})^T \int \boldsymbol{f}_c dt \tag{A.18}$$

where any non-impulsive forces are neglected during the short impact interval. Projecting onto constrained coordinates (multiplying by $J_c H^{-1}$) gives:

$$J_c\dot{\boldsymbol{q}}^+ - J_c\dot{\boldsymbol{q}}^- = J_c H^{-1} J_c^T \int \boldsymbol{f}_c dt \tag{A.19}$$

Then assuming a sticky inelastic impact (no bouncing), then the constraint is respected after the impact ($J_c\dot{\boldsymbol{q}}^+ = 0$) and it is possible to solve for the impact force:

$$\int \boldsymbol{f}_c dt = -\left(J_c H^{-1} J_c^T\right)^{-1} J_c\dot{\boldsymbol{q}}^- \tag{A.20}$$

and also for the velocity after the impact:

$$\dot{\boldsymbol{q}}^+ = -\Big[I - H^{-1}J_c^T\left(J_c H^{-1} J_c^T\right)^{-1} J_c\Big]\dot{\boldsymbol{q}}^- \tag{A.21}$$

Or change in velocity:

$$\Delta\dot{\boldsymbol{q}} = \Big[H^{-1}J_c^T\left(J_c H^{-1} J_c^T\right)^{-1} J_c\Big]\dot{\boldsymbol{q}}^- \tag{A.22}$$

Alternatively, it possible to solve for velocity $\dot{\boldsymbol{q}}^+$ and impulsive forces $\int \boldsymbol{f}_c dt$ simultaneously by solving the following system of $n + c$ equations:

$$\begin{bmatrix} H & -J_c^T \\ J_c & 0 \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{q}}^+ \\ \int \boldsymbol{f}_c dt \end{bmatrix} = \begin{bmatrix} H\dot{\boldsymbol{q}}^- \\ 0 \end{bmatrix} \tag{A.23}$$

A.4 Hybrid system dynamics

Hybrid dynamical system can be represented in the general form:

$$\text{Continuous evolution: } \left(\dot{\boldsymbol{x}}, \dot{k}\right) = \left(f_k(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}) \, , \, 0 \right) \tag{A.24}$$

$$\text{Discrete jumps: } \left(\boldsymbol{x}^+, k^+\right) = \left(h_{ij}(\boldsymbol{x}^-, \boldsymbol{u}^-), j\right) \quad \text{if} \quad (\boldsymbol{x}, k, \boldsymbol{u}) \in D_{ij} \tag{A.25}$$

where $\boldsymbol{x}$ is a continuous state vector, and k is a discrete mode and D_{ij} is the domain mapping conditions leading to a transition $k : i \to j$. For robotic systems, the discrete mode can represent discrete configurations of the robot , like gear-ratios in this thesis, and contact/non-contact conditions. The jump map then represents the impulsive response when contact is made.

A.4.1 Switched system

A restricted class of hybrid system, called switched system, are hybrid systems for which the jump map for continuous state is the identify function:

$$\text{Continuous evolution: } \left(\dot{\boldsymbol{x}}, \dot{k}\right) = \left(f_k(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}) \, , \, 0 \right) \tag{A.26}$$

$$\text{Discrete jumps: } \left(\boldsymbol{x}^+, k^+\right) = \left(\boldsymbol{x}^-, j\right) \quad \text{if} \quad (\boldsymbol{x}, k, \boldsymbol{u}) \in D_{ij} \tag{A.27}$$

Switched system where the discrete mode is a control input

In the situation where the discrete operating mode k is a control input, then there is no need to keep track of discrete mode evolution and only the piece-wise continuous

differential equations are sufficient to model the system evolution:

$$\dot{\boldsymbol{x}} = f_k(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}) \tag{A.28}$$

The model for robots using variable transmissions with discrete configurations, that is used in Chapter 4, is of this category.

Appendix B

Mathematical Derivations

B.1 Simplified equations of motion for diagonal R

Here the derivation of the simplified form of equations of motion, where intrinsic and extrinsic forces are two separate terms, is derived for a class of n-DoF fully actuated robots.

B.1.1 Assumptions

Three assumptions are necessary for this form. First, the robotic system is fully-actuated, i.e. $m = n$, so that the matrix R is square and invertible. Second, that a coordinates system can be selected so that the gear-ratio matrix R is diagonal for all possible configurations:

$$R_{i,j} = 0 \quad \forall\, i \neq j \tag{B.1}$$

Third, that dynamic forces related to viscous damping and inertial forces in motor rotor are linear with respect to rotor velocity:

$$\boldsymbol{\tau}_{rotor-inertia} = I\dot{\boldsymbol{w}} \quad \boldsymbol{\tau}_{rotor-damping} = B\boldsymbol{w} \tag{B.2}$$

where matrices I and D are diagonal since motor rotors are not coupled directly.

B.1.2 Derivation

Starting from the general eq.(4.9) the EoM are:

$$H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} = R^T\left[\boldsymbol{\tau} - I\dot{\boldsymbol{w}} - B\boldsymbol{w}\right] \tag{B.3}$$

Then substituting motor velocities with joint coordinates, using the kinematic relation of eq. (4.8):

$$H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} = R^T\left[\boldsymbol{\tau} - IR\ddot{\boldsymbol{q}} - BR\dot{\boldsymbol{q}}\right] \tag{B.4}$$

$$H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} = R^T\boldsymbol{\tau} - R^TIR\ddot{\boldsymbol{q}} - R^TBR\dot{\boldsymbol{q}} \tag{B.5}$$

Then because R, I and B matrix are diagonal, they can be permuted and also $R^T = R$. Hence, the EoM can be rearranged:

$$H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g} = R\boldsymbol{\tau} - RRI\ddot{\boldsymbol{q}} - RRB\dot{\boldsymbol{q}} \tag{B.6}$$

Then, assuming the robotic system is fully actuated, the R matrix is square and invertible. Then multiplying by R^{-1} from the left on both side:

$$R^{-1}\left[H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g}\right] = \boldsymbol{\tau} - RI\ddot{\boldsymbol{q}} - RB\dot{\boldsymbol{q}} \tag{B.7}$$

Then rearranging:

$$R^{-1}\left[H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g}\right] = \boldsymbol{\tau} - R\left[I\ddot{\boldsymbol{q}} + B\dot{\boldsymbol{q}}\right] \tag{B.8}$$

and thus obtain the desired final form:

$$\boldsymbol{\tau} = R^{-1}\underbrace{\left[H\ddot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} + D\dot{\boldsymbol{q}} + \boldsymbol{g}\right]}_{\boldsymbol{\tau}_E} + R\underbrace{\left[I\ddot{\boldsymbol{q}} + B\dot{\boldsymbol{q}}\right]}_{\boldsymbol{\tau}_I} \tag{B.9}$$

B.2 Optimal gear-ratio along a known trajectory

Assuming that the robot is fully actuated and viscous forces linear in speed, it is possible to derive closed form expression for the optimal gear-ratios on a known trajectory.

B.2.1 Single DoF

Starting with the EoM in inverse dynamic form (from eq. (4.7)):

$$\tau = \frac{\tau_E}{R} + R\tau_I \tag{B.10}$$

Using a quadratic cost function to minimize:

$$J = \tau^2 = \frac{\tau_E^2}{R^2} + 2\tau_E\tau_I + R^2\tau_I^2 \tag{B.11}$$

Finding the gear-ratio that minimizes this cost can be formulated as:

$$R^* = \underset{R}{\operatorname{argmin}}\, J \tag{B.12}$$

$$\text{s.t.} \quad R \in \Re \quad \& \quad R > 0 \tag{B.13}$$

A non-real value would have no physical sense. A negative R value would be physically possible, for instance the reverse gear in a car. However, for symmetric electric motors, in the sense that they behave the same way for any sign of torque and speed, there should be no gain obtained by changing the direction of the motor velocity. This is consistent with cost function which is symmetric with respect to R:

$$J(R) = J(-R) \tag{B.14}$$

Derivation First finding the partial derivative of the cost J with respect to R:

$$\frac{\partial J}{\partial R} = 2\tau\frac{\partial \tau}{\partial R} = 2\left(\frac{\tau_E}{R} + R\tau_I\right)\left(-\frac{\tau_E}{R^2} + \tau_I\right) \tag{B.15}$$

$$\frac{\partial J}{\partial R} = 2\left(R\tau_I^2 - \frac{\tau_E^2}{R^3}\right) \tag{B.16}$$

Then the second derivative:

$$\frac{\partial^2 J}{\partial R^2} = 2\left(\tau_I^2 + 3\frac{\tau_E^2}{R^4}\right) \tag{B.17}$$

Hence, on the domain of interest, the second derivative is always positive:

$$\frac{\partial^2 J}{\partial R^2} \geq 0 \quad \forall \quad R \in (0, +\infty) \tag{B.18}$$

Thus the cost function J is convex on the desired interval of possible R values. The minimum of the function can thus be found by solving for the point where the first derivative is equal to zero:

$$0 = \frac{\partial J}{\partial R} = 2\left(R\tau_I^2 - \frac{\tau_E^2}{R^3}\right) \tag{B.19}$$

$$R\tau_I^2 = \frac{\tau_E^2}{R^3} \tag{B.20}$$

Since $R > 0$, it is possible to multiply both side by R^3, leading to

$$R^4\tau_I^2 = \tau_E^2 \tag{B.21}$$

Then, assuming a non-degenerative case of $\tau_I \neq 0$, it leads to

$$R^4 = \frac{\tau_E^2}{\tau_I^2} \tag{B.22}$$

$$R^2 = \pm\sqrt{\frac{\tau_E^2}{\tau_I^2}} = \pm\frac{\tau_E}{\tau_I} \tag{B.23}$$

$$R = \pm\sqrt{\pm\frac{\tau_E}{\tau_I}} \tag{B.24}$$

Then, the only real and positive solution to this equation is given by:

$$R = \sqrt{\left|\frac{\tau_E}{\tau_I}\right|} \tag{B.25}$$

Solution The minimal cost value is thus obtain with the optimal gear-ratio value:

$$R^* = \underset{R>0}{\operatorname{argmin}}\, J = \sqrt{\left|\frac{\tau_E}{\tau_I}\right|} \tag{B.26}$$

Which lead to the minimum cost:

$$J^* = 2\tau_E\tau_I + 2\left|\tau_E\tau_I\right| \tag{B.27}$$

Note that the the minimized cost is zero when extrinsic and intrinsic forces have opposite signs.

Degenerative cases Here degenerative situations when the intrinsic forces or extrinsic forces are equal to zero are investigated, based on (B.11). If the intrinsic forces are equal to zero, then the cost tends towards zero as the gear-ratio R tends toward ∞:

$$\tau_I = 0 \quad \& \quad R \to \infty \quad \Rightarrow \quad J \to 0 \tag{B.28}$$

If the extrinsic forces are equal to zero, then the cost tends towards zero as the gear-ratio R tends toward zero:

$$\tau_E = 0 \quad \& \quad R \to 0 \quad \Rightarrow \quad J \to 0 \tag{B.29}$$

If both the extrinsic forces and intrinsic forces are equal to zero, then the cost is zero for any gear-ratio:

$$\tau_E = 0 \quad \& \quad \tau_I = 0 \quad \Rightarrow \quad J = 0 \quad \forall R \tag{B.30}$$

B.2.2 Multiple DoF

Starting with the EoM in inverse dynamic form (from eq. (4.10)):

$$\boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E + R\boldsymbol{\tau}_I \tag{B.31}$$

Using the following quadratic cost function:

$$J = \boldsymbol{\tau}^T\boldsymbol{\tau} \tag{B.32}$$

Finding the gear-ratios matrix R that minimize this cost can be formulated as

$$R^* = \underset{R}{\operatorname{argmin}}\, J \tag{B.33}$$

$$\text{s.t.} \quad R_{i,j} \in \Re \quad \& \quad R_{i,j} > 0 \tag{B.34}$$

An analytic solution is available if the gear-ratios matrix R is diagonal.

Derivation Using index notation, the EoM and cost function can be written as:

$$\boldsymbol{\tau} = R^{-1}\boldsymbol{\tau}_E + R\boldsymbol{\tau}_I \quad \Rightarrow \quad \tau_i = \sum_j \left[R^{-1}\right]_{i,j} \tau_j^E + R_{i,j}\tau_j^I \tag{B.35}$$

$$J = \boldsymbol{\tau}^T\boldsymbol{\tau} \quad \Rightarrow \quad J = \sum_i \tau_i^2 \tag{B.36}$$

Note that here, superscript instead of subscript are used to identify extrinsic and intrinsic forces, to avoid confusion with indexes. Then the properties due to the diagonality of matrix R can be used:

$$R_{i,j} = 0 \quad \forall \quad i \neq j \tag{B.37}$$

$$\left[R^{-1}\right]_{i,j} = 0 \quad \forall \quad i \neq j \tag{B.38}$$

$$\left[R^{-1}\right]_{i,i} = \left(R_{i,i}\right)^{-1} \tag{B.39}$$

Then the equations can be simplified to:

$$\tau_i = (R_{i,i})^{-1} \tau_i^E + R_{i,i} \tau_i^I \tag{B.40}$$

$$J = \sum_i \left[(R_{i,i})^{-1} \tau_i^E + R_{i,i} \tau_i^I \right]^2 \tag{B.41}$$

By inspection, it is possible to see that the cost J is the sum of n independent terms (one per DoF), and that given the assumptions those terms are independent. Hence, the cost J can be minimized by minimizing individually each term with the appropriate $R_{i,i}$. The solution for minimizing each of those term is identical to the one for a single DoF robot, see section B.2.1. Leading to

$$R_{i,i}^* = \sqrt{\left| \frac{\tau_i^E}{\tau_i^I} \right|} \tag{B.42}$$

Solution The optimal gear-ratio matrix, is thus constructed from independent solutions on each DoF:

$$R^* = \begin{cases} R_{i,j}^* = \sqrt{\left| \frac{\tau_i^E}{\tau_i^I} \right|} & \forall \quad i = j \\ R_{i,j}^* = 0 & \forall \quad i \neq j \end{cases} \tag{B.43}$$

Leading to the following total minimum cost:

$$J^* = 2 \sum_i \left[\tau_i^E \tau_i^I + \left| \tau_i^E \tau_i^I \right| \right] \tag{B.44}$$

B.3 Stability proofs

B.3.1 R* Computed Torque controller

In this section, the stability of motions when using the R* Computed Torque Controller is demonstrated for any arbitrary sequence of selected gear-ratio. However, here perfect knowledge of the equation of motions is assumed.

The equation of motions can take this simple but general form:

$$H_k \ddot{\boldsymbol{q}} + \boldsymbol{c}_k = R_k \boldsymbol{\tau} \quad \forall k \in \{1, ..., l\} \tag{B.45}$$

where subscript k is used to emphasized the dependence to the discrete gear-ratio selection. The total inertia matrix H_k and state-dependent forces c_k are given by:

$$H_k = H(\boldsymbol{q}) + R_k^T I R_k \tag{B.46}$$

$$\boldsymbol{c}_k = \left(D + R_k^T B R_k\right) \dot{\boldsymbol{q}} + C(\dot{\boldsymbol{q}}, \boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) \tag{B.47}$$

In the computed torque scheme, it is assumed that based on state measurement those term can be computed exactly. In addition here, it is assumed the controller is also aware of the discrete gear-ratio state k.

The control law for motor torques takes the following form:

$$\boldsymbol{\tau} = R_k^{-1} \left(H_k \ddot{\boldsymbol{q}}_r + \boldsymbol{c}_k\right) \tag{B.48}$$

where the vector $\ddot{\boldsymbol{q}}_r$ represents instantaneous desired acceleration. The control law for the gear-ratio selection takes the form of an optimization, however here stability is demonstrated for the more general case of arbitrary gear-ratio sequence. Hence the stability result can be extended for any type of gear-ratio selection scheme used in conjunction with the computed torque.

When substituting the control law, eq.(B.48), in the equations of motion, eq.(B.45),

then the resulting closed-loop behavior is simply:

$$\ddot{\boldsymbol{q}} = \ddot{\boldsymbol{q}}_r \quad \forall k \tag{B.49}$$

Hence, the system is not only linear in behavior (form a $\ddot{\boldsymbol{q}}_r$ input point of view) but also continuous, non-linearities and discontinuities are compensated by computing the inverse dynamic of the system.

Specifying $\ddot{\boldsymbol{q}}_r$ based on state is equivalent to designing a controller for a linear second-order n-DoF system. For trajectory tracking the following proportional-derivative control laws can be used:

$$\ddot{\boldsymbol{q}}_r = \ddot{\boldsymbol{q}}_d + K_D(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}) + K_P \underbrace{(\boldsymbol{q}_d - \boldsymbol{q})}_{\boldsymbol{q}_e} \tag{B.50}$$

leading to second order error dynamics:

$$0 = \ddot{\boldsymbol{q}}_e + K_D\dot{\boldsymbol{q}}_e + K_P\boldsymbol{q}_e \tag{B.51}$$

Hence, convergence of error to zero is guaranteed if both gain matrix K_D and K_P are positive definite:

$$\boldsymbol{q}_e \to \mathbf{0} \quad \text{with} \quad K_D > 0\,,\, K_P > 0 \tag{B.52}$$

B.3.2 R* Sliding Mode controller

In this section, the stability of motions when using the R* Sliding Mode Controller is demonstrated for any arbitrary sequence of selected gear-ratio, in the presence of bounded uncertainty.

The equation of motions can take this simple but general form:

$$H_k\ddot{\boldsymbol{q}} + \boldsymbol{c}_k = R_k\boldsymbol{\tau} + \boldsymbol{d} \quad \forall k \in \{1, ..., l\} \tag{B.53}$$

where $\boldsymbol{d}$ is an unknown generalized force vector that can represent disturbance or modeling errors.

The proposed control law for motor torques takes the form:

$$\boldsymbol{\tau} = R_k^{-1}\left(H_k\ddot{\boldsymbol{q}}_r + \boldsymbol{c}_k - G_k sgn(\boldsymbol{s})\right) \tag{B.54}$$

leading to the following closed-loop behavior:

$$H_k(\ddot{\boldsymbol{q}} - \ddot{\boldsymbol{q}}_r) = \boldsymbol{d} - G_k sgn(\boldsymbol{s}) \quad \forall k \tag{B.55}$$

which has two types of discontinuities: 1) discontinuous torque term introduced by the sliding mode controller and 2) arbitrary selected gear-ratio k. The following variables are then introduced:

$$\boldsymbol{q}_e = \boldsymbol{q}_d - \boldsymbol{q} \tag{B.56}$$

$$\dot{\boldsymbol{q}}_r = \dot{\boldsymbol{q}}_d - \lambda\boldsymbol{q}_e \tag{B.57}$$

$$\boldsymbol{s} = \dot{\boldsymbol{q}}_e + \lambda\boldsymbol{q}_e = \dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_r \tag{B.58}$$

$$\dot{\boldsymbol{s}} = \ddot{\boldsymbol{q}}_e + \lambda\dot{\boldsymbol{q}}_e = \ddot{\boldsymbol{q}} - \ddot{\boldsymbol{q}}_r \tag{B.59}$$

where lambda is a positive constant. Then convergence to the desired trajectory can be guaranteed if the sliding variables $\boldsymbol{s}$ converge to zero [56]. The basic idea is that if a Lyapunov-like quadratic function of the sliding variable is only decreasing in time,

then sliding variables will converge to zero, which also implies that the error will converge to zero:

$$V = \boldsymbol{s}^T \boldsymbol{s} \to 0 \quad \Rightarrow \quad \boldsymbol{s} \to \mathbf{0} \quad \Rightarrow \quad \boldsymbol{q}_e \to \mathbf{0} \tag{B.60}$$

Note that here the Lyapunov-like function must be continuous and the same for all discrete mode, and its derivative negative definite for all discrete modes k for the argument to hold for arbitrary sequence of gear-ratio k [37]:

$$\dot{V}_k < 0 \quad \forall k \qquad \Rightarrow \quad V \to 0 \tag{B.61}$$

One way to guarantee the sliding condition is individually for each DoF i and each possible mode k:

$$\frac{d(s_i^2)}{dt} < -\eta |s_i| \quad \forall i \; \forall k \tag{B.62}$$

where η is a small positive constant. From eq.(B.55), sliding variable derivative is given by:

$$\dot{\boldsymbol{s}} = H_k^{-1}(\boldsymbol{d} - G_k sgn(\boldsymbol{s})) \tag{B.63}$$

Substituting in the sliding condition equation leads to:

$$s_i \dot{s}_i = s_i \left(\left[H_k^{-1} \boldsymbol{d} \right]_i - \left[H_k^{-1} G_k sgn(\boldsymbol{s}) \right]_i \right) < -\eta |s_i| \tag{B.64}$$

If the matrix gain is parametrized in the following way:

$$G_k = H_k K_k \tag{B.65}$$

where K_k is a diagonal matrix. Then the discontinuous gain term is uncoupled in the

sliding condition equation:

$$s_i \left(\left[H_k^{-1} \boldsymbol{d} \right]_i - K_{ii} sgn(s_i) \right) < -\eta |s_i| \tag{B.66}$$

$$\left[H_k^{-1} \boldsymbol{d} \right]_i s_i + \eta |s_i| < K_{ii} |s_i| \tag{B.67}$$

$$\eta \pm \left[H_k^{-1} \boldsymbol{d} \right]_i < K_{ii} \tag{B.68}$$

which can be guaranteed if the gain are selected such that:

$$K_{ii} = \max_{\boldsymbol{d}} \left[H_k^{-1} \boldsymbol{d} \right]_i + \eta \tag{B.69}$$

Thus, if the gains are defined based on disturbance bounds and according to eq.(B.69), the sliding condition is guaranteed for all gear-ratio mode k, and thus convergence to the desired trajectory is guaranteed for any gear-ratio sequence. Note that the gain K_{ii} is a function of the discrete selected gear-ratio mode k.

B.4 Chattering bounds with Rollout gear selection

This section computes lower bounds for the time interval between successive gear-shifts when using the Rollout gear-selection scheme.

Problem setting First, using the Rollout approach, the optimal gear-ratio mode k^* selection is done as follow:

$$k^*(t) = \underset{k}{\operatorname{argmin}} \left[J_k(t) + Q \left[[k \neq k_{previous}] \right] \right] \tag{B.70}$$

where $J_k(t)$ is the computed cost over the simulated trajectory over the time horizon h using the gear-ratio mode k. An additional instantaneous cost Q is added when the gear-ratio mode is changed. The integral cost is given by:

$$J_k(t) = \int_t^{t+h} C_k(\hat{\boldsymbol{x}}_k)\, d\hat{t} \tag{B.71}$$

where C_k is the instantaneous cost, $\hat{t}$ is the virtual time in the simulations and $\hat{\boldsymbol{x}}$ the predicted trajectory. The virtual state trajectory is computed by simulating the system in closed-loop, using the base-policy, and integrating forward starting from the actual state at time t:

$$\hat{\boldsymbol{x}}_k(\hat{t}, t) = \int_t^{\hat{t}} \dot{\hat{\boldsymbol{x}}}_k\, d\tilde{t} + \boldsymbol{x}(t) \tag{B.72}$$

If both the real system and simulations are robustly staying on the desired trajectory, then the virtual states in the simulation are independent of actual starting simulation time:

$$\hat{\boldsymbol{x}}_k(\hat{t}, t) = \boldsymbol{x}(\hat{t}) = \boldsymbol{x}_d(\hat{t}) \tag{B.73}$$

In this thesis the instantaneous cost is usually taken to be squared actuator torques:

$$C_k = \boldsymbol{\tau}_k^T \boldsymbol{\tau}_k \tag{B.74}$$

where $\boldsymbol{\tau}_k$ is the computed torque in predictive simulation with the base policy when using the gear-ratio mode k. It will be assumed that an upper bound can be found for the instantaneous cost, at least in a domain of interest D, given the desired trajectory and the feedback policy:

$$C_k^{max} = \max_{\boldsymbol{x}, t \in D} C_k(\boldsymbol{x}, t) \tag{B.75}$$

Such a bound should always exist given reasonable assumptions:

- Domain of interest D constraint all states to finite values
- Desired trajectory has bounded acceleration, speed and position values
- Absence of singularity where inertia matrix values can tend toward infinity

For instance, if using computed torque as base policy and a torque-squared criteria, the maximum instantaneous cost value is given by:

$$C_k^{max} = \max \boldsymbol{\tau}_k^T \boldsymbol{\tau}_k \leq \sum \max \left[\boldsymbol{\tau}_k\right]_i^2 \tag{B.76}$$

where $\max \left[\boldsymbol{\tau}_k\right]_i$ is the maximum value that can be computed for joint i when using the gear-ratio k.

B.4.1 On a trajectory

Here a minimum time delay for a back-and-forth gear-shift sequence is derived for the situation where the robot has converged on a trajectory and is staying on the trajectory. In that case, it is assumed that both the simulation trajectory and the real system follow the same desired trajectory, see Fig. B-1.

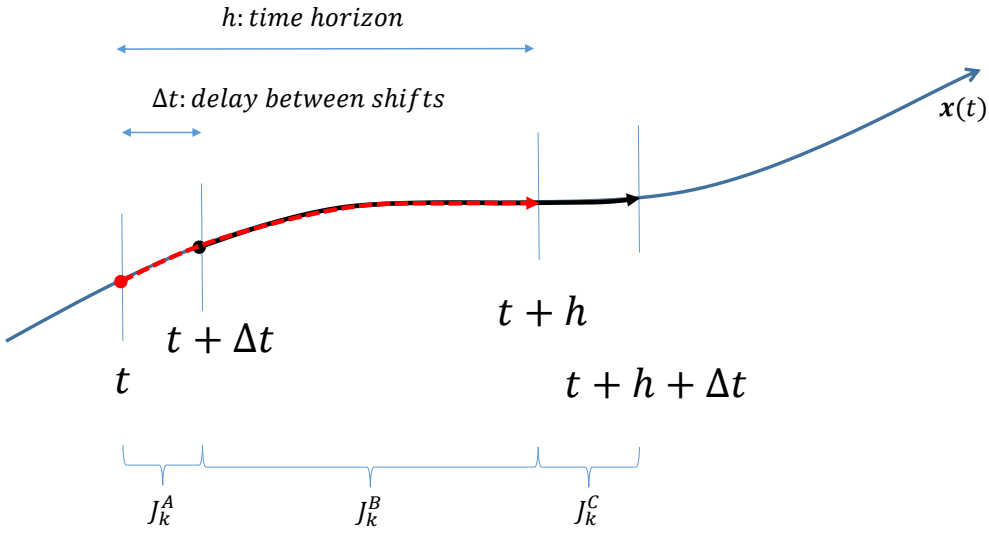


Figure B-1: Rollout controller behavior on a given trajectory

Proof Without loss of generality, lets assume a gear shift sequence where the robot is first using mode i, then shift to mode j at time t and then shift back to mode i at time $t + \Delta t$. If such a sequence happened while using the gear-ratio selection law given by eq. (B.70), then the following inequality must have been satisfied:

$$i \to j \quad \text{at time} \quad t \quad \Rightarrow \quad J_j(t) + Q < J_i(t) \tag{B.77}$$

$$j \to i \quad \text{at time} \quad t + \Delta t \quad \Rightarrow \quad J_i(t + \Delta t) + Q < J_j(t + \Delta t) \tag{B.78}$$

Then to simplify the notation, integral cost on the trajectory is given by the following value for each three sections of interest:

$$J_k^A = \int_t^{t+\Delta t} C_k d\hat{t} \qquad J_k^B = \int_{t+\Delta t}^{t+h} C_k d\hat{t} \qquad J_k^C = \int_{t+h}^{t+\Delta t+h} C_k d\hat{t} \tag{B.79}$$

Then it is possible to express the computed cost of predictive simulation done at Δt time difference by the sum of an identical part and a different part:

$$J_k(t) = \int_t^{t+h} C_k d\hat{t} = J_k^A + J_k^B \tag{B.80}$$

$$J_k(t+\Delta t) = \int_{t+\Delta t}^{t+\Delta t+h} C_k d\hat{t} = J_k^B + J_k^C \tag{B.81}$$

The difference between computed costs separated by Δt is:

$$\Delta J_k = J_k(t+\Delta t) - J_k(t) = J_k^C - J_k^A \tag{B.82}$$

Now assuming there is an upper bound on instantaneous cost C_k^{max}, an upper bound also exist integral costs over a finite amount of time. Since the instantaneous cost is always positive definite, the integral cost cannot be negative. Hence:

$$0 \leq J_k^C \leq C_k^{max}\Delta t \tag{B.83}$$

$$0 \leq J_k^A \leq C_k^{max}\Delta t \tag{B.84}$$

The cost variation is thus bounded is this range:

$$-C_k^{max}\Delta t \leq \Delta J_k \leq C_k^{max}\Delta t \tag{B.85}$$

Hence the time evolution of computed cost can be bounded:

$$J_j(t+\Delta t) \leq J_j(t) + C_k^{max}\Delta t \tag{B.86}$$

$$J_i(t+\Delta t) \geq J_i(t) - C_k^{max}\Delta t \tag{B.87}$$

Finally it is possible to combine all those inequality, starting with the condition for the second gearshift:

$$J_i(t) - C_k^{max}\Delta t + Q \leq J_i(t+\Delta t) + Q < J_j(t+\Delta t) \leq J_j(t) + C_k^{max}\Delta t \tag{B.88}$$

$$J_i(t) + Q \leq J_j(t) + 2C_k^{max}\Delta t \tag{B.89}$$

$$J_j(t) + 2Q \leq J_j(t) + 2C_k^{max}\Delta t \tag{B.90}$$

$$Q \leq C_k^{max}\Delta t \tag{B.91}$$

Hence resulting in the desired inequality relating instantaneous cost and time between gear-shift:

$$\Delta t \geq \frac{Q}{C_k^{max}} \tag{B.92}$$

Note that if there is more than 2 discrete gear-ratio options, this analysis gives not insight about possible sequences of gear-shift during this interval, but still guarantee the minimum time for a full cycle (coming back to the starting gear-ratio).

B.4.2 Arbitrary

Fig. B-2 gives a graphical support illustrating the 2D space, where at all time virtual trajectories branch-off the real robot trajectory.

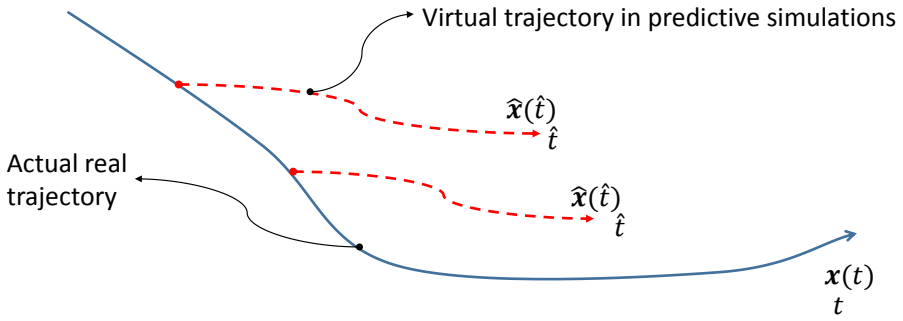


Figure B-2: Real and simulated trajectory in the Rollout controller

The time derivative of the cost J_k, representing how much the computed cost

change if the simulation is conducted dt later in time, is given by:

$$\dot{J}_k(t) = \frac{d}{dt}\int_t^{t+h} C_k \, d\hat{t} \tag{B.93}$$

Using Leibniz's rule:

$$\dot{J}_k(t) = C_k(t+h) - C_k(t) + \int_t^{t+h} \frac{\partial}{\partial t} C_k \, d\hat{t} \tag{B.94}$$

for which it is possible to find an upper bound:

$$\dot{J}_k(t) \leq C_k^{max} - 0 + \int_t^{t+h} \dot{C}_k^{max} \, d\hat{t} \tag{B.95}$$

$$\dot{J}_k(t) \leq C_k^{max} + \dot{C}_k^{max} h \tag{B.96}$$

Note that compared to the last section analysis (fixed trajectory), there is an additional term $\dot{C}_k^{max} h$ representing sensitivity of states trajectory in simulations. Then, most of the previous analysis is still valid, except that the bounds on cost variation of eq. (B.85), are now given instead by:

$$-\left[C_k^{max} + \dot{C}_k^{max} h\right] \Delta t \leq \Delta J_k \leq \left[C_k^{max} + \dot{C}_k^{max} h\right] \Delta t \tag{B.97}$$

And the final result is modified to be:

$$\Delta t \geq \frac{Q}{C_k^{max} + \dot{C}_k^{max} h} \tag{B.98}$$

Sensitivity of cost along simulated trajectories

Regarding the new term, the time derivative of instantaneous cost in the simulation can be expressed as:

$$\dot{C}_k = \frac{\partial C_k}{\partial t} = \frac{\partial C_k}{\partial \boldsymbol{\tau}} \frac{\partial \boldsymbol{\tau}}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial t} \tag{B.99}$$

This term represent how much the computed cost can vary on simulated trajectories, between two simulations computed with dt time interval.

Cost function sensitivity The term $\frac{\partial C_k}{\partial \boldsymbol{\tau}}$ is the sensitivity of the used cost function with respect to computed torques. For a quadratic torque criterion $C_k = \boldsymbol{\tau}_k^T \boldsymbol{\tau}_k$, like always used in this thesis, it is equal to:

$$\frac{\partial C_k}{\partial \boldsymbol{\tau}_k} = 2\boldsymbol{\tau}_k^T \tag{B.100}$$

Control law sensitivity The term $\frac{\partial \boldsymbol{\tau}}{\partial \hat{\boldsymbol{x}}}$ is a Jacobian matrix representing the sensitivity of computed torque with respect to states. This matrix could be computed analytically with the feedback laws, for the computed torque control law:

$$\frac{\partial \boldsymbol{\tau}}{\partial \hat{\boldsymbol{x}}} = \begin{bmatrix} H_k^{-1} K_D & 0 \\ 0 & H_k^{-1} K_P \end{bmatrix} + \begin{bmatrix} H_k^{-1}(D + C + R_k^T B R_k) & 0 \\ 0 & H_k^{-1} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{q}} \end{bmatrix} \tag{B.101}$$

For the sliding mode control law, because of the additional discontinuous term $\boldsymbol{\tau}_d = G\,\mathrm{sgn}(\boldsymbol{s})$, the derivative $\frac{\partial \boldsymbol{\tau}}{\partial \hat{\boldsymbol{x}}}$ can be unbounded when $s_i \approx 0$. However, the difference over a finite time interval Δt is bounded since worst case scenario:

$$-2G \leq [\Delta \boldsymbol{\tau}_d]_i \leq 2G \tag{B.102}$$

Lets assume, for simplicity, that the cost function would penalize independently the continuous torque term and the discontinuous torque term:

$$C_k = C_c + C_d = C_c + \boldsymbol{\tau}_d^T \boldsymbol{\tau}_d \tag{B.103}$$

then the resulting modified lower bound on minimum delay would be:

$$\Delta t \geq \frac{Q - \max\left[\boldsymbol{\tau}_d^T \boldsymbol{\tau}_d\right] h}{C_c^{max} + \dot{C}_c^{max} h} \tag{B.104}$$

This shows that with large discontinuous gains, it would be harder to guarantee a minimum delay. There is thus a double advantages for using a smoothing technique (ex: boundary layer [56] or higher order sliding mode [46]) for which the control law is not discontinuous. In addition to the advantage of smoothing the torque command, it would also make it easier to design a rollout gear selection that guarantee a minimum delay. Alternatively, as proposed in section 4.4.5, reformulating the cost function to make it independent of the sign of $\boldsymbol{s}$ it another approach to alleviate this problem.

Virtual trajectory sensitivity The term $\frac{\partial \hat{\boldsymbol{x}}}{\partial t}$ represents by how much the states at time $\hat{t}$ (in the simulation) would change if the simulation would have started dt later. The virtual trajectory can be parametrized as the sum of the desired trajectory and errors:

$$\hat{\boldsymbol{x}}(\hat{t}, t) = \hat{\boldsymbol{x}}_e(\hat{t}, t) + \hat{\boldsymbol{x}}_d(\hat{t}) \tag{B.105}$$

Because the desired trajectory is independent of starting simulation time t, the sensitivity of the trajectory is only the sensitivity of this error term:

$$\frac{\partial \hat{\boldsymbol{x}}}{\partial t} = \frac{\partial \hat{\boldsymbol{x}}_e}{\partial t} \tag{B.106}$$

For the closed-loop system with the computed torque controller, the error dynamic is stable and linear, hence:

$$\dot{\boldsymbol{x}}_e = A\boldsymbol{x}_e \quad \Rightarrow \quad \boldsymbol{x}_e(t) = e^{At}\boldsymbol{x}_e(t=0) \tag{B.107}$$

Similarly for the sliding mode controller, after all sliding surfaces are reached (happens in finite time) then the error dynamic is also linear and stable. Using this matrix exponential solution for error trajectories in the simulations leads to

$$\hat{\boldsymbol{x}}_e(\hat{t}, t) = e^{A(\hat{t}-t)}\boldsymbol{x}_e(t) \tag{B.108}$$

since initial error in the simulations is the real error at time t. It is then possible to compute the sensitivity of virtual state trajectory with respect to simulation initialization time:

$$\frac{\partial \hat{\boldsymbol{x}}}{\partial t} = -Ae^{A(\hat{t}-t)}\boldsymbol{x}_e(t) + e^{A(\hat{t}-t)}\dot{\boldsymbol{x}}_e(t) \leq \boldsymbol{b} \quad \forall\, \hat{t} > t \tag{B.109}$$

Hence trajectory sensitivity can thus be bounded, because the closed-loop error dynamic is linear and stable in simulations (matrix A only has eigenvalues with negative real parts).

Sensitivity conclusions All in all, it is thus possible to conclude that:

1. Sensitivity $\dot{C}_k$ is bounded (with a continuous control law)
2. Sensitivity $\dot{C}_k \to 0$ far ahead in simulations as $\hat{t} \to \infty$
3. Sensitivity $\dot{C}_k = 0$ if there is no tracking error ($\boldsymbol{x}_e = \boldsymbol{0} \quad \dot{\boldsymbol{x}}_e = \boldsymbol{0}$)

Note that conclusion 3 is consistent with the previous analysis (sec. B.4.1) where it was assumed a situation where all trajectories (real, desired and simulations) were all identical. Also conclusion 2 suggest that the sensitivity of future cost would no grow forever as the time horizon h is increased, hence by using the following:

$$\int_t^{t+h} \dot{C}_k \, d\hat{t} \leq \dot{C}_k^{max} h \tag{B.110}$$

the result is very conservative when the time horizon h is large.

Bibliography

[1] H. H. Asada and J.-J. E. Slotine. *Robot Analysis and Control*. John Wiley & Sons, 1986.

[2] H. H. Asada and K. Youcef-Toumi. *Direct-Drive Robots: Theory and Practice*. MIT Press, 1987.

[3] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2nd edition, 2000.

[4] J. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Advances in Design and Control. Society for Industrial and Applied Mathematics, 2010.

[5] S. Bologna. Electric propulsion system for vehicles, June 2014. U.S. Patent.

[6] B. L. Bonilla and H. H. Asada. A robot on the shoulder: Coordinated human-wearable robot control using Coloured Petri Nets and Partial Least Squares predictions. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 119–125, May 2014.

[7] F. Borrelli, M. Baotić, A. Bemporad, and M. Morari. Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica*, 41(10):1709–1721, October 2005.

[8] R. Buckingham, V. Chitrakaran, R. Conkie, G. Ferguson, A. Graham, A. Lazell, M. Lichon, N. Parry, F. Pollard, A. Kayani, M. Redman, M. Summers, and B. Green. Snake-Arm Robots: A New Approach to Aircraft Assembly. In *SAE Technical Paper*. SAE International, 2007.

[9] D. Z. Chen and L. W. Tsai. The generalized principle of inertia match for geared robotic mechanisms. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1282–1287 vol.2, April 1991.

[10] J. F. Duval and H. M. Herr. FlexSEA-Execute: Advanced motion controller for wearable robotic applications. In *IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1056–1061, June 2016.

[11] B. Gao, Q. Liang, Y. Xiang, L. Guo, and H. Chen. Gear ratio optimization and shift control of 2-speed I-AMT in electric vehicle. *Mechanical Systems and Signal Processing*, 50–51:615–631, January 2015.

[12] M. Gerdts. Solving mixed-integer optimal control problems by branch&bound: a case study from automobile test-driving with gear shift. *Optimal Control Applications and Methods*, 26(1):1–18, January 2005.

[13] H. Giberti, S. Cinquemani, and G. Legnani. Effects of transmission mechanical characteristics on the choice of a motor-reducer. *Mechatronics*, 20(5):604–610, August 2010.

[14] A. Girard and H. H. Asada. A two-speed actuator for robotics with fast seamless gear shifting. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4704–4711, Hamburg, September 2015.

[15] A. Girard and H. H. Asada. A practical optimal control approach for two-speed actuators. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4572–4577, Stockholm, May 2016.

[16] A. Girard and H. H. Asada. Leveraging Natural Load Dynamics with Variable Gear-ratio Actuators. *IEEE Robotics and Automation Letters*, 2(2):741–748, April 2017.

[17] G. Goleski, D. Hoffman, and R. Lippert. Multi-speed transmission, April 2015. U.S. Patent.

[18] A. Gray, Y. Gao, T. Lin, J. K. Hedrick, H. E. Tseng, and F. Borrelli. Predictive control for agile semi-autonomous ground vehicles using motion primitives. In *2012 American Control Conference (ACC)*, pages 4239–4244, June 2012.

[19] H. Hanafusa and H. H. Asada. Stable prehension by a robot hand with elastic fingers. *Transactions of the Society of Instrumentation and Control Engineers*, pages 361–368, 1977.

[20] S. Hirose, C. Tibbetts, and T. Hagiwara. Development of X-screw: a load-sensitive actuator incorporating a variable transmission. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 193–199 vol.1, 1999.

[21] S. Hirose, K. Yoneda, K. Arai, and T. Ibe. Design of prismatic quadruped walking vehicle TITAN VI. In *International Conference on Advanced Robotics*, pages 723–728 vol.1, June 1991.

[22] F. Hogan and A. Rodriguez. Feedback Control of the Pusher-Slider System: A Story of Hybrid and Underactuated Contact Dynamics. *CoRR*, abs/1611.08268, 2016.

[23] N. Hogan and S. Buerger. Impedance and Interaction Control. In *Robotics and Automation Handbook*. CRC Press, 2004.

[24] T. Holdstock, A. Sorniotti, M. Everitt, M. Fracchia, S. Bologna, and S. Bertolotto. Energy consumption analysis of a novel four-speed dual motor drivetrain for electric vehicles. In *2012 IEEE Vehicle Power and Propulsion Conference, VPPC 2012*, pages 295–300, Seoul Olympic Parktel, Seoul, S Korea, 2012.

[25] J. Hollerbach, I. Hunter, and J. Ballantyne. A comparative analysis of actuator technologies for robotics. In *The Robotics Review*, volume 2, pages 299–342. Mit press edition, 1992.

[26] A. Jafari, N. G. Tsagarakis, I. Sardellitti, and D. G. Caldwell. A New Actuator With Adjustable Stiffness Based on a Variable Ratio Lever Mechanism. *IEEE/ASME Transactions on Mechatronics*, 19(1):55–63, February 2014.

[27] G. Kenneally, A. De, and D. E. Koditschek. Design Principles for a Family of Direct-Drive Legged Robots. *IEEE Robotics and Automation Letters*, 1(2):900–907, July 2016.

[28] B. Kim, J. Park, and J. Song. Improved manipulation efficiency using a serial-type dual actuator unit. In *International Conference on Control, Automation and Systems*, pages 30–35. IEEE, 2007.

[29] B. Kim, J. Song, and J. Park. A Serial-Type Dual Actuator Unit With Planetary Gear Train: Basic Design and Applications. *IEEE/ASME Transactions on Mechatronics*, 15(1):108–116, 2010.

[30] D. E. Kirk. *Optimal Control Theory: An Introduction*. Dover Publications, Mineola, N.Y, 2004.

[31] K. Koganezawa, T. Nakazawa, and T. Inaba. Antagonistic control of multi-DOF joint by using the actuator with non-linear elasticity. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2201–2207, May 2006.

[32] M.-O. Lacerte, G. Pouliot, J.-S. Plante, and P. Micheau. Design and Experimental Demonstration of a Seamless Automated Manual Transmission using an Eddy Current Torque Bypass Clutch for Electric and Hybrid Vehicles. *SAE International Journal of Alternative Powertrains*, 5(1):13–22, May 2016.

[33] S. M. Lavalle. Rapidly-Exploring Random Trees: A New Tool for Path Planning. Technical report, 1998.

[34] S. M. Lavalle. *Planning Algorithms*. Cambridge University Press, Cambridge ; New York, 1 edition edition, 2006.

[35] D. Leach, F. Gunther, N. Maheshwari, and F. Iida. Linear multi-modal actuation through discrete coupling. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2437–2442, October 2012.

[36] H. Lee and Y. Choi. A New Actuator System Using Dual-Motors and a Planetary Gear. *IEEE/ASME Transactions on Mechatronics*, 17(1):192–197, 2012.

[37] D. Liberzon. *Switching in Systems and Control*. Birkhäuser, Boston, MA, 2003 edition edition, 2003.

[38] T. Lozano-Pérez and M. A. Wesley. An Algorithm for Planning Collision-free Paths Among Polyhedral Obstacles. *Communications of the ACM*, 22(10):560–570, October 1979.

[39] N. Majdoub, A. Sakly, and M. Benrejeb. Hybrid approach for optimal control problem of switched systems. In *2010 IEEE International Conference on Systems Man and Cybernetics (SMC)*, pages 4161–4168, October 2010.

[40] N. McKeegan. Antonov's 3-speed transmission for electric vehicles boosts efficiency by 15 percent, July 2011. www.gizmag.com.

[41] D. Meike and L. Ribickis. Energy efficient use of robotics in the automobile industry. In *International Conference on Advanced Robotics (ICAR)*, pages 507–511, June 2011.

[42] M. Menon and H. H. Asada. Design and Control of Paired Mobile Robots Working Across a Thin Plate With Application to Aircraft Manufacturing. *IEEE Transactions on Automation Science and Engineering*, 8(3):614–624, July 2011.

[43] J. B. Morrell and J. K. Salisbury. Parallel-Coupled Micro-Macro Actuators. *The International Journal of Robotics Research*, 17(7):773–791, July 1998.

[44] F. Parietti and H. H. Asada. Supernumerary Robotic Limbs for aircraft fuselage assembly: Body stabilization and guidance by bracing. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1176–1183, May 2014.

[45] F. Parietti and H. H. Asada. Supernumerary Robotic Limbs for Human Body Support. *IEEE Transactions on Robotics*, 32(2):301–311, April 2016.

[46] W. Perruquetti and J. Barbot. *Sliding Mode Control In Engineering*. CRC Press, New York, 1st edition edition, 2002.

[47] A. W. Phillips, S. H. Wittkopp, J. M. Hart, and C. E. Carey. 10-speed transmission, November 2010. U.S. Patent.

[48] G. Pouliot, M.-O. Lacerte, J.-S. Plante, and P. Micheau. Design of an Eddy Current Torque Bypass Clutch for Seamless Automated Manual Transmissions of Electric and Hybrid Vehicles. *SAE International Journal of Alternative Powertrains*, 4:388–397, 2015.

[49] G. Pratt and M. Williamson. Series elastic actuators. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 399–406 vol.1, August 1995.

[50] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng. ROS: an open-source Robot Operating System, 2009.

[51] Q. Ren, D. Crolla, and A. Morris. Effect of transmission design on Electric Vehicle (EV) performance. In *IEEE Vehicle Power and Propulsion Conference, 2009. VPPC '09*, pages 1260–1265, September 2009.

[52] A. Richards, T. Schouwenaars, J. P. How, and E. Feron. Spacecraft Trajectory Planning with Avoidance Constraints Using Mixed-Integer Linear Programming. *Journal of Guidance, Control, and Dynamics*, 25(4):755–764, 2002.

[53] B. Roy and H. H. Asada. Nonlinear Feedback Control of a Gravity-Assisted Underactuated Manipulator With Application to Aircraft Assembly. *IEEE Transactions on Robotics*, 25(5):1125–1133, October 2009.

[54] B. Schoolcraft. Gear scheme for infinitely variable transmission, August 2011. U.S. Patent.

[55] Y. J Shin, H. J. Lee, K. S. Kim, and S. Kim. A Robot Finger Design Using a Dual-Mode Twisting Mechanism to Achieve High-Speed Motion and Large Grasping Force. *IEEE Transactions on Robotics*, 28(6):1398–1405, 2012.

[56] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[57] N. Srivastava and I. Haque. A review on belt and chain continuously variable transmissions (CVT): Dynamics and control. *Mechanism and Machine Theory*, 44(1):19–41, January 2009.

[58] N. L. Tagliamonte, F. Sergi, D. Accoto, G. Carpino, and E. Guglielmelli. Double actuation architectures for rendering variable impedance in compliant robots: A review. *Mechatronics*, 22(8):1187–1203, December 2012.

[59] K. Tahara, S. Iwasa, S. Naba, and M. Yamamoto. High-backdrivable parallel-link manipulator with Continuously Variable Transmission. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1843–1848, September 2011.

[60] G. Tonietti, R. Schiavi, and A Bicchi. Design and Control of a Variable Stiffness Actuator for Safe and Fast Physical Human/Robot Interaction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 526–531, April 2005.

[61] P. Walker, B. Zhu, and N. Zhang. Powertrain dynamics and control of a two speed dual clutch transmission for electric vehicles. *Mechanical Systems and Signal Processing*, 85:1–15, February 2017.

[62] F. Y. Wu and H. H. Asada. Hold-and-manipulate with a single hand being assisted by wearable extra fingers. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6205–6212, May 2015.

[63] X. Xu and P.J. Antsaklis. Optimal control of switched systems based on parameterization of the switching instants. *IEEE Transactions on Automatic Control*, 49(1):2–16, January 2004.

[64] Z. Zhang, C. Zuo, W. Hao, Y. Zuo, X. L. Zhao, and M. Zhang. Three-speed transmission system for purely electric vehicles. *International Journal of Automotive Technology*, 14(5):773–778, October 2013.